Sandro Bologna
Bernhard Hämmerli
Dimitris Gritzalis
Stephen Wolthusen (Eds.)

LNCS 6983

# Critical Information Infrastructure Security

6th International Workshop, CRITIS 2011
Lucerne, Switzerland, September 2011
Revised Selected Papers

Springer

# Lecture Notes in Computer Science 6983

*Commenced Publication in 1973*
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Sandro Bologna   Bernhard Hämmerli
Dimitris Gritzalis   Stephen Wolthusen (Eds.)

# Critical Information Infrastructure Security

6th International Workshop, CRITIS 2011
Lucerne, Switzerland, September 8-9, 2011
Revised Selected Papers

Springer

Volume Editors

Sandro Bologna
ENEA, S. Maria di Galeria (Rome), Italy
E-mail: sandro.bologna@enea.it

Bernhard Hämmerli
Acris GmbH, Lucerne, Switzerland
E-mail: bmhaemmerli@acris.ch

Dimitris Gritzalis
Athens University of Economics and Business, Athens, Greece
E-mail: dgrit@aueb.gr

Stephen Wolthusen
Royal Holloway, University of London, Egham, UK
and
Gjøvik University College, Norway
E-mail: stephen.wolthusen@rhul.ac.uk

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-41475-6 e-ISBN 978-3-642-41476-3
DOI 10.1007/978-3-642-41476-3
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013950109

CR Subject Classification (1998): K.6.5, C.2, K.4, D.4.6, E.3, J.1

LNCS Sublibrary: SL 4 – Security and Cryptology

*Typesetting:* Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

# Preface

This volume contains the proceedings of the 6th International Workshop on Critical Information Infrastructures Security (*CRITIS 2011*). All contributions in the volume were thoroughly reviewed by several distinguished experts and researchers from all areas of critical infrastructure protection research in a blind review process and have been revised to reflect these reviews as well as discussions and comments during the workshop.

As in previous years, the program reflects the breadth of on-going research in the field, and we have attempted to preserve the workshop character that has proven to be highly rewarding in previous years. To this end, the Program Committee selected 16 contributions from 38 submissions as full papers for an acceptance rate of 42%. Moreover, to provide an opportunity for on-going work to be presented, the committee also chose six contributions as short papers representing work in progress. The committee hopes that as in previous years, this format provided a maximum of exposure to ideas and opportunities for discussion, criticism, and the exploration of collaboration opportunities.

The 2011 workshop was organized by the University of Applied Sciences in Lucerne, Switzerland, and held in the University's Josef Mäder-Saal. CRITIS 2011 continued a successful line of workshops that have become one of the main venues for the discussion of research and policy in the area of critical infrastructure protection and was proud to continue this tradition with broad international participation by authors and contributors as well as being able to draw on the expertise of a highly international Program Committee.

The program over two days was divided into six regular sessions for talks based on full papers as well as a poster session providing for brief talks on short papers and more informal discussions. We were delighted to once again have been able to secure the presence of three distinguished speakers from important stakeholder communities in the critical infrastructure domain. Dr. Evangelos Ouzounis from the European Network and Information Security Agency (ENISA) is responsible for the security policy section of ENISA and has long-standing experience on policy, strategy, and regulatory issues related to the resilience of public communication networks. Dr. Andrea Servida has helped shape the European research and development landscape in the critical infrastructure domain in his long-standing role as deputy head of unit at the European Commission's Information Society and Media Directorate General, while Dr. Sujeet Shenoi of the University of Tulsa's Computer Science Department (Oklahoma, USA) not only has a distinguished research track record in the area but is also highly active in the international community as head of the IFIP Working Group 11.10 on Critical Infrastructure Protection and in education.

A final session of the conference was devoted to a round-table of experts on "Leveraging CIP Through Sharing Knowledge and Expertise," discussing one of the key issues at both policy and, to a lesser extent, technical levels of critical infrastructure protection. This has resulted in insights that are likely to shape on-going discussions and both national and European-level information sharing arrangements.

The workshop was fortunate to have a dedicated Technical Program Committee that provided insightful reviews and comments to contributors with typically three reviews for each submission, thereby ensuring the high caliber of contributions and the event overall. The Program Committee chairs would like to thank all members of the committee for taking the time to provide this service to the community as well as the authors of what we believe were once again stimulating and interesting days at CRITIS 2011. Organizing such workshops does, however, entail a large amount of effort that is largely invisible particularly when events are running smoothly. This is the result of considerable effort by the local organizing chairs and general chairs, and we wish to also acknowledge their efforts.

We were privileged to have had two full but rewarding days and are grateful for all efforts that went into realizing this workshop once again. As part of the workshop's objective was also to open new perspectives and vantage points, the vistas afforded by the location in Lucerne — one of the most beautiful places in Switzerland — indubitably contributed, as did the environment provided by the local organizers that gave room for further free-ranging discussions.

August 2011

Sandro Bologna
Stephen Wolthusen
Bernhard M. Hämmerli
Dimitris Gritzalis

# Organization

## Executive Committee

### General Chairs

| | |
|---|---|
| Bernhard M. Hämmerli | University of Applied Sciences Lucerne and Acris GmbH, Switzerland; Gjøvik University College, Norway |
| Dimitris Gritzalis | Athens University of Economics and Business, Greece |

### Program Chairs

| | |
|---|---|
| Sandro Bologna | AIIC, Italy |
| Stephen D. Wolthusen | Gjøvik University College, Norway; Royal Holloway, University of London, UK |

### Honorary Chairs

| | |
|---|---|
| Andrea Servida | DG INFSO, European Commission |
| Paolo Verissimo | Universidad de Lisboa, Portugal |

### Organizing Chairs

| | |
|---|---|
| Stefan Brem | CIP Coordinator, BABS, Switzerland |
| Philippe Schnyder | University of Applied Sciences, Switzerland |

## Program Committee

| | |
|---|---|
| Robin Bloomfield | City University London, UK |
| Matt Broda | Microsoft, UK |
| João Batista Camargo | University of São Paulo, Brazil |
| Genseric Cantournet | Telecom Italia, Italy |
| Emiliano Casalicchio | Università di Tor Vergata, Italy |
| Peter Daniel | Selex Communication Ltd, UK |
| Gregorio D‘Agostino | ENEA, Italy |
| Geert Deconinck | K.U. Leuven, Belgium |
| Giovanna Dondossola | RSE, Italy |
| Stelios Dritsas | Athens University of Economics and Business, Greece |
| Myriam Dunn | ETH Centre for Security Studies, Switzerland |
| Katrin Franke | Gjøvik University College, Norway |
| Claudia Eckert | Fraunhofer SIT, Germany |
| Steven Furnell | University of Plymouth, UK |
| Richard Garber | DRDC Centre for Security Science, Canada |
| Stefan Geretshuber | IABG, Germany |

| | |
|---|---|
| Robert Ghanea-Hercock | British Telecom, UK |
| Adrian Gheorghe | Old Dominion University, USA |
| Janusz Gorski | Gdansk University of Technology, Poland |
| Stefanos Gritzalis | University of the Aegean, Greece |
| Chris Johnson | Glasgow University, UK |
| Floor Koornneef | Delft University of Technology, The Netherlands |
| Panos Kotzanikolaou | University of Piraeus, Greece |
| Eric Luiijf | TNO Defence, Security and Safety, The Netherlands |
| Paulo Maciel | Federal University of Pernambuco, Brazil |
| Marcelo Masera | EU Joint Research Centre Ispra, European Commission |
| Amin Massoud | University of Minnesota, USA |
| Tom McCutcheon | Defence Science and Technololgy Laboratory, UK |
| Doug Montgomery | U.S. National Institutes of Standards and Technology, USA |
| Igor Nai Fovino | EU Joint Research Centre Ispra, European Commission |
| Eiji Okamoto | University of Tsukuba, Japan |
| Cirian Osborn | Centre for the Protection of National Infrastructure, UK |
| Evangelos Ouzounis | European Network and Information Security Agency |
| Stefano Panzieri | Roma Tre University, Italy |
| Dirk Reinermann | German Information Security Agency, Germany |
| Andrea Rigoni | Global CyberSecurity Center, Italy |
| Steven M. Rinaldi | Sandia National Laboratories, USA |
| Erich Rome | Fraunhofer IAIS, Germany |
| Michael Samsa | Argonne National Laboratories, USA |
| William H. Sanders | University of Illinois, USA |
| Roberto Setola | Università CAMPUS Bio-Medico, Italy |
| Sujeet Shenoi | University of Tulsa, USA |
| James P. Smith | Los Alamos National Laboratories, USA |
| Angelos Stavrou | George Mason University, USA |
| Neeraj Suri | TU Darmstadt, Germany |
| Barend Taute | Council for Scientific and Industrial Research, South Africa |
| Marianthi Theoharidou | Athens University of Economics and Business, Greece |
| Paul Theron | Thales Information Systems Security, France |

| | |
|---|---|
| Paul Trushell | Attorney General's Department, Australia |
| Rita A. Wells | Idaho National Laboratory, USA |
| Rhys Williams | Centre for the Protection of National Infrastructure, UK |
| Annemarie Zielstra | Centre for the Protection of National Infrastructure, The Netherlands |

# Table of Contents

# Ten National Cyber Security Strategies: A Comparison

H.A.M. Luiijf[1], Kim Besseling[1], Maartje Spoelstra[1], and Patrick de Graaf[2]

[1] TNO, P.O. Box 96864, 2509 JG The Hague, The Netherlands
[2] Capgemini Netherlands BV, P.O. Box 2575, 3500 GN Utrecht, The Netherlands
{eric.luiijf,kim.vanbuul}@tno.nl, maartje.spoelstra@gmail.com, ap.degraaf@ziggo.nl

**Abstract.** A number of nations developed and published a national cyber security strategy (NCSS). Most of them were published in the period 2009 - 2011. Despite the fact that each of these NCSS intends to address the cyber security threat, large differences exist between the NCSS approaches. This paper analyses and compares the NCSS of Australia, Canada, Czech Republic, France, Germany, Japan, The Netherlands, New Zealand, the United Kingdom, and the United States. Thirteen observations lead to a set of conclusions which nations with an NCSS and developers of future NCSS may use to their advantage.

**Keywords:** cyber security, strategy, policy, critical infrastructure, national security.

## 1 Introduction

A number of nations have developed and published a National Cyber Security Strategy (NCSS) in the period 2009–2011. During the development of the Dutch NCSS, a short analysis was made of some of earlier published NCSS. During the first half of 2011, a wider set of NCSS became available. We extended our analysis to ten NCSS (Australia, Canada, Czech Republic, France, Germany, Japan, The Netherlands, New Zealand, the United Kingdom, and the United States). As each of these ten NCSS intends to address the same global cyber security threat, one would expect a major strategic drive for international collaboration and harmonisation in the various NCSS as well as a common set of national actions. Below, we will analyse whether that is the case or not, make comparisons, and analyse the differences between the NCSS. We will analyse the underlying reasons for the differences. We will use tables as a quick reference to the main elements of the ten NCSS as the NCSS vary in size from nine up to sixty pages. The analysis and final conclusions may be of help to current NCSS implementers and to the developers of future NCSS.

## 2 Analysis Framework

To analyse and compare the ten NCCS, we look at the following topics:

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 1–17, 2013.

1. What does the notion ‘Cyber Security’ mean to nations?
2. What are the perceived threats that the various NCSS address?
3. What is the scope of the various NCSS?
4. Is there a relationship with other national strategies?
5. What are the strategic objectives and guiding principles of the NCSS?
6. Which stakeholders are addressed and how are they addressed?
7. What are the key action lines and planned actions?
8. Are emerging threats covered?
9. How are national functions institutionalised by the various NCSS?

## 3 Cyber Security: A Gamut of Definitions

Table 1 presents an overview of the various definitions and descriptive understandings of the notion ‘Cyber Security’ in the various NCSS.

Only five nations provide a definition for ‘Cyber Security’. Canada and the UK use a descriptive text to indicate what cyber security means to them. The Czech Republic, Japan and the USA do not provide a definition or description. It can be observed that some nations focus on the information security aspects whereas other nations consider Cyber Security as a property to address and counter threats from cyberspace.

From the table above, it is clear that there is no harmonised understanding of the notion ‘Cyber Security’ by the ten nations; three nations even fail to present a definition or a description of the notion in their NCSS.

**Observation 1.** An internationally accepted and harmonized definition of ‘Cyber Security’ is lacking.

All ten nations consider the cyber threat as an international threat. All NCSS plan activities for international collaboration to secure cyberspace. This requires a common understanding of notions like ‘cyber crime’, ‘cyber security’, etceteras. Recently, a joint Russian-U.S. bilateral working group of the EastWest Institute (EWI) and Moscow University has suggested a terminology framework [1] which may be a starter for harmonisation although some of the proposed cyber terminology requires more debate. According to them cyber security is “*a property of cyberspace that is an ability to resist intentional and unintentional threats and respond and recover*”. Compared to the Table 1 definitions, this definition may replace most of the current national notions and definitions when accompanying semantics clarify whether the set of threats include or exclude threats like physical and electromagnetic disruptions of cyberspace. Moreover, the risk acceptance aspect in the German definition has to be covered in some way or another.

**Observation 2.** A global harmonised definition and understanding of ‘cyber security’ (and related terminology framework) would be beneficial to all nations.

**Table 1.** NCSS definitions of Cyber Security

| | *Definition?* | *Cyber Security is …* |
|---|---|---|
| AUS | definition | Measures relating to the confidentiality, availability and integrity of information that is processed, stored and communicated by electronic or similar means. |
| CR | no definition | |
| CAN | descriptive | An appropriate level of response and/or mitigation to cyber attacks – the intentional or unauthorised access, use, manipulation, interruption or destruction (via electronic means) of electronic information and/or the electronic and physical infrastructure used to process, communicate and/or store that information. |
| FRA | definition | An information system allowing to resist likely events resulting from cyber space which may compromise the availability, the integrity or confidentiality of data stored, processed or transmitted and of the related services that Information and Communication (ICT) systems offer. |
| GER | definition | (Global) cyber security is the desired objective of the IT security situation, in which the risk of (global) cyberspace has been reduced to an acceptable minimum. *German, civil, and military cyber security are defined in similar wordings.* |
| JPN | no definition | |
| NLD | definition | Cyber security is to be free from danger or damage due to the disruption or destruction of ICT, or due to the abuse of ICT. |
| NZ | definition | The practice of making the networks that constitute cyber space as secure as possible against intrusions, maintaining confidentiality, availability, and integrity of information, detecting intrusions and incidents that occur, and responding and recovering from them. |
| UK | descriptive | Cyber security embraces both the protection of UK interests in cyber space and also the pursuit of wider UK security policy through exploitation of the many opportunities that cyber space offers. |
| USA | implicit | References to 'information security'. |

## 4 Ten National Cyber Security Strategies

Table 2 below contains base information for the NCSS of Australia (AUS), Canada (CAN), Czech Republic (CR), France (FRA), Germany (GER), Japan (JPN), The Netherlands (NLD), New Zealand (NZ), the United Kingdom (UK), and the United States (USA) such as the publication date and language(s), reference(s), and scope.

### 4.1 The NCSS – General Remarks

It is interesting to note that four of the five non-native English speaking countries have published an English translation of their NCSS simultaneously with their native language version. Most of the ten NCSS have been published for the first time. Note that an updated version of UK's 2009 NCSS is expected to appear in the Autumn of 2011. The USA strategy was published in 2003 when the notion 'cyber security' was less in use. In 2010, the Obama Administration undertook a Cyberspace Policy Review which resulted in a set of new national cyber security activities [2].

### 4.2 Scope of the NCSS

Most NCSS relate their cyber security activities to cyberspace in their descriptive texts. The German NCSS states that it considers 'only information and communication technology (ICT) connected in a certain way to Internet. The Australian and the Canadian NCSS suggest that these national strategies focus on internet connected ICT only. The Dutch NCSS explicitly states that it addresses the full range of ICT which apart from Internet-connected ICT comprises e.g., chip cards, in-car systems, and information transferral media. The other NCSS are less outspoken about this topic but do not restrict their focus.

**Observation 3.** Some NCSS are restricted to Internet-connected ICT only leaving the protection of other ICT that might very well be hampered out-of-scope.

### 4.3 Relationship with Other National Strategies

Most of the ten NCSS relate to the nations' National Security Strategies (Table 2). Most often, an earlier national threat and risk assessment is the main instigator of the NCSS development. The Dutch take a different approach. One of their NCSS actions is to deliver a national cyber threat and risk assessment for inclusion in the national risk assessment register (NRB) on a yearly basis. As a result, the NRB process may trigger the need for an update of the Dutch NCSS.

Although the cyber security threat to Critical Infrastructure (CI) is explicitly discussed by most NCSS, the relationship of NCSS with existing Critical Infrastructure Protection (CIP) strategies is less explicit. Critical (Information) Infrastructure (C(I)I) operators may become confused. Moreover, none of the European NCSS refers to the European program on CIP (EPCIP).

**Observation 4.** Most of the ten nations mention the cyber threat to their CI. Their NCSS, however, lack to clarify the relationship of existing national and international CIP strategies and the national cyber security strategy.

Most of the ten NCSS address the economical aspects of the cyberspace realm. Cyber security is considered as a minimal requirement to enhance the prosperity of the population and to foster economic welfare. The EU Digital Agenda [3] should be a driver for cyber security activities of the 28 European member countries, but only the German and Dutch NCSS refer to the Digital Agenda in their NCSS.

In most of the ten nations a discussion takes place about which governmental department or agency is the leading agency when a major cyber attack or disruption affects the nation. As part of their cyber defence strategy, nations may develop military cyber operations/ cyber defence capabilities as outlined in the British NCSS, the French NCSS reference to the French national security and defence strategy, and references in the German and Dutch NCSS to strategic Cyber Operations plans.

### 4.4 Perceived Threats

**Risk**

With respect to the perceived cyber threats, most of the ten nations explicitly mention the threats to their C(I)I and their national security. Only Australia and Canada explicitly state the cyber security risk to their defence abilities. France, despite its national defence driven NCSS and the Netherlands implicitly address this last threat.

From an economic point of view, both Germany and Japan mention the risk of stagnation of globalisation when the cyber security risk is insufficiently addressed. Related to this threat is the threat of disruption of the societal and social ICT-life of citizens. Most nations, with exception of France, Germany and the USA mention this threat. The Netherlands is the only nation which formulates the threat of loss of public confidence in the use of ICT.

**Observation 5.** Most NCSS address the general cyber crime and e-spionage type of threats. Only a small set of nations consider threats to their national defence, economy, and public confidence.

**Threat Actors**

All nations except Japan and the USA pinpoint individuals, criminals, and organised crime as threat actors. Cyber espionage (e-spionage) is mentioned by all nations but the Czech Republic and Japan. All nations but Australia and New Zealand mention the threat of hostile activities by foreign nations (e.g., cyber warfare). Despite the 2011 set of attacks in cyberspace by groups like Anonymous and LulzSec, only the Dutch and New Zealand's NCSS mention (h)activists as threat actors.

The terrorist threat to cyberspace is mentioned by all nations but Japan. There are however large differences. Some nations fear (potential) cyber attacks by terrorists on their C(I)I, something which has not occurred so far. Other nations consider information published in cyber space by terrorists, the ability for terrorists to communicate using ICT, and the gathering of intelligence on terrorists as topics that belong to their national cyber security approach.

**Observation 6.** The NCSS do not show a common understanding of the terrorist threat in cyberspace.

Both the Germany and Japanese NCSS explicitly address the threat of large-scale cyber attacks to their C(I)I. For Japan, this is not surprising as Japan has experienced several large-scale cyber attacks to its governmental and business systems in the recent past. Germany, however, has not yet experienced large-scale cyber attacks.

Both the German and Japanese NCSS mention the threat of mismatches between functional ICT developments (in other words: ICT innovation) and an appropriate level of cyber security related to those developments as a threat to be addressed. Interestingly, none of the other nations address this important topic.

The UK NCSS comprises jamming and signal modification (e.g., of GPS signals) and high-power radio frequency transmission (e.g., High Power Microwave) damaging unprotected electronics to be part of set of cyber security threats they intend to addressed. None of the other NCSS publically refer to these specific threats which are often only dealt with by the military despite growing concerns about criminal use.

**Table 2.** National Cyber Security Strategies (NCSS) ■ = explicitly described, □ = implicitly referenced

| | *AUS* | *CAN* | *CR* | *FRA* | *GER* |
|---|---|---|---|---|---|
| Reference to NCSS document<br>Other language(s) | English [4]<br>n/a | English [5]<br>French [6] | Czech<br>English [7] | French [8]<br>n/a | German [10]<br>English [11] |
| Issued | 2009 | 10.2010 | 15.07.2011 | 15.02.2011 | 23.02.2011 |
| First NCSS version? | yes | yes | yes [1] | yes | yes |
| All Cyber threats to ICT? | only Internet connected systems | only Internet connected systems | yes | yes | only Internet connected systems |
| Relates to: | | | | | |
| - National Security Strategy | ■ | ■ | ■ | ■ | ■ |
| - Critical Infrastructure Protection Strategy | | ■ | | | ■ |
| - National Digital Agenda | ■ | no | no | no | no |
| - EU Digital Agenda [17] | n/a | n/a | no | no [2] | ■ |
| - National Defence Strategy | | | | ■ [9] | □ |
| Addresses cyber threats to: | | | | | |
| - Critical infrastructure | ■ | ■ | ■ | ■ | ■ |
| - Defence abilities | ■ | ■ | | □ | |
| - Economic prosperity | ■ | ■ | ■ | | ■ |
| - Globalisation | | | | | ■ |
| - National Security | ■ | ■ | ■ | ■ | □ |
| - Public Confidence in ICT | | | | | |
| - Social Life of Citizens | ■ | ■ | □ | | |
| Addresses cyber threats from: | | | | | |
| - Activism | | | | | |
| - Criminals/Organised crime | ■ | ■ | ■ | ■ | ■ |
| - Espionage | ■ | ■ | | ■ | ■ |
| - Foreign nations/ cyber warfare | | ■ | ■ | ■ | ■ |
| - Terrorists | ■ | ■ | ■ | ■ | ■ |
| - Large-scale attacks | | □ | | | ■ |
| - Mismatch technology development and security | | | | | □ |

[1] The Czech NCSS has been issued as a draft document awaiting discussion, first in the Czech National Security Council, next by the Government of the Czech Republic.

[2] The EU Digital Agenda was published after the publication of UK's NCSS.

**Table 2.** (*continued*)

| | *JPN* | *NLD* | *NZ* | *UK* | *USA* |
|---|---|---|---|---|---|
| Reference to NCSS document | Japanese | Dutch [14] | English [16] | English [17] | English [20] |
| Other language(s) | English [12] | English [15] | n/a | n/a | n/a |
| Issued | 03.02.2009 | 22.02.2011 | 07.06.2011 | 25.06.2009 | 2003 |
| First NCSS version? | no: 2006 [13] | yes | yes | yes | yes |
| All Cyber threats to ICT? | implicitly | yes | networked systems only | yes | implicitly |
| Relates to: | | | | | |
| - National Security Strategy | | □ | | ■ [18-19] | ■ |
| - Critical Infrastructure Protection Strategy | | □ | | □ | |
| - National Digital Agenda | no | ■ | no | ■ | no |
| - EU Digital Agenda [17] | n/a | ■ | n/a | no | n/a |
| - National Defence Strategy | | □ | | □ | |
| Addresses cyber threats to: | | | | | |
| - Critical infrastructure | □ | ■ | ■ | ■ | □ |
| - Defence abilities | | □ | | | |
| - Economic prosperity | ■ | ■ | ■ | ■ | ■ |
| - Globalisation | ■ | | | | |
| - National Security | ■ | □ | ■ | ■ | ■ |
| - Public Confidence in ICT | | ■ | □ | | |
| - Social Life of Citizens | ■ | ■ | | ■ | |
| Addresses cyber threats from: | | | | | |
| - Activism | | ■ | ■ | | |
| - Criminals/Organised crime | □ | ■ | ■ | ■ | □ |
| - Espionage | □ | ■ | ■ | ■ | ■ |
| - Foreign nations / cyber warfare | ■ | ■ | | ■ | ■ |
| - Terrorists | | ■ | ■ | ■ | ■ |
| - Large-scale attacks | ■ | | | | □ |
| - Mismatch technology development and security | ■ | | | | |

**Observation 7.** Only the UK addresses the jamming, signal modification and high-power transmission threats in its national cyber security approach.

# 5 Strategic Level Topics of the NCSS

## 5.1 Strategic Objectives

Table 3 outlines the strategic objectives in the ten NCSS. Major differences in the national strategic approaches are found depending on the differences in starting points: economic prosperity, national security, or (military) defence. Apart from that, the German NCSS does not clearly state strategic objectives. It mentions a set of strategic priority areas which other NCSS present as action line. The Australian, Canadian and New Zealand's NCSS structure their strategies along an alike three-fold approach: government, critical businesses, and citizens/individuals.

**Table 3.** Strategic objectives of the ten NCSS

| | |
|---|---|
| AUS | The maintenance of a secure, resilient and trusted electronic operating environment that supports Australia's national security and maximises the benefits of the digital economy:<br>1. All Australians are aware of cyber risks, secure their computers and take steps to protect their identities, privacy and finances online;<br>2. Australian businesses operate secure and resilient ICT to protect the integrity of their own operations and the identity and privacy of their customers;<br>3. The Australian government ensures its own operations and the identity and privacy of their customers. |
| CAN | Meeting the Cyber Security threat by:<br>1. Securing government systems;<br>2. Partnering to secure vital cyber systems outside the federal government;<br>3. Helping the Canadians to be secure online. |
| CR | To maintain a safe, secure, resistant and credible environment that makes use of available opportunities offered by the digital age. |
| FRA | 1. To be a world power in cyber defence;<br>2. To guarantee the French national freedom to decide by protecting national information;<br>3. To reinforce the cyber security of critical infrastructures;<br>4. To ensure the safety in the cyberspace. |
| GER | Strategic security areas rather than objectives are presented:<br>1. Protection of Critical Infrastructures;<br>2. Secure IT systems in Germany;<br>3. Strengthening IT security in the public administration;<br>4. National Cyber Response Centre;<br>5. National Cyber Security Council;<br>6. Effective crime control in cyberspace;<br>7. Effective coordinated action to ensure cyber security in Europe and worldwide;<br>8. Use of reliable and trustworthy IT;<br>9. Personnel development in federal authorities;<br>10. Tools to respond to cyber attack. |
| JPN | 1. Reinforced policy to counter cyber attacks;<br>2. Policies to adapt to changes in cyber security environment;<br>3. Active/dynamic cyber security measures (see [12]). |
| NLD | To reinforce the security of the digital society, in order to increase confidence in the use of ICT by citizens, business and government in order to stimulate the Dutch economy and to increase prosperity and well-being of its citizens.<br>Proper legal protection in the digital domain is guaranteed and societal disruption is prevented. Adequate action will be taken if things were to go wrong. |
| NZ | 1. Raise awareness and on-line security of individuals and small businesses;<br>2. Protecting government systems;<br>3. Build strategic relationships to improve cyber security for critical infrastructure and other businesses. |
| UK | Citizens, business and government can enjoy the full benefits of a safe, secure and resilient cyber space. The government will secure UK's advantage in cyberspace by reducing risk, and exploiting opportunities in cyber space by improving knowledge, capabilities and decision-making. |
| USA | 1. Prevent cyber attacks against America's critical infrastructure;<br>2. Reduce national vulnerability to cyber attacks;<br>3. Minimize damage and recovery time from cyber attacks that do occur. |

**Table 4.** Guiding principles of the ten NCSS

| | |
|---|---|
| AUS | 1. National leadership.<br>2. Shared responsibilities.<br>3. Partnerships.<br>4. Active international engagement.<br>5. Risk management.<br>6. Protecting Australian values. |
| CAN | See remark in text. |
| CR | 1. Abide the principles of a democratic society and duly consider legitimate interests of its citizens, business sector and public administrations and agencies in relation to citizens.<br>2. Adequate cyber security measures to protect and guarantee national security will respect privacy, fundamental rights and liberties, free access to information, and other democratic principles.<br>3. National cyber security measures balance the need to guarantee security with the respect for fundamental rights and liberties. |
| FRA | None. |
| GER | All stakeholders have to act as partners and fulfil protection tasks together. Enforcement of international rules of conduct, standards and norms. |
| JPN | None. |
| NLD | 1. Linking and reinforcing existing cyber security initiatives.<br>2. Public-private Partnership and clear responsibilities, powers & safeguards.<br>3. Individual responsibility to secure cyberspace (citizens, businesses, the public administration and its agencies).<br>4. Active international collaboration.<br>5. Security measures are balanced and proportional with respect to public and national security versus safeguarding of fundamental human rights.<br>6. Self-regulation if possible, legislation and regulation when required. |
| NZ | None. |
| UK | 1. Set of core values: human rights, rule of law, legitimate and accountable government, justice, freedom, tolerance, and opportunity for all.<br>2. Hard-headed about risk, aims, and capabilities.<br>3. Tackle security challenges early.<br>4. Nationally, partnership approach; internationally a multilateral approach; internal government an integrated approach.<br>5. Retain strong, balanced and flexible capabilities.<br>6. Continue to invest, learn and improve to strengthen UK's security. |
| USA | Privacy and civil liberties need to be protected. |

The French NCSS strategic objectives stem from a national power projection point of view. France is the only of the ten nations which takes that approach, although some other NCSS support power projection. The UK, for instance, makes clear that it wants to gather and use intelligence on criminals, terrorists, and other adverse actors in cyberspace. Explicitly, their NCSS mentions the exploitation of such information and the disruption of adversary activities. Recently, it was published that MI6 hacked into Al Qaeda's on-line magazine Inspire and replaced an article on 'Make a bomb in the Kitchen of your Mom' with a page of recipes for 'The Best Cupcakes in America' [21].

Despite the differences in wording, most NCSS aim for a safe, secure and resilient ICT environment for the citizens, society, and economic prosperity. As only nation, Japan recognises the need for agile adaption to new and upcoming cyber security threats in their set of strategic objectives.

**Observation 8.** All but one NCSS lack a strategic objective which reflects the need for agile adaption to emerging cyber security threats.

## 5.2 Guiding Principles and Framework Conditions

Seven of the ten nations relate the content of their NCSS to guiding principles or framework conditions (see Table 4 above). Although Canada does not explicitly list any guiding principle in their NCSS, they consider most of the guiding principles of the USA, UK and Australia to resemble their own. Each of the seven nations considers the protection of civil liberties and other (inter)national democratic core values as guiding principles to their NCSS. UK's guiding principles are by far the most outspoken reassuring their citizens about the basics of its national cyber security approach.

**Observation 9.** The NCSS of France, Japan and New Zealand lack guiding principles/ framework conditions for their cyber security actions and activities.

**Table 5.** The NCSS directly addresses the following types of stakeholders with respect to threats, vulnerabilities and measures (□ when discussed in NCSS but limited set of actions/activities)

| | *Citizens* | *SME* | *ISP* | *Large organisations* | *CI Operators* | *The State / national security* | *Global infrastructure & issues* |
|---|---|---|---|---|---|---|---|
| AUS | ■ | ■ | ■ | ■ | ■ | ■ | □ |
| CAN | ■ | □ | □ | ■ | ■ | ■ | □ |
| CR | ■ | ■ | ■ | ■ | ■ | ■ | |
| FRA | ■ | ■ | □ | ■ | ■ | ■ | □ |
| GER | ■ | ■ | ■ | □ | ■ | ■ | ■ |
| JPN | □ | □ | | □ | ■ | ■ | □ |
| NLD | ■ | ■ | □ | ■ | ■ | ■ | □ |
| NZ | ■ | ■ | ■ | ■ | ■ | ■ | |
| UK | ■ | ■ | □ | ■ | ■ | ■ | ■ |
| USA | ■ | ■ | □ | ■ | ■ | ■ | ■ |

## 5.3 Stakeholders

With respect to stakeholders, the Japanese NCSS limits itself to the government and the critical sectors (see Table 5). Internet Service Providers (ISP) are only explicitly addressed by the Australian, Czech, German and New Zealand's NCSS. Australia's ISP, supported by the Australian government, undertake a set of joint activities to raise the cyber security of their operations and their customers. An ISP Code of Practice and identifying compromised customer systems are part of that approach (see [1]). Germany, the UK and the USA NCSS explicitly consider the global cyber infrastructures as stakeholders despite that it will be hard to pinpoint who is responsible.

# 6 Tactical/Operational Level Topics of the NCSS

## 6.1 Key Action Lines and Planned Actions

As Table 6 shows, most of the NCSS present a limited set of planned action lines and related sets of, often operational, subsidiary actions. Where feasible, we directly refer to numbering in the specific NCSS.

Both Japan and the USA are the only nations explicitly addressing the dynamics of the cyber security threat. Japan sees the agile adaption to emerging cyber security threats even as a strategic objective. Japan approaches the cyber security issues more from a wider (holistic) security perspective than the other nations. Some nations mention specific emerging cyber threats in their NCSS such as France and Japan which plan to address the cyber security of cloud computing. Japan also plans to address the security of IP version 6 and of home appliances taking part in smart grids.

All nations address the protection of their critical infrastructures and their critical information infrastructures including the government's own ICT. Some nations refer in their NCSS to already existing activities rather than starting new ones. Only some of the ten nations refer to their military cyber security capabilities and plans. The Dutch NCSS points to cyber operations structures and activities planned by the Ministry of Defence which were published as part of the Defence reform plans shortly after the Dutch NCSS was published. In a similar way, the German NCSS points to cyber operations plans of the German Armed Forces (Bundeswehr).

Most nations have cyber security awareness programs and plans for cyber security education. Apart from community-wide programs, some nations (e.g., Germany, the Netherlands, UK and USA) develop high-priority programs to educate and train a large number of cyber defence and law enforcement experts. Apart of New Zealand and the UK, all nations work on specific ICT crisis management measures to address major cyber-related disruptions. National and sector-specific exercises are often related to these activities. At the same time, most NCSS refer to the development of national detection capabilities and national response capabilities.

Most NCSS mention international collaboration as an action line or high priority topic. However, only a few specific actions are mentioned in the various NCSS. This despite the fact that the majority of the cyber threats require swift collaborative international action as adversaries and cyber criminals will not wait until multiple national authorities finally agree to act. Germany, The Netherlands and USA expressed that they intend to promote the Cybercrime Convention to other nations [22]. Canada intends to ratify the Cybercrime Convention treaty; the UK did that recently. The Czech Republic intends to update their legislation and to mandate a set of cyber security standards to protect their government systems and their C(I)I.

**Table 6.** Key action lines and planned actions ■ = specific activities; □ implicitly indicated

| *Key action lines* | *AUS* | *CAN* | *CR* | *FRA* | *GER* |
|---|---|---|---|---|---|
| Active/dynamic security measures | | | | | |
| Awareness & training/ Information Security Campaign | ■ | ■<br>(objective 3) | ■ | action 7 | action 2 |
| Adaptable policy to new ICT risk | | | | | |
| Continuity & contingency plans | | | | | |
| Critical Infrastructure Protection | ■ | ■ | ■ | action 4<br>(objective 3) | action 1 |
| Cryptographic Protection | | | | ■ | (action 8) |
| Defence Cyber Operations/ intervention, training & exercises | | ■ | | □ | ■ |
| Economic growth | ■ | ■ | ■ | | |
| Education | ■ | ■ | ■ | ■ | (action 9) |
| Exercises | ■ | ■ | | | ■ |
| Explicit holistic view | | | | | |
| Exploitation to combat threats | | | | | |
| ICT crisis management | ■ | ■ | ■ | ■ | action 4 |
| Improved security of ICT products | | | | | |
| Information Exchange (PPP) | ■ | | | | |
| Information Sharing | ■ | ■ | ■ | | action 4 |
| Intelligence gathering on threat actors | ■ | ■ | | | |
| International collaboration | ■ | ■ | ■ | action 6 | action 7 |
| Knowledge development | | | | | |
| Legislation | | | ■ | | |
| Mandating security standards | | | ■ | | |
| National Detection Capability | ■ | ■ | ■ | action 2 | |
| National Response Capability | ■ | ■ | ■ | action 2 | action 4 |
| Privacy protection | ■ | ■ | | □ | |
| Promote Cyber Crime Convention | | □ | | | action 6 |
| Protection of non-critical infra | ■ | ■ | ■ | | |
| Public-private Partnership | ■ | (objective 2) | ■ | | |
| Reducing adversary's motivation & capabilities | | | | | |
| Research & development | ■ | ■ | ■ | action 3 | |
| Resilience against disturbances/ threat & vulnerability reduction | ■ | | | action 4 | |
| Secure protocols and software | | | | ■ | action 2 |
| Secure sourcing of products | | | | ■ | action 8 |
| Self Protection of the Government | ■ | (objective 1) | ■ | ■<br>(objective 2) | action 3 |
| Strategic Cyber Security Council | | | ICBCS | | action 5 |
| Threat & vulnerability analysis | ■ | ■ | ■ | action 1 | action 4 |
| Tracing criminals & Prosecution | ■ | ■ | | action 5 | action 6 |
| Actions defined in SMART way? | no | no | no | no | no |

**Table 6.** (*continued*)

| *Key actions and action lines* | *JPN* | *NLD* | *NZ* | *UK* | *USA* |
|---|---|---|---|---|---|
| Active/dynamic security measures | ■<br>(objective 3) | | | | ■ |
| Awareness & training/ Information Security Campaign | ■ | on-going; intensify | ■ | ■ | priority 3 |
| Adaptable policy to new ICT risk | ■<br>(objective 2) | | | | |
| Continuity & contingency plans | ■ | telecom law | | telecom law | □ |
| Critical Infrastructure Protection | action line 1 | on-going | ■ | ■ | on-going |
| Cryptographic Protection | ■ | | | | |
| Defence Cyber Operations/ intervention, training & exercises | | ■ | | ■ | ■ |
| Economic growth | action line 4 (objective) | □<br>(objective) | | □<br>(objective) | □ |
| Education | | action line 6 | | ■ | □ |
| Exercises | | ■ | ■ | □ | □ |
| Explicit holistic view | action line 3 | | | | priority 5 |
| Exploitation to combat threats | | | | ■ | |
| ICT Crisis Management | action line 2 | ■ | | | ■ |
| Improved security of ICT products | | ■ | | | |
| Information Exchange (PPP) | | ■ | | ■ | |
| Information Sharing | | ■ | | | ■ |
| Intelligence gathering on threat actors | | ■ | | ■ | |
| International collaboration | action line 5 | ■ | | ■ | priority 5 |
| Knowledge development | | ■ | | ■ | |
| Legislation | | | review | | |
| Mandating standards | | | | | |
| National Response Capability | | action line 4 | ■ | | priority 1 |
| Privacy protection | ■ | ■ | | | |
| Promote Cyber Crime Convention | | ■ | considering | [3] | ■ |
| Protection of non-critical infra | □ | □ | | □ | |
| Public-Private Partnership | ■ | action line 1 | | ■ | |
| Reducing adversary's motivation & capabilities | | | | ■ | |
| Research & development | | action line 6 | ■ | ■ | ■ |
| Resilience against disturbances/ threat & vulnerability reduction | action line 1 | action line 3 | ■ | ■ | priority 2 |
| Secure protocols and software | | | | | ■ |
| Secure sourcing of products | | | | □ | |
| Self Protection of the Government | ■ | ■ | ■ | ■ | priority 4 |
| Strategic Cyber Security Council | | all actors | | only gov. | |
| Threat & vulnerability analysis | | action line 2 | | | ■ |
| Tracing criminals & Prosecution | ■ | action line 5 | | | ■ |
| Actions defined in SMART way? | yes | no | no | no | no |

[3] The UK ratified the Cyber Crime Convention In May 2011.

The Netherlands NCSS intends to put the software security quality issue on the international agenda. Software liability may reduce the amount of insecure software being delivered to the market.

As discussed before, the UK plans to gather intelligence and use that to reduce the motivation and capabilities of adversaries operating in cyberspace as part of the exploitation objective in their NCSS.

Only the French and German NCSS explicitly refer to secure sourcing and own development of so-called government-off-the-shelf (GOTS) hardware and software to be used as part of the critical and sensitive government infrastructures and sometimes in national critical infrastructure. The UK implicitly mentions its information assurance agencies. The other NCSS do not make clear whether GOTS hardware and software is a high priority issue or not.

Germany, Japan and the Netherlands plan a cyber security council (CSC) at the strategic level. The Japanese one is an intra-governmental board. The Dutch CSC will have members from public, private, and R&D institutions/academic organizations. The German CSC will be a council in which private stakeholders may participate as observers.

Because of the sense of urgency expressed by most NCSS, one would expect that most actions would be defined in a SMART way: Specific, Measurable, Achievable, Realistic and Timely. Apart from the Japanese NCSS and some minor actions mentioned in other NCSS that is not the case.

**Observation 10.** The NCSS lack a notion of collaborative international detection and response capabilities.

**Observation 11.** The Japanese NCSS takes a wide view to cyber security and includes an agile adaptation to emerging cyber security threats.

**Observation 12.** The Netherlands requests international action to enhance the software security quality globally by promoting software liability.

**Observation 13.** Only one of the ten NCSS defines its set of planned actions in a SMART way. Therefore, most nations are unable to measure and determine afterwards whether their strategy is a success and where strengthening is required by taking additional measures.

### 6.2 NCSS Institutionalisation by the Various Nations

Table 7 shows that most nations plan to institutionalise by enlarging mandates and efforts of existing government organisations and agencies like The Netherlands and the UK. Australia, Czech Republic, and Germany create new cyber security operational centres. Germany, the Netherlands and the UK will establish cyber security councils at the strategic level. Germany and the Netherlands refer to new military operational cyber security capabilities; Canada will extend their existing defence capabilities.

**Table 7.** NCSS institutionalisation (CS = Cyber Security)

| | *AUS* | *CAN* | *CR* | *FRA* | *GER* |
|---|---|---|---|---|---|
| Extends existing organisations | | CSE; DND/CF | | | BSI |
| Establishes new organisations | CERT AUS;<br>CS Operations Centre (CSOC) | | Interdpt. Coordination Board for CS (ICBCS);<br>CERT-CR | | National CS Council;<br>National Cyber Abwehrzentrum (NCAZ) |

| | *JPN* | *NLD* | *NZ* | *UK* | *USA* |
|---|---|---|---|---|---|
| Extends existing organisation(s) | National Information Security Center (NISC) | GovCERT.nl<br>KLPD/THTC | National CS Centre (absorbs CCIP) | | DHS as centre of excellence on cyber security |
| Establishes new organisation(s) | | Nationale CS Raad (NCSR);<br>National CS Centre (NCSS);<br>Defence Cyber Expertise Centre | | Office of CS (OSC)<br>CS Operations Centre (CSOC) | |

## 7 Conclusions

Only half of the ten NCSS are based on a strict definition of cyber security. The other nations either use descriptive text or a kind of ‘common understanding’. Because of the lack of a harmonized terminology set, nations will be hampered in collaboratively addressing threats to cyber space.

Comparing the ten NCSS, major differences in approaches stemming from the differences in starting points are found: economics, national security, or military defence. Another major difference is the scope of cyber security: internet connected systems only versus the whole of ICT. Most NCSS lack a holistic approach to the threats to cyberspace; only the UK explicitly mentions the electromagnetic spectrum threats to cyberspace. Emerging cyber security threats are only explicitly addressed by Japan in their NCSS.

Most NCSS recognise the need for a society-wide approach: citizens, businesses, the public sector, and the government. However, the set of actions specially aimed at citizens is most often limited to awareness campaigns and minor security education actions at schools. Only Australia has an outreach program which supports the citizens with national cyber security tools. This is also a demonstration that most nations underrate the (inter)national risk of loss of public confidence in ICT which may seriously hamper economic prosperity.

Most NCSS are developed without a clear descriptive section on how the NCSS relates to existing national and international strategies and policies, such as the protection of critical infrastructures.

All NCSS recognise the international cyber security threat and plan weakly described activities for international legal and operational collaboration. Given the threats and the cyber security trouble most nations experience on a daily basis, a more aggressive approach and leadership is expected, especially from the EU nations. In May 2011, the US issued their International Strategy for Cyberspace [23] and ask other nations to endorse the guiding principles, to harmonise legal approaches (with an explicit reference to the Council of Europe Cybercrime convention [20]), to build and enhance military alliances to 'confront potential threats in cyberspace', and to work on the governance issues.

Only one NCSS addresses the issue of insecure software and the need for software manufacturers to be held accountable.

Last but not least, all but one NCSS is developed with national political sensitivities and the departmental playing fields in mind. As a result, all activity lines and set of actions are far from being SMARTly defined. This may cause less progress to be made when the national political focus temporarily shifts. Given the sense-of-urgency expressed in almost all NCSS, this may result in a boomerang effect to nations when they are not properly prepared for dealing with the cyber security risk.

## References

1. Rauscher, K.F., Yashenko, V. (eds.): Critical Technology Foundations. EastWest Institute, London (2011), http://www.ewi.info/system/files/reports/Russia-U%20S%20%20bilateral%20on%20terminology%20v76%20%282%29.pdf
2. The White House, Cyberspace Policy Review: Assuring a Trusted and Resilient Information and Communications Infrastructure, Washington, DC, USA (2010), http://www.whitehouse.gov/assets/documents/Cyberspace_Policy_Review_final.pdf
3. European Commission, A Digital Agenda for Europe – COM(2010) 245 final/2, Brussels, Belgium (2010), http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:52010DC0245R(01):EN:NOT
4. Attorney General, Cyber Security Strategy, Australia (2009), http://www.ag.gov.au/www/agd/agd.nsf/Page/CyberSecurity_CyberSecurity#h2strategy
5. Public Safety Canada, Canada's Cyber Security Strategy: For a stronger and more prosperous Canada (2010), http://www.publicsafety.gc.ca/prg/ns/cybr-scrty/_fl/ccss-scc-eng.pdf
6. Sécurité publique Canada, Stratégie de cybersécurité du Canada: Renforcer le Canada et accroître sa prospérité, Ottawa, Canada (2010), http://www.publicsafety.gc.ca/prg/ns/cybr-scrty/_fl/ccss-scc-fra.pdf
7. Ministry of Interior, Cyber Security Strategy of the Czech Republic for the 2011-2015 period (Strategii pro oblast kybernetické bezpečnosti České republiky na období 2011-2015), Prague, Czech Republic (2011), http://www.enisa.europa.eu/media/news-items/CZ_Cyber_Security_Strategy_20112015.PDF
8. Secrétariat général de la défense et de la sécurité nationale, Défense et sécurité des systèmes d'information: Stratégie de la France (2011), http://www.ssi.gouv.fr/IMG/pdf/2011-02-15_Defense_et_securite_des_systemes_d_information_strategie_de_la_France.pdf

9. Secrétariat général de la défense et de la sécurité nationale, Défense et Sécurité nationale: Le Livre Blanc, Paris, France (2008), http://www.defense.gouv.fr/portail-defense/enjeux2/politique-de-defense/livre-blanc-2008
10. Bundesministerium des Innern, Cyber Sicherheitsstrategie für Deutschland (2011), http://www.bmi.bund.de/SharedDocs/Downloads/DE/Themen/OED_Verwaltung/Informationsgesellschaft/cyber.pdf?__blob=publicationFile
11. Federal Ministry of the Interior, Cyber Security Strategy for Germany, Germany (2011), http://www.cio.bund.de/SharedDocs/Publikationen/DE/Strategische-Themen/css_engl_download.pdf?__blob=publicationFile
12. Information Security Strategy for Protecting the Nation, Information Security Policy Council, Tokyo, Japan (2010), http://www.nisc.go.jp/eng/pdf/New_Strategy_English.pdf
13. The First National Strategy on Information Security: towards the realization of a trustworthy society, Information Security Policy Council, Tokyo, Japan (2006), http://www.nisc.go.jp/eng/pdf/national_strategy_001_eng.pdf
14. Netherlands Ministry of Security and Justice, De Nationale Cyber Security Strategie: Slagkracht door samenwerking, The Hague, The Netherlands (2011), http://www.rijksoverheid.nl/bestanden/documenten-en-publicaties/rapporten/2011/02/22/nationale-cyber-security-strategie-slagkracht-door-samenwerking/de-nationale-cyber-security-strategie-definitief.pdf
15. Netherlands Ministry of Security and Justice, The National Cyber Security Strategy (NCSS): Success through Cooperation, The Hague, Netherlands (2011), http://www.enisa.europa.eu/media/news-items/dutch-cyber-security-strategy-2011
16. Ministry of Economic Development, New Zealand's Cyber Security Strategy, New Zealand (2011), http://www.dpmc.govt.nz/dpmc/publications/nzcss
17. Cabinet Office, Cyber Security Strategy of the United Kingdom: safety, security and resilience in cyber space, London, United Kingdom (2009), http://www.official-documents.gov.uk/document/cm76/7642/7642.pdf
18. HM Government, The National Security Strategy Update 2009: Security for the Next Generation, London, United Kingdom (2009), http://www.official-documents.gov.uk/document/cm75/7590/7590.pdf
19. HM Government, A Strong Britain in an Age of Uncertainty: The National Security Strategy, London, United Kingdom (2010), http://www.direct.gov.uk/prod_consum_dg/groups/dg_digitalassets/@dg/@en/documents/digitalasset/dg_191639.pdf?CID=PDF&PLA=furl&CRE=nationalsecuritystrategy
20. The National Strategy to Secure Cyberspace, The White House, USA (2003), http://www.us-cert.gov/reading_room/cyberspace_strategy.pdf
21. http://lafiga.firedoglake.com/2011/06/03/finally-an-intelligent-use-for-cupcakes-hacking-terrorist-sites (last visited June 30, 2013)
22. Council of Europe, Convention on Cybercrime, ETS No. 185 (2001), http://conventions.coe.int/Treaty/EN/Treaties/html/185.htm
23. The White House, International Strategy for Cyberspace, Washington, DC, USA (2011), http://www.whitehouse.gov/sites/default/files/rss_viewer/international_strategy_for_cyberspace.pdf

# Analysis of Dependencies in Critical Infrastructures

Adedayo O. Adetoye, Michael Goldsmith, and Sadie Creese

e-Security Group, International Digital Laboratory
University of Warwick, Coventry, CV4 7AL, UK
{a.adetoye, m.h.goldsmith,s.creese}@warwick.ac.uk
http://www.digital.warwick.ac.uk

**Abstract.** Unforeseen and unexpected dependencies and interactions within a critical infrastructure (CI) network may pose serious threats, and the lack of knowledge or understanding of such dependencies can be a risk to the system. This is true whether events that propagate adverse impacts through these dependencies have malicious intent or not. We therefore present a framework for modelling and reasoning about dependencies within CIs. The framework includes a domain-specific modelling language for CI dependencies and configuration, a formalism and calculus for reasoning about dependencies within the CI model, and tools to provide analytical capabilities that may be used for decision support during risk assessment and analysis of CIs.

**Keywords:** Dependency Taxonomy, CI Interdependency, Analytical Tools.

## 1 Introduction

The level of threat faced by CIs is widely evidenced internationally. There is a wealth of evidence that suggests that many failures of CIs in various parts of the world are due to unanticipated dependencies. For example, recent cascading failures in telecommunication services within the UK are supposedly due to unexpected dependencies, or at best to ones that are known but considered to be very unlikely cause of significant problems. Dependencies can exist across and within CIs at the sector level as well as the organisational level, and may even cross national boundaries. While dependencies may exacerbate the impacts of failures within CIs on the services provided by the infrastructure, their reach may propagate the impact of failures to people and societies far away from the original source of the failure. For example, there is ample anecdotal evidence that there exist financial interdependencies within the international banking system, that contributed to the global economic crisis of 2007. The crisis has had a far-reaching impact on global economies. Fong *et al* [7] explain in their report that the global banking problem during the financial crisis of 2007-2009 highlights the importance of monitoring the *interconnectivity* among financial institutions and financial systems. The interconnectivity suggests the existence of dependencies, which often serve as bases for the propagation of financial contagions [10,4,8].

This research work is carried out in the context of the project *SATURN* (**S**elf-organising **A**daptive **T**echnology **U**nderlying **R**esilient **N**etworks)[1], a collaboration

[1] www.saturn-project.org.uk

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 18–29, 2013.

between UK industry and academia, which is sponsored by the UK Technology Strategy Board and the Engineering and Physical Sciences Research Council. The project *SATURN* seeks to develop methodologies for resilient, self-healing CIs. One of the overarching objectives of the project *SATURN* is to understand the nature and communicability of dependencies between organisations. The importance of knowing and understanding the dependencies that may exist within and across CIs and the fact that such networks are typically large and evolving, necessitates the need for models and tools, which can systematically identify, reason about and extract (in particular, the subtle) dependencies that may exist within the CI. This paper presents our initial research into formal modelling and analysis techniques as a foundation for the automated exploration of the CI dependencies. We have begun to extend the basic foundation presented in this paper, for example, by introducing automated *What If?* analysis into the analytical framework for the exploration and discovery of CI dependencies [5]. Our ultimate aim is to develop a modelling capability which can identify potentially unknown dependencies automatically.

### 1.1 Related Work

The study of infrastructure dependency is a well-established area. To the best of our knowledge, we are the first to introduce a formal calculus as a basis for reasoning about infrastructure dependencies, introducing key notions such as co-dependency and redundant dependency to respectively reason systematically about single-points-of-failure and redundancies within the CI. The classical taxonomies of dependency, such as those proposed in [13,6,12,14], classify dependency based on a series of types which describe the fundamental reason for the dependency. In the context of critical infrastructure, Rinaldi *et al* [14] propose the following dimensions of infrastructure dependencies, namely, *Physical*, *Cyber*, *Geographic*, and *Logical* dependencies. In [13], three general categories of failures induced by CI dependencies are identified, namely, *cascading*, *escalating*, and *common-cause failures*. In [2], *failure* is generally viewed as a threat to the dependability and security of computer-based systems (for example, modern CIs). Graph-theoretic models of dependencies have been well-studied [15,6,3,9]. In [15] the authors develop a flexible way for describing the behaviours of nodes in the network while incorporating various models of network services. The approach is based on service dependencies between infrastructure nodes which is validated within a simulation environment. Our formalism, however, does not limit the network nodes considered to infrastructure components, but also admits actors from the environment such as people, natural phenomena (earthquakes, floods, etc.), and other externalities which along with their impacts can be incorporated directly into the model. This potentially allows richer scenarios to be reasoned about during analyses and risk assessments. In [11], a dependency model based on *Quality* (various properties and indicators such as *quantity*, *speed* and *reliability*), *Response* (*input* and *time*-induced variability in service output), *State* of operation (*normal*, *stressed*, *crisis* and *recovery* states) and *Environmental* factors are considered. The basic idea is to support richer models of dependencies that can capture real-world scenarios. While the result presented in this paper focuses on a foundational calculus of dependency, our recent extensions [5] can currently capture more generally (with the exception of timing issues) all the four areas identified in [11].

## 2 Dependency Modelling

We refer to the notion of *dependency* in a broad sense to encompass situations where variations in the behaviour of one entity influences, or has effect on, another. That is, an entity $A$ is dependent on another entity $B$ if events associated with $B$ have an impact on $A$. In *SATURN* we are interested in the consequences of dependencies in CIs, and specifically in understanding where the *failure* of one subsystem may have an impact on other parts of the system which depend upon it. The notion of *failure* is used in a very broad sense to mean that the entity in question does not perform to some predefined expectation. This is consistent with the definition of *service failure* in [2]. In particular, the aim of our analysis framework is to provide a mechanism for identifying failure-induced dependencies which may not be immediately obvious to stakeholders due to the typical scale and complexity of real systems.

Whilst our emphasis is on *failure dependency* our analyses can be transposed to general dependency settings, which are more broadly concerned about the effects of variations in behaviour that are not necessarily related to failures (e.g. resulting from normal operations). In the literature the terms *interdependency* and *dependency* are commonly used interchangeably. The reader should note however that, strictly speaking, interdependency has to do with the mutual dependency of two entities. It is also useful to bear in mind that dependency is the more primitive construct: think of interdependency as two dependency arrows going in the opposite directions. Indeed, there are many types of dependency which may be of interest when considering CI dependency issues.

### 2.1 Dependency Taxonomies

The taxonomies [6,12,14] mentioned above instantiate classes of dependencies, which we believe can be clustered and reasoned about in a principled way. In particular, because we focus on analytical tools for modelling and reasoning about dependencies in CIs we shall consider *classes* of dependency rather than *instances* such as those proposed in those taxonomies. To illustrate this distinction, notice, for example, that the geographical or geo-spatial dependency [12,14], which describes dependencies due to geo-spatial proximity, is a particularly useful way to assess the dependency issues for physical assets: such a classification could tell us that if there is a disaster within a relevant proximity of two entities that are geo-spatially close enough to each other, then this may affect both. However, such a classification is not sufficient for our purposes because it narrows down the application by tying the reasoning to the specific geo-spatial type of dependency, and we thus miss out on a category of reasoning technique that detects dependencies induced by any ‘common cause’ or ‘common source’ or ‘common facilitator’. Such common-* dependencies appear in many guises, but all of these have a similar risk-mitigation strategy: to eliminate or reduce the reliance on the common factor. In a geo-spatial setting, the risk reduction strategy may be to avoid co-locating CI in areas where their proximity can lead to their common failure, for example, due to natural or man-made disasters. By focusing on classes, rather than instances, of dependencies, we hope to develop a more generic reasoning capability for classes of dependencies that may exist within CIs. We now present the dependency classes that form the basis of our analyses.

**Generic Dependency.** We define a *dependency* relationship between two entities to exist if one entity relies on, or is impacted by, events associated with another (where event is taken to be a general concept that has to do with the action or state of an entity or its environment). If some event associated with an entity $A$ has an effect on the entity $B$, then we say that $B$ is *dependent* on $A$. We denote the dependency relationship between $A$ and $B$ in a dependency graph via an arrow pointing away from the dependency source to the dependent entity, as shown in Fig. 1. Since a dependency relationship between two entities is not necessarily symmetric, we use a *directed graph* to model it, where the direction of the graph edge denotes the "direction" of the dependency relationship. Formally, we write $B\langle D\rangle A$ whenever $B$ is dependent on $A$. We use the concept of an *entity* to refer to a variety of components and actors (such as people) within the CI. For example, an entity may be an asset or a group of assets, an organisation or a group of organisations, or externalities such as natural and man-made disasters which may impact other entities within the CI.

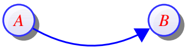


**Fig. 1.** Dependency Graph showing the dependency of $B$ on $A$

**Indirect Dependency.** We say that an entity $C$ is indirectly dependent on an entity $A$ when events in $A$ indirectly influence events in $C$ by first inducing events in a third entity $B$, which in turn induce the events in $C$.

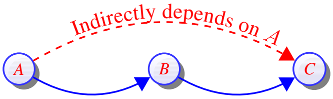
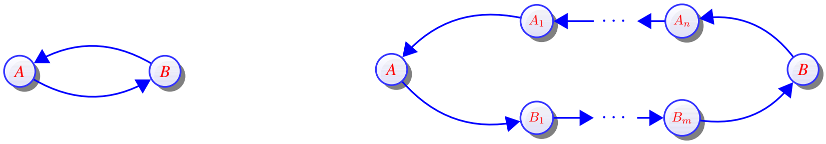


**Fig. 2.** Indirect dependency of $C$ on $A$

The problem of *cascading failure* (which occurs when a disruption in one infrastructure causes the failure of a component in a second infrastructure, which subsequently causes a disruption in the second infrastructure [14]) is closely associated with the existence of indirect dependencies (although our model considers cascade effects between entities which may or may not be infrastructures). Such cascade effects would occur when a failure at one node causes a chain of failures as the failure propagates along the dependency chain – thus evidencing an indirect dependency. Formally, there exists an indirect dependency between $C$ and $A$, whenever if $C\ \langle D\rangle\ B$ and $B\ \langle D\rangle\ A$ hold then $C\ \langle D\rangle\ A$ also holds. This relationship is depicted in Fig. 2. Because of indirect dependency, we say that the dependency relation is transitive. Any number of entities could be involved in an indirect dependency chain, and interdependencies (described next) can arise through an arbitrarily long cycle of direct dependencies.

**Interdependency.** An *interdependency* is a two-way relationship where two entities are mutually dependent on each other. If the dependency relationship is *bi-directional* between two entities, then it is referred to as an *interdependency* [14]. Thus, if the dependency relation $\langle D\rangle$ is such that it is also symmetric, that is $A\,\langle D\rangle\,B \iff B\,\langle D\rangle\,A$, then it is an interdependency relation. An interdependency between two entities can lead to an escalation of failures of the entities as well as of other entities that depend on them due to the feedback caused by the interdependency, which may reinforce the interdependency over time. Fig. 3(a) and Fig. 3(b) demonstrate respectively direct and indirect interdependencies between entities $A$ and $B$.



(a) Direct interdependency

(b) Indirect interdependency

**Fig. 3.** Interdependencies between $A$ and $B$

**Co-dependency.** A *co-dependency* exists between two or more nodes when they mutually depend on a third node, whose failure can lead to a simultaneous failure of the co-dependent nodes. *Common-cause failures*, which occur when two or more infrastructure network components are disrupted by the same event, are due to the existence of co-dependencies on the common-cause. Consider the geo-spatial example above. Here it is the fact that there exists a *co-dependency* of the two entities on their geo-spatial location, which links the failure of one to the other. This is a special case of our *co-dependency* analysis, which considers the dependency induced between two (or more) entities when another entity that they depend on can cause simultaneous failure in both. Other examples include simultaneous failure of two servers that occurs due to the exploit of a common vulnerability, failure of an entire sensor network due to the malicious compromise of the control system used to manage the sensors, or if an organisation chooses, for redundancy reasons, two upstream suppliers, which themselves are both dependent on a further upstream supplier, a failure of this supplier could still propagate down and disrupt the supply system of the organisation seeking to be redundant in its supply. So complex supply chains can negate the perceived risk reduction gained from using multiple suppliers for a single input if they are actually all co-dependent further upstream on the single-point-of-failure. Similarly, if two information providers rely on a common upstream source, then misinformation in the common source may contaminate the information obtained from both providers.

Formally, we define *co-dependency* as a binary relation $\langle \mathit{CoD}\rangle \subseteq \mathcal{P}(\mathcal{N}) \times \mathcal{P}(\mathcal{N})$ between sets of entities (graph nodes). So, for example, $\{B, C\}\ \langle \mathit{CoD}\rangle\ \{A\}$ holds for the dependency graphs of Fig. 4, where for any set $X$ and $Y$ of entities, $X\ \langle \mathit{CoD}\rangle\ Y$ means that the entities within $X$ are co-dependent on the entities within $Y$. Furthermore, the domain of the co-dependency relation is closed under subset inclusion, so that for any node $A$ and set of nodes $X$, such that $Z \subseteq X$ is a non-empty subset, we have that $X\ \langle \mathit{CoD}\rangle\ \{A\} \Rightarrow Z\ \langle \mathit{CoD}\rangle\ \{A\}$. Since the failure of any entity in the co-domain of a co-dependency relation may lead to the simultaneous failure of all entities in the domain of the co-dependency relation, we may thus view co-dependency as generalising direct and indirect dependencies: it follows that these dependencies can be specified by the

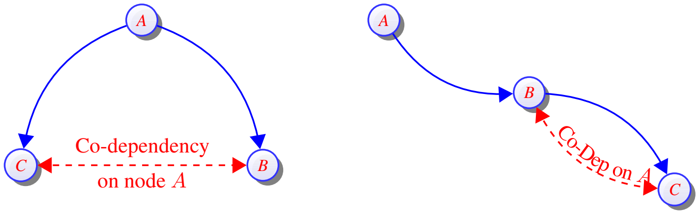


**Fig. 4.** Co-dependency of $B$ and $C$ induced by their mutual dependencies on node $A$

co-dependency relation. For example, the dependency relations $C\ \langle D\rangle\ B\ \langle D\rangle A\ \Rightarrow \{B\}\ \langle \mathit{CoD}\rangle\, \{A\}\ \wedge \{C\}\ \langle \mathit{CoD}\rangle\ \{A\}$; but the converse is not necessarily true, as we cannot deduce any relationship between $B$ and $C$ from the co-dependencies stated. However, in general, for any set $X$ of entities such that $X\ \langle \mathit{CoD}\rangle\ \{A\}$ holds, we have that for all $B \in X, \{B\}\ \langle \mathit{CoD}\rangle\ \{A\}$ by the subset closure property of the domain of the co-dependency relation and $B\ \langle D\rangle\ A$ by the transitivity of the dependency relation.

**Redundant Dependency.** While *co-dependency*, on one hand, captures the notion of the simultaneous failure in two or more entities induced by the failure of a common dependency source entity, *redundant dependency*, on the other hand, exists between a target entity and two or more other source entities, whereby all the source entity must fail before the target entity is sufficiently impacted (see Fig. 5). Redundant dependency, as its name suggests, occurs in redundant systems, where redundancy prevents a total failure of the system as long as at least one of the redundant entities is still functioning. While redundancy may be planned as a resilience strategy, it is also possible that redundancy exists within a given infrastructure which are unplanned. The discovery of such unplanned redundant-dependencies is still useful, as it not only gives further assurances, but it may give the system manager an opportunity to redeploy the redundancy to areas within the system where that redundancy is likely to provide the greatest benefits.

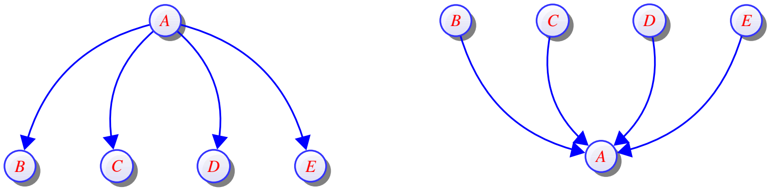


**Fig. 5.** Co-dependency of $B, C, D, E$ on $A$ Vs Redundant dependency of $A$ on $B, C, D$ and $E$

## 3 The Modelling and Analytical Framework

Our approach to creating the modelling and analytical framework is to develop a language for expressing the system being assessed which can then be either directly parsed by our own bespoke tools, or transposed into other modelling languages in order to exploit existing tool sets. We describe here the *SATURN Dependency Modelling Language*

(***SDML***), a *domain-specific language* (DSL) used to describe the architecture, properties, and configuration of components within the infrastructure network.

We give an example in Section 3.2 of how ***SDML*** might be used to model a system and how the model can be assessed using graph-theoretic causal influences backed up by a formal axiomatic model to establish the presence of dependencies and reason about network risks. Note that the application of graph theory or relational models to dependency networks, is not in-itself novel (see, for example, [15,6,3,9]). What is new is our application, specifically, a formal modelling and analysis methodology for system abstraction, which is guided by our conceptual model and a formal Ontology [1,5]. We have also extended the analytical framework to support automated *What If?* analyses as well as the capability to reason about properties and states of the CI [5].

### 3.1 *SDML* Modelling Constructs

This section presents key ***SDML*** constructs used in this paper.

The ***providesService*** construct enables the specification of service provided by an entity and, optionally, the concrete entities that the services are provided to. The ***requires*** construct is used in our model to describe the configurations under which a given entity is operational to a given level of expectation. The semantics that we ascribe to the ***requires*** specification is such that the entity with the ***requires*** construct fails if its requirements are not met. Through the ***isA*** construct, we can specify type hierarchies, which provide useful structuring abstractions for entities within our model. The failure of a node may come from the influence of externalities that are not required for its operation. For example, a power substation may fail due to flooding, or a virus might infect a computer system – causing it to fail. These types of causal factors are introduced into our model via the ***impacts*** constructor.

### 3.2 A Sample Model

The recent unfortunate earthquake on March 11 2011 in Japan highlights some of the dependencies that exist within the critical infrastructure and the society at large, triggered by the natural disaster. While Japan is quite used to and prepared for earthquakes, certain subtle dependencies had unforeseen consequences that exacerbated the problems and made recovery more difficult. Consider, for example, the Fukushima Daiichi nuclear plant, Dai-1, which generates electricity. The Dai-1 nuclear complex has six boiling-water reactors, three of which were down for routine maintenance at the time. The energy released from the nuclear fission reaction is used to boil water into steam, which in turn drives turbines used to generate electricity. Water is also used to keep the fuel-rods cool, preventing overheating that could result in a meltdown while also preventing radiations from leaking to the atmosphere. After the earthquake, although the Daiichi reactors were not badly damaged and their emergency shutdown was successful, circulating cold water was still needed to keep the fuel-rods cool. But the Tsunami following the earthquake had taken out the power generators that drove the water cooling system. Even though backup generators and battery powered backup systems were rushed in, these neither kept the temperatures down nor maintained the required water level. This led to a series of explosions damaging the reactors. Also, the subsequent spikes in radiation levels meant that the Daiichi workers and external support staff could not prevent the reactors deteriorating further. This highlights an interdependency between the safety of the workers and the restoration of the nuclear power plant, specifically because the radiation impacted the safety of the workers, which in turn meant that they could not effectively work to keep the reactor safe resulting in a potential for further deterioration, leading to more radiation.

To illustrate the modelling and analysis process, we have developed a hypothetical model of the Daiichi nuclear power plant, showing direct dependencies from which more detailed dependency relationships, such as co-dependencies, are automatically derived through tool support. The ***SDML*** model of Fig. 6 is a fictitious (and simplified) configuration of the Daiichi plant. It consists of three nuclear reactors $R1, R2$, and $R3$, and being hot-water reactors, we specify through their supertype, *Reactor*, that each reactor "***requires*** *Water*". Since a *Reactor* can generate harmful radiation if it breaks down, we add the clause "*Reactor* ***impacts*** *Radiation*" to say so. Hence, $R1, R2$ and $R3$ can all "impact" *Radiation* in the event of their failure. Similarly, *Radiation* ***impacts*** *People*, since high doses are harmful. The water pumps $P1$ and $P2$ supply water to reactor pairs $R1, R2$ and $R2, R3$ respectively. The repair and control of the three reactors are carried out by people as declared in the model. Reactor $R3$ provides electricity to the water pumps, and generators $G1$ and $G2$ act as backups to the electricity generated by $R3$, which is the primary source of power to the water pumps. However, a tsunami will disable the generators, and earthquakes can result in a tsunami: specified as *Tsunami* ***impacts*** *G1, G2* and *Earthquake* ***impacts*** *Tsunami*. The resulting dependency graph is shown in Fig. 7, where the nodes $E$, $T$, *Rad*, *Peo* respectively stand for the *Earthquake*, *Tsunami*, and *Radiation* events, and *People*.

*Reactor* ***requires*** *Water* ; *Reactor* ***impacts*** *Radiation* ; *Radiation* ***impacts*** *People* ; *R1, R2, R3* ***isA*** *Reactor* ;<br>
*Water_Pump* ***requires*** *Electricity* ; *P1, P2* ***isA*** *Water_Pump* ; *G1, G2* ***isA****Generator* ;<br>
*Tsunami* ***impacts*** *G1, G2* ; *P1* ***providesService*** *Water* ***to*** *R1, R2* ; *P2* ***providesService*** *Water* ***to*** *R2, R3* ;<br>
*People* ***providesService*** *Control, Repair* ***to*** *R1, R2, R3* ; *R3* ***providesService*** *Electricity* ***to*** *P1, P2* ;<br>
*G2* ***providesService*** *Electricity* ***to*** *P2* ; *G1* ***providesService*** *Electricity* ***to*** *P1* ; *Earthquake* ***impacts*** *Tsunami*

**Fig. 6.** A Hypothetical Model of the Daiichi Nuclear Power Plant in ***SDML***

***SDML*** models are relatively easy to develop by humans because the emphasis is on direct causal influences between the entities, which humans are very good at. Moreover, the language is deliberately close to natural language for ease of use, although it is backed up by a formal model (partly introduced in Section 4) and a formal Ontology. The problem is that, even in very small models, humans are not very good at spotting very subtle dependencies and causal relationships, which makes the use of automated reasoning tools imperative. For example, through our tool, we discover that an earthquake can lead to a collapse of the whole system, leaving it in a state that is difficult to fix because of cyclic interdependencies. Specifically, this is because the whole system is *co-dependent* on the *Earthquake* event, and because of the interdependency cycle whereby the reactors impact the radiation level, which in turn impacts the people who are responsible for repairing the reactors, a fix is difficult. The knowledge of such interdependency will be helpful during risk assessment of the facility, and will be discovered by playing what-if games, whereby entities are made to randomly fail. More

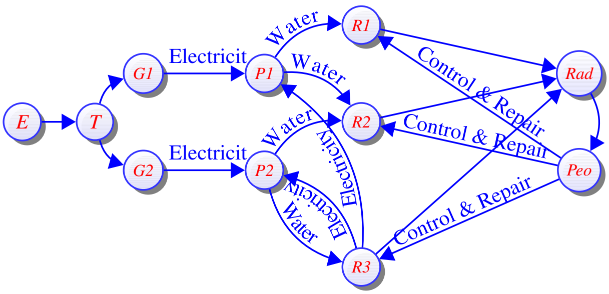


**Fig. 7.** Dependency Graph of the Hypothetical Daiichi Model

specifically, the problem results from a combination of the procedural shutdown of the reactors, coupled with the backup generators being taken out by the Tsunami. To see why, consider that the earthquake prompted the shutdown of the reactors, which means that the electricity produced by $R3$ fails (denoted by $\mathit{fails}(\mathit{R3.Electricity})$). However, a possible causal chain is that $\mathit{fails}(\mathit{Earthquake}) \Rightarrow \mathit{fails}(\mathit{Tsunami}) \Rightarrow \mathit{fails}(\mathit{G2}) \Rightarrow \mathit{fails}(\mathit{G2.Electricity})$. The last implication is based on an axiom where for any entity $X$ and service $Y$ that it provides, the failure of $X$ implies the failure of its services: $\mathit{fails}(X) \Rightarrow \mathit{fails}(X.Y)$. But we know that $\mathit{fails}(\mathit{G2.Electricity}) \wedge \mathit{fails}(\mathit{R3.Electricity}) \Rightarrow \mathit{fails}(\mathit{P2})$ because *P2* requires *Electricity*. Similarly, we have that $\mathit{fails}(\mathit{G1.Electricity}) \wedge \mathit{fails}(\mathit{R3.Electricity}) \Rightarrow \mathit{fails}(\mathit{P1})$. Thus, the whole water pump system fails, leading to the failure of the reactors, impacting radiation and in turn people. This is discovered by our analytic tools in a what-if scenario where electricity generated by *R3* fails during an earthquake event.

## 4 The Formal Model

We now turn our attention to the formal model for the *SATURN* dependency analysis framework, which provides a basis for the reasoning techniques that we have implemented in tools for reasoning about infrastructure dependency. For the graph-theoretic analysis employed within the *SATURN* framework, we denote all entities of interest as nodes in the dependency graph. The set of all nodes is represented by the set $\mathcal{N}$. The dependency relationship between entities in a model are represented by directed *edges* (or *arcs*, or *arrows*) within the dependency graph. When there is an arrow going from a node $A$ to a node $B$, then there exists a dependency relationship between the two. More specifically, $B$ is *dependent on* $A$, which is denoted by $B \langle D \rangle A$. The dependency relation may be annotated to specify what sort of relationship it is. For example, it might be the case that $B$ depends on $A$ for water supply, say. This is formalised as $\models_D B \langle D \rangle A : \{\mathit{water}\}$. This asserts that $B$ depends on $A$ for *water*. More generally, for any two entities $A$ and $B$, and a set of entities $X$ (usually, services), the entailment $\models_D B \langle D \rangle A : X$ means that $B$ depends on $A$ for all the "services" $C \in X$.

We define a polymorphic *failure* predicate that can be used to assert the failure of entities in a model, such as the failure of nodes and edges. To assert the failure of an entity $A$ (nodes and services) in the model, we simply specify $\mathit{fails}(A)$. The *failure* operation are used in our deduction system for reasoning about failure dependencies, for example, $\mathit{fails}(A) \Rightarrow \mathit{fails}(B)$ specifies that the failure $A$ leads to the failure of $B$.

### 4.1 Dependency Rules

We formalise in this section important properties of the dependency definitions and model. It forms the basis of our approach to deriving the dependency relationships between entities as specified in an ***SDML*** model.

$$A, B, C \in \mathcal{N} \qquad X, Y, Z \subseteq \mathcal{N}$$

$$\text{[Serv-DR]}\ \frac{\forall j \in J.\quad A\,\textbf{\textit{providesService}}\ \mathit{serv}_j\ \textbf{to}\ B}{\models_D B\,\langle D\rangle\, A : \{\mathit{serv}_j \mid j \in J\}}\ \mathit{serv}_j \in \mathcal{S} \qquad \text{[DepFail-DR]}\ \frac{\mathit{fails}(A) \Rightarrow \mathit{fails}(B)}{B\,\langle D\rangle\, A}$$

$$\text{[Trans-DR]}\ \frac{A\,\langle D\rangle\, B \quad B\,\langle D\rangle\, C}{A\,\langle D\rangle\, C} \qquad \text{[ServDepWeak-DR]}\ \frac{\models_D B\,\langle D\rangle A : X}{B\,\langle D\rangle A}$$

$$\text{[CoDFail-DR]}\ \frac{\forall B \in X \quad \mathit{fails}(A) \Rightarrow \mathit{fails}(B)}{X\,\langle \mathit{CoD}\rangle\,\{A\}} \qquad \text{[CoDWeak-DR]}\ \frac{X\,\langle \mathit{CoD}\rangle\,\{A\}}{Y\,\langle \mathit{CoD}\rangle\,\{A\}}\ Y \subseteq X$$

$$\text{[CoDJoin-A-DR]}\ \frac{\forall j \in J.\quad X_j\,\langle \mathit{CoD}\rangle\,\{A_j\}}{\bigcap_{j\in J} X_j\,\langle \mathit{CoD}\rangle\,\{A_j \mid j \in J\}} \qquad \text{[CoDJoin-B-DR]}\ \frac{\forall j \in J.\quad \{A_j\}\,\langle \mathit{CoD}\rangle\, X_j}{\{A_j \mid j \in J\}\,\langle \mathit{CoD}\rangle\,\bigcap_{j\in J} X_j}$$

$$\text{[CoDSingle-DR]}\ \frac{X\,\langle \mathit{CoD}\rangle\, Y}{X\,\langle \mathit{CoD}\rangle\,\{A\}}\ A \in Y \qquad \text{[CoDSetWeak-DR]}\ \frac{X \subseteq X' \quad Y \subseteq Y' \quad X'\,\langle \mathit{CoD}\rangle\, Y'}{X\,\langle \mathit{CoD}\rangle\, Y}$$

$$\text{[MaxCodep-DR]}\ \frac{X = \{B \in \mathcal{N} \mid B\,\langle D\rangle\, A\}}{\models_{max} X\langle \mathit{CoD}\rangle\,\{A\}} \qquad \text{[CoDtoDep-DR]}\ \frac{X\,\langle \mathit{CoD}\rangle\,\{A\}}{B\,\langle D\rangle\, A}\ B \in X$$

$$\text{[ServFail-DR]}\frac{A\,\textbf{\textit{providesService}}\ \mathit{serv}}{\mathit{fails}(A) \Rightarrow \mathit{fails}(A.\mathit{serv})}\ \mathit{serv} \in \mathcal{S}$$

**Fig. 8.** A Selection of Dependency Rules

A description of the dependency rules of Fig. 8 is as follows. The rule [Serv-DR], which is used to obtain *service dependency*, states that if entity $A$ provides a set of services to the entity $B$, then the entity $B$ is *service-dependent* on $A$ based on those services. In the graphical model, each service $\mathit{serv}_j$ represents an arrow that goes from $A$ to $B$ indicating a dependency of $B$ on $A$ for that service. The rule [DepFail-DR] says that if the failure of $A$ leads to the failure of $B$ then $B$ is (*failure*)*-dependent* on $A$. The rule [Serv-DR] asserts the transitivity of the dependency rule: if $A$ depends on $B$ and $B$ depends on $C$, then $A$ (indirectly) depends on $C$. The rule [ServDepWeak-DR] states that we can weaken the statement that $B$ depends on $A$ for services in $X$ to the assertion that $B$ simply depends on $A$. The rule [CoDFail-DR] shows how to construct co-dependency from failure properties: if all the entities $B$ in the set $X$ of entities fail simultaneously on the failure of the entity $A$, then the set $X$ of entities is co-dependent on $A$. We can weaken a co-dependency assertion through the rule [CoDWeak-DR] by saying that if the set $X$ of entities is co-dependent on entity $A$, then a smaller set $Y \subseteq X$ is also co-dependent on $A$. The rule [CoDJoin-A-DR] lets us combine co-dependencies by moving from co-dependency on a single entity to co-dependency on a set of entities. Since a set $X$ of entities is co-dependent on a set $Y$ of entities if the failure of any node in $Y$ leads to the failure of all entities in $X$, then the domain of the co-dependency relation is intersected while we union the entities in the co-domain. For example if $\{B1, B_2, B_3\}\ \langle \mathit{CoD} \rangle\ \{A_1\}$ and $\{B_2, B_3, B_4\}\ \langle \mathit{CoD} \rangle\ \{A_2\}$ then $\{B_2, B_3\}\ \langle \mathit{CoD} \rangle\ \{A_1, A_2\}$, that is, the failure of $A_1$ or $A_2$ leads to the failure of $B_2$ and $B_3$. Similarly, the rule [CoDJoin-B-DR] allows us to strengthen the domain of the co-dependency relation by taking the intersection of the co-domain.

The rule [CoDSingle-DR] asserts that if the set of entities $X$ is co-dependent on the set of entities $Y$, then it is also the case that the set of entities $X$ is co-dependent on any entity $A \in Y$. The rule [CoDtoDep-DR] asserts that a co-dependency also implies a dependency: if a set of entities $X$ is co-dependent on an entity $A$, then every entity $B \in X$ is dependent (either directly or indirectly – by transitivity of $\langle D \rangle$) on $A$. The [CoDSetWeak-DR] allows us to weaken the domain and co-domain of a co-dependency relation by replacing both with subsets. The rule [MaxCodep-DR] allows us to find the largest set of entities that are co-dependent on a given entity $A$ within a model. Finally, axiom [ServFail-DR] states that if an entity fails, its services also fail.

The objective of these foundational rules is to provide generic default reasoning about dependencies. The rules, as presented here, do not take into account buffering or other more dynamic properties that CI entities may exhibit. However, through ***SDML*** language extensions [5], we have shown how to add and specify dynamic behavioural properties and states of infrastructure entities, such as failure categories and the dynamics of failure propagation. These extensions are built on top of the foundational rules to provide more fine-grained domain-specific reasoning capability.

## 5 Future Work

We plan to extend the analytical framework to incorporate a more complex capture of state (partly achieved in [5]) enabling us to incorporate the use of risk controls, their performance or dampening effects, the capability of malicious attackers (within and external to organisations) and the impact of other externalities such as regulatory pressures and financial markets. This will be achieved through a natural extension of the dependency modelling proposed here into the process algebra CSP, and the deployment of the accompanying model checking tool FDR (to enable automatic analytical support). This should also allow us to handle the additional complications in the analysis of redundant systems, where upstream co-dependencies may negate the resilience that redundancy was intended to bring.

## References

1. Adetoye, A., Creese, S., Goldsmith, M., Hopkins, P.: A modelling approach for interdependency in digital systems-of-systems security - extended abstract. In: Xenakis, C., Wolthusen, S. (eds.) CRITIS 2010. LNCS, vol. 6712, pp. 153–156. Springer, Heidelberg (2011)
2. Avižienis, A., Laprie, J.-C., Randell, B., Landwehr, C.: Basic concepts and taxonomy of dependable and secure computing. IEEE Trans. Dependable Secur. Comput. 1(1), 11–33 (2004)

3. Balakrishnan, A., Magnanti, T.L., Mirchandani, P.: Connectivity-splitting models for survivable network design. Networks 43(1), 10–27 (2004)
4. Boss, M., Elsinger, H., Summer, Thurner: Network topology of the interbank market. Quantitative Finance 4, 677–684 (2004)
5. Creese, S., Goldsmith, M.H., Adetoye, A.O.: A logical high-level framework for critical infrastructure resilience and risk assessment. In: The 3rd International Workshop on Cyberspace Safety and Security (CSS 2011), Milan, Italy (September 2011) (to appear)
6. Dudenhoeffer, D.D., Permann, M.R., Manic, M.: CIMS: a framework for infrastructure interdependency modeling and analysis. In: Felipe Perrone, L., Lawson, B., Liu, J., Wieland, F.P. (eds.) Proceedings of the Winter Simulation Conference, WSC 2006, Monterey, California, USA, December 3-6, pp. 478–485. WSC (2006)
7. Fong, T., Fung, L., Lam, L., Ip-wing, Y.: Measuring the interdependence of banks in Hong Kong. Working Papers 0919, Hong Kong Monetary Authority (2009)
8. Müller, J.: Interbank Credit Lines as a Channel of Contagion. Journal of Financial Services Research 29, 37–60 (2006)
9. Neville, J., Jensen, D., Chickering, M.: Relational dependency networks. Journal of Machine Learning Research 8, 2007 (2007)
10. Nier, E., Yang, J., Yorulmazer, T., Alentorn, A.: Network models and financial stability. Journal of Economic Dynamics and Control 31(6), 2033–2060 (2007); Tenth Workshop on Economic Heterogeneous Interacting Agents - WEHIA 2005
11. Nieuwenhuijs, A., Luiijf, E., Klaver, M.: Modeling Dependencies in Critical Infrastructures. In: Nieuwenhuijs, A., Luiijf, E., Klaver, M. (eds.) Critical Infrastructure Protection II. IFIP, vol. 290, pp. 205–213. Springer, Boston (2008)
12. Pederson, P., Dudenhoeffer, D., Hartley, S., Permann, M.: Critical infrastructure interdependency modeling: A survey of U.S. and international research. Technical Report INL/EXT-06-11464, Idaho National Laboratory, Idaho Falls, Idaho 83415 (August 2006)
13. Peerenboom, J.P., Fisher, R.E.: Analyzing cross-sector interdependencies. In: 40th Annual Hawaii International Conference on System Sciences, HICSS 2007, p. 112 (January 2007)
14. Rinaldi, S.M., Peerenboom, J.P., Kelly, T.K.: Identifying, understanding, and analyzing critical infrastructure interdependencies. IEEE Control Systems Magazine 21, 11–25 (2001)
15. Svendsen, N.K., Wolthusen, S.D.: Multigraph dependency models for heterogeneous infrastructures. In: Goetz, E., Shenoi, S. (eds.) Critical Infrastructure Protection. IFIP, vol. 253, pp. 337–350. Springer, Boston (2007)

# Assurance and Trust Indicators to Evaluate Accuracy of On-line Risk in Critical Infrastructures

Thomas Schaberreiter[1], Filipe Caldeira[2], Jocelyn Aubert[1],
Edmundo Monteiro[2], Djamel Khadraoui[1], and Paulo Simões[2]

[1] Centre de Recherche Public Henri Tudor, Service Science & Innovation (SSI),
29, Avenue John F. Kennedy, L-1855 Luxembourg, Luxembourg
{firstname.lastname}@tudor.lu

[2] CISUC - DEI, University of Coimbra
Coimbra, 3030-290, Portugal
{fmanuel,psimoes,edmundo}@dei.uc.pt

[3] Polytechnic Institute of Viseu
Viseu, 3504-510, Portugal

**Abstract.** Critical infrastructure (CI) services are consumed by the society constantly and we expect them to be available 24 hours a day. A common definition is that CIs are so vital to our society that a disruption or destruction would have a severe impact on the social well-being and the economy on national and international levels.

CIs can be mutually dependent on each other and a failure in one infrastructure can cascade to another (inter)dependent infrastructure and cause service disruptions. Methods to better assess and monitor CIs and their (inter)dependencies at run-time in order to be able to evaluate possible risks have to be developed. Furthermore, methods to ensure the validity of evaluated risk have to be investigated.

In this work, we build on existing work of CI security modelling, a CI model that allows modelling the risks of CI services at run-time. We conduct a study of indicators allowing to evaluate the correctness of calculated service risk, taking into account various sources contributing to this evaluation. Trust-based indicators are introduced to capture the dynamically changing behaviour of a system.

**Keywords:** Critical infrastructures, ICT security, Trust and reputation management.

## 1 Introduction

Critical infrastructures (CI) provide services that build the centre of our society and economy. For example, telecommunication infrastructures allow us to communicate with people and businesses at remote locations, transport and air traffic infrastructure allow us to travel to places far away for free-time or business activities. The electricity infrastructure enables a variety of services and applications that we take for granted. Furthermore, CIs depend on each other. A good

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 30–41, 2013.

example is the electricity infrastructure that is a requirement for all other CIs, since nowadays almost everything relies on a constant supply of energy. A failure in one CI can cascade to other CIs and cause service disruptions.

To operate complex systems like CIs can be problematic and CI providers put substantial effort into keeping CIs running and reduce risks of any kind, for example the risk of failure, the risk of intrusion or the risk of incorrect operation. In this context, the idea of *CI security modelling* was introduced. The core of the idea is to be able to estimate the risk in *CI services* in real-time. The risk that is taken into account is the risk of a breach of confidentiality, the risk of a breach of integrity and the risk of a degradation of availability (CIA) of a service. To be able to estimate this risk, evidence is gathered from measurements taken from CI components (*base measurements*). One important aspect of the CI security model is that it allows taking into account the risk of *dependent CI services* in the risk calculation of a CI service.

In this publication we try to answer one key question that was not sufficiently answered in the originally proposed CI security model: "How can the risk estimated for a CI service be validated?". The accuracy of risk estimated for a CI service relies on the correctness of the base measurements. The correctness is defined by the accuracy of the base measurements as well as their dynamic behaviour during operation. For example, due to changing environmental conditions, the accuracy of base measurements and therefore the accuracy of estimated CI service risk can change.

The contribution of this work is in presenting indicators capable of evaluating the correctness of CI service risk based on the evaluation of the accuracy of base measurements (*base measurement assurance*) and the evaluation of dynamic behaviour of measurements by using a trust-based approach to capture the dynamically changing accuracies (*risk alert trust*) as well as the dynamically changing behaviour (*behaviour trust*). Furthermore, we evaluate the applicability of the indicators by giving an example using simulated data.

The remainder of the paper is organised as follows: Section 2 discusses related work, Section 3 introduces the CI security modelling approach. Section 4 specifies the identified CI risk accuracy indicators and Section 5 gives an example to show their applicability. Finally, Section 6 concludes the work and gives an outlook to future work.

## 2 Related Work

The concept of CI security modelling relates to several research areas: CI modelling and simulation, CI (inter)dependency identification and risk estimation and calculation in CIs. Identifying the various kinds of dependencies among CIs has been subject to previous research. In [11] Rinaldi et al. provide an excellent overview on the dimensions where interdependencies can occur. Several publications propose CI models based on various different modelling techniques. For example, conceptual modelling is used in [13] by Sokolowski et al. to represent an abstract and simplified view of CIs. In [10] Panzieri et al. utilise the complex adaptive systems (CAS) approach for CI modelling. The model is derived by modelling the mutually dependent sub-systems of the infrastructure. Risk models for CIs were proposed by some authors. For example, in [8] Haslum et al. use continuous-time hidden Markov models for real-time risk calculation and estimation. In [4] Baiardi et al. propose a risk management strategy based on a hyper-graph model to detect complex attacks as well as to support risk mitigation. In [7] Haimes et al. propose an eight step risk ranking and filtering framework based on risk scenarios, using hierarchical holographic modelling. In general, previously published CI models and CI risk models vary greatly in their purpose and the extent to which they were implemented. The models are usually too high-level and therefore lack practical relevance or they are focused on a specific CI and therefore lack generality. The idea of the CI security modelling approach presented in [3,2] differs greatly from the models previously published. It tries to establish abstract models of CIs that can be compared with each other while maintaining generality by enabling it to be applied to all kinds of CI sectors.

This publication is concerned with investigating indicators that can be used to evaluate the correctness of calculated risk in the CI security model. Risk based security assurance was investigated by Savola et al. [12] and Ouedraogo et al. [9]. In their work, the goal is to gather evidence from a system to be able to categorise the systems security assurance into 5 classes (Class 1 meaning low confidence in the system, Class 5 meaning high confidence).

Trust and reputation is now a commonly used concept that is mostly focused on the development or refinement of trust models with application in areas like e-commerce web sites or, more generally, in situations where transactions between unknown systems or people occur [1]. The use of trust and reputation in the context of critical infrastructures has been proposed in [6,5] allowing the use of trust and reputation indicators in order to evaluate the correctness of dependency risk information exchanged among partner CIs and to be able to more accurately estimate the impact of received risks. In this publication, we adapt the concept of risk based security assurance and trust-based indicators to derive assurance indicators that can be used to reason about the accuracy of calculated CI service risk.

## 3 CI Security Modelling

CI security modelling was presented in [3,2]. As illustrated in Figure 1, the aim of the approach is to transform real-world infrastructure information into common abstract risk related information (in our case confidentiality, integrity and availability - CIA), to use this information to monitor the state of the infrastructure and to share it with (inter)dependent infrastructures in order to be able to evaluate the current infrastructure risk by taking into account the dependencies. The methodology, as illustrated in Figure 1, is composed of three steps: an *off-line risk assessment*, a *measurement aggregation* and an *on-line monitoring* step.

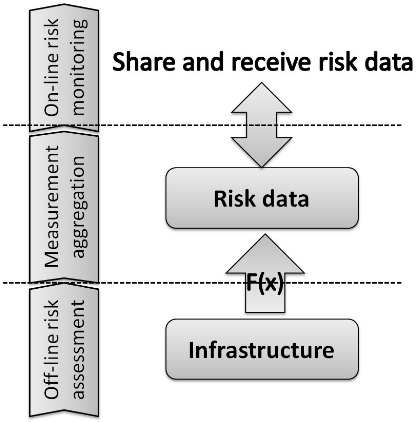


**Fig. 1.** CI security modelling methodology

### 3.1 Off-line Risk Assessment

The off-line risk assessment step is of special importance in the CI security model since risk estimation and monitoring can only be accurate if the structure of the systems is captured adequately. The off-line risk assessment allows analysing CIs and identify the entities that define the CI security model, namely the critical services, critical service (inter)dependencies and base measurements. The method is based on gathering information from various social (e.g. management, technical personal,...) and technical (documentation, manuals, vulnerability feeds,...) sources. Using those sources the critical services that define a CI can be captured. To address the problem of complexity of CIs, the model allows decomposing each identified service into more fine-grained sub-services. Each service entity can now be investigated separately to identify base measurements from CI components that define the state of the CI service and to identify dependencies to other internal or external CI services. After the risk assessment step, CIs are reduced to a directed graph containing only three entities: the critical services, the dependencies between critical services and the base measurements assigned to each critical service. To address the problem of the variety of CIs and the issue of continuous measurements that will represent different (physical) quantities and will represent different ranges, the base measurements need to be normalised. This can be done by estimating the measurement output in normal operation and defining ranges for allowed deviation from normal operation. For the CI security model, base measurement outputs are discretised to 5 levels (1 meaning normal output and 5 representing the maximum deviation from normal output). Another important step of the off-line risk assessment is to weight the importance of each dependency and each base measurement according to their importance to CIA of the service. This allows quantifying the different influence a base measurement or a dependency has for the risk calculation of a service.

### 3.2 Measurement Aggregation

In the measurement aggregation step, the service risk ($R_S$) is calculated by an averaged weighted sum of the normalised base measurements ($\mu$) and dependent service risk ($R_{Dep}$) using the weights ($\omega$) assigned in the off-line risk assessment step. Each risk indicator (CIA), representing a risk level between [1..5], is calculated according to Equation 1.

$$R_S = \left\lfloor \frac{\sum_{i=1}^{n} \mu_i * \omega_{\mu_i} + \sum_{i=1}^{m} R_{Dep_i} * \omega_{Dep_i}}{\sum_{i=1}^{n} \omega_{\mu_i} + \sum_{i=1}^{m} \omega_{Dep_i}} \right\rfloor \tag{1}$$

### 3.3 On-line Risk Monitoring

The on-line risk monitoring step is concerned with a constant distribution of changed risk values to dependent services and presenting the aggregated risk to an operator. It is important to present the risk in an easy and comprehensible way so that an operator can react quickly to changing risk and determine the root cause of the risk. Risk in this context can be seen as CI behaviour different from normal behaviour. This can be applied to virtually any situation where a CI service behaves different from normal operation. In our approach, this can be expressed numerically with the CIA indicators. The reduction to five levels of risk was chosen as a trade-off between the granularity of risk representation and the interpretability of risk information by an operator in a stress situation.

## 4 CI Service Risk Assurance Indicators

After setting the context of this work, this Section introduces the proposed CI service risk assurance indicators. As illustrated in Figure 2, three indicators will be described: the *base measurement assurance*, the *risk alert trust* and the *behaviour trust.*

### 4.1 Base Measurement Assurance

Assurance in this context can be seen as the confidence in the aggregated risk levels of a service. In other words, a service assurance level is an addition to the service risk level representing evidence that determines the accuracy of an aggregated risk level. This evidence is collected from the lowest entity in the CI security model, the base measurements. Each base measurement is associated with an assurance level. The determination of confidence in the correctness of each single base measurement is assumed to be done by domain experts during the off-line risk assessment step of the CI security model, supported by hard evidence of the correctness of a base measurement. For example, a domain expert will have a feeling for the correctness of a measurement taken from a system he knows. Let's assume this base measurement is a voltage level taken from a system of an electricity CI. A voltage meter that is used to take this measurement

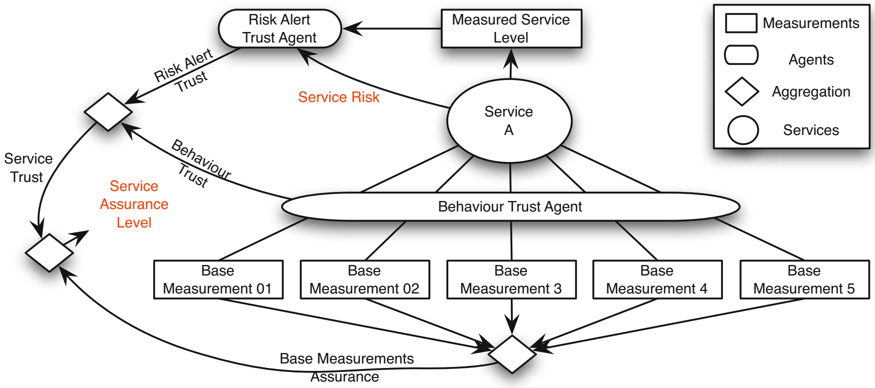


**Fig. 2.** System overview

will have an accuracy class that is determined by the manufacturer and can be used as hard evidence of the accuracy of the base measurement. Combining the subjective opinion of the expert with the hard evidence of the voltage meter accuracy class allows the domain expert to assign an assurance level to the base measurement.

The assurance level is represented by an integer number in the range [1..5]. This representation was chosen for the same reasons as the risk level scale, as a trade-off between accuracy and interpretability by an operator in a stress situation. Also, the assignment of assurance levels to base measurements by a domain expert is manageable in this way. The expert needs to have a decent amount of different choices and at the same time the choice needs to be limited to be able to have a meaningful comparison between the values.

Base measurement assurance levels are, like base measurement risk values, aggregated to represent the confidence in the accuracy of the service risk level. The chosen aggregation method for this work is, like for service risk level aggregation, an averaged weighted sum. The weight represents the importance a base measurement has for the risk calculation of a service and therefore the same weight as for service risk level aggregation can be used. Other than the service risk level which will change dynamically based on the current base measurement values, service assurance levels are assumed to be more static. They only change if the off-line risk assessment is repeated due to a change in the system or if faulty assumptions were detected in the risk assessment that need correction.

For illustration, a simple example is given. For a simple service five base measurements ($\mu$) were identified. An expert has low confidence in two base measurements, medium confidence in one base measurements and high confidence in two base measurements ($AL_\mu = \{1, 1, 3, 5, 5\}$). For illustrative reasons, the importance of the base measurements with low assurance level was estimated to be high and the importance of the base measurements with a high assurance level was estimated to to be low ($W_\mu = \{0.9, 0.9, 0.5, 0.1, 0.1\}$).

The aggregated service assurance level ($AL_S$) is calculated according to Equation 2.

$$AL_S = \left\lfloor \frac{\sum_{i=1}^{n} (AL_{\mu_i} * W_{\mu_i})}{\sum_{i=1}^{n} W_{\mu_i}} \right\rfloor = \lfloor 1.72 \rfloor = 2 \quad (2)$$

It can be seen that the aggregated service assurance level is relatively low, since the base measurements with the low confidence are assumed to be important to the service. On the other hand, high importance assumed for base measurements with a high assurance level will lead to a high aggregated service assurance level. For the above example, assuming $W_\mu = \{0.1, 0.1, 0.5, 0.9, 0.9\}$ will lead to an aggregated service assurance level $AL_S = 4$.

## 4.2 Risk Alert Trust

Risk alert trust is seen as the trust in the correctness of the calculated service risk. The idea behind the concept of risk alert trust is to compare the service risk ($Rl_t$) with the actually measured service level ($Ml_t$) as a measure of the quality-of-service. For example, if a power generation service has a high risk of availability degradation and the measured service level does not indicate that degradation, the trust in the accuracy of that service risk level should be lowered. The measured service level should be gathered using measurement equipment that must be independent of the service itself.

To be able to evaluate the trust in the service risk level, the first goal is to define an accuracy value for each calculated service risk. For this purpose, the concept of Risk Alert Event is introduced as one of the following situations: An event starts when one or both indicators ($Rl_t$ and $Ml_t$) are different from one (no risk) and 5 (maximum measured level) respectively. The event ends when both indicators return to their normal values.

The Risk Alerts Trust Agent is monitoring the service risk ($Rl_t$) and the current measured service level ($Ml_t$) in order to detect events. Both $Rl_t$ and $Ml_t$ belong to the [1..5] range. The accuracy of each event $A(Event_n)$ is defined as the average of all comparisons made during the event (value $T$), between the measured service level and the service risk level. Function $f(Ml_t, Rl_t)$ is a discrete function so a sample rate for the time factor is needed. This sample rate can be different for each service and will depend on the information available on the system. One small sample rate allows more realistic observations.

$$A(Event_n) = 100 - \left( \frac{\sum_{t=1}^{T} f(Ml_t, Rl_t)}{T} * 100 \right) \quad (3)$$

The calculation of $A(Event_n)$ is shown in Equation 3, where $f(Ml_t, Rl_t) = \mid \frac{Ml_t - Rl_t}{4} \mid^k, k \in R^+$. The value $k$ allows penalising the larger differences or the small differences. In this approach, the duration of an event is not considered as we are, for now, only focusing on the accuracy of each calculated service risk.

Using the same principles detailed in [5], the trust that we have in the service risk evaluated for service X is represented by $T_{(X)}$ and is calculated by the average of the accuracy of each past event for that particular service (Equation 4). The concept of ageing is used, applying a discount factor $D$, to give more weight onto recent events. The ageing factor should always depend on the context. In our model, the ageing factor needs to be defined individually for each service.

In this context, $T'_{(X)}$ can be computed for the N$th$ event as:

$$T'_{(X)} = \frac{(D * (N-1) * T_{(X)}) + A(Event_N)}{D * (N-1) + 1} \tag{4}$$

$D$ will be a value in the [0..1] interval and a small value of $D$ will raise the importance of the last events while a value of $D$ near 1 will provide less ageing for the oldest events.

A human factor reflecting the CI operator opinion and contribution to the trust calculation is also considered in trust evaluation (Equation 5).

$$T(final)_{(X,t)} = \alpha(T_{(X)}) + (1-\alpha)(TO_{(X)}) \tag{5}$$

The factor $\alpha$ is in the range [0..1] and assigned by the CI operator depending on the confidence he or she has in $(TO_{(X)})$. $T(final)_{(X,t)}$ represents the confidence in the service risk taking into account also the CI operator perspective. In order to understand how the risk alert trust indicators evolve over time, and to define a relation among them, a time value is associated with each $T(final)$.

### 4.3 Behaviour Trust

Behaviour trust refers to the trust in the correct behaviour of an entity (for example a service or a single component). The main goal is to understand and quantify the behaviour of each monitored entity considering what should be its normal behaviour. When a deviation from normal behaviour is detected, an event is triggered in order to incorporate this event in the behaviour trust indicator.

The events used to evaluate trust in service behaviour are all the monitored interactions among services (internal or external). For instance, the events can be Intrusion Detection System (IDS) alerts, failed connection attempts, attempts to read/write information without permission or the fact that some entity does not update risk information for a long period of time.

Another important source of information used to evaluate the behaviour trust is the base measurement entities. As it is simple to define the normal behaviour of those entities we can generate a security event when an abnormal behaviour is detected. For instance, although the normal temperature of an equipment can range from 0c to 70c, it is abnormal if the sensor reads 10 and one second later reads 60 and repeats this cycle. This situation can be seen as an abnormal behaviour of the sensor. Another example of an abnormal behaviour is the lack of information coming from a sensor. If we define as normal that one temperature sensor should at least inform the temperature every 30 seconds, it is possible to say that the behaviour is not normal if this is not happening. Also we should quantify that in terms of how the behaviour is different from the normal. Behaviour Information is normalised based on a security model that identifies relevant behaviour patterns. This system consists, basically, of tables mapping possible observed values and behaviour trust event values. For instance, it is possible to define that if a sensor takes $2,5$ minutes to send new information then the confidence in the behaviour of that sensor decays to 3 as shown in Table 1. This model acts as an adaptor between heterogeneous sources and the trust estimator algorithm. By employing these adaptors, it is possible to infer behaviour trust indicators as all entities are quantified and can be used in a common calculation.

**Table 1.** Normalisation Table Example

| Received information from sensor X / Minute | | |
|---|---|---|
| Trust Indicator Level | Description | Seconds since last value |
| 1 | No Failures | <= 30 |
| 2 | One Value received / Minute | > 30 and < 60 |
| 3 | One Value received / 2 Minutes | >= 60 and < 120 |
| 4 | One Value received / 3 Minutes | >= 120 and < 180 |
| 5 | No value received on the last 3 minutes | >= 180 |

In the behaviour context, we expect to receive alerts only when misbehaviour is detected, leading to a situation where almost only negative events are received and used in the evaluation. This situation would generate low behaviour trust over time. In order to evaluate a precise indicator, the factor time and the concept of inactivity were introduced. Time is divided into a set of time slots. Inactivity in one slot means that the entity behaviour indicators will have the maximum value for that period. If information is received during one slot, the value for that slot becomes the average of all values received during that slot [5].

For the time slot $s$, the trust in entity $E$ ($T'_{(B,s)}$) is calculated using Equation 6, where $D$ is the ageing factor, $T_{(E)}$ is the indicator evaluated for the slot $(s-1)$ and $Event_{(Slot\ s)}$ is the event value of the slot $s$.

$$T'_{(E,s)} = \frac{(D * (s-1) * T_{(E)}) + Event_{(Slot\ s)})}{D * (s-1) + 1} \tag{6}$$

Using Equation 7 the operator trust is included. The $\theta$ factor is assigned by the CI operator representing the confidence in the subjective trust $(TO_{(E,B)})$ that he or she has on the behaviour of $CI_B$ concerning entity $E$.

$$T(Final)_{(E)} = \theta(TO_{(E)}) + (1-\theta)(T_{(E)}) \ \ ,(0 < \theta < 1) \tag{7}$$

The proposed trust model also evaluates one indicator encompassing all monitored entities. Using a weight factor for each entity, the service behaviour reputation can be computed, considering the operator information. This indicator, $TBehaviour'_{(X,t)}$, represents the reputation of the behaviour of service $X$ at time $t$ [5]. In this work the behaviour reputation for a service is generally referred to as behaviour trust and includes the behaviour trust of all managed service entities.

## 5 Example

The proposed indicators have been validated using simulation and the outcome is promising as the simulation results are in-line with the main goals. In this Section, a small example is presented in order to demonstrate the proposed approach and to help understand the influence of trust indicators in the service assurance level.

The simple scenario in this simulation contains one single service with five base measurements (derived from sensors), similar to the scenario presented in Figure 2. For simplicity reasons we assume that this service does not depend on other services. We assume that the measurements are evaluated each minute and the simulation runs for 50 minutes. The contribution of each base measurement to the service has been defined as follows: $S1 - 10\%$ ; $S2 - 10\%$; $S3 - 30\%$; $S4 - 20\%$; $S5 - 10\%$.

The service risk $R_S$ is aggregated using the previously defined averaged weighted sum method[1]. In the simulation, the measured service level $M_S$ is aggregated using a similar setup of five independent sensors. We assume that the confidence in the correctness of all the base measurements is high and results in a base measurement assurance level of 5. The service trust is derived from the risk alert trust and the behaviour trust, using the following weights: 0.5 for the risk alert trust and 0.5 for the behaviour trust. The service assurance level is derived from the service trust and the base measurement assurance, using the following weights: 0.4 for the service assurance level and 0.6 for the service trust.

All indicators are defined using a scale of 1 to 5. For base measurements, trust indicators and assurance levels, 5 represents the best situation and 1 represent the worst. For the service risk level, 5 represents the highest risk and 1 represents the lowest risk.

In the first 20 minutes of the simulation, all sensors are reporting the value 5 leading to a risk level $R_S = 1$. Also the sensors used to aggregate the measured service level are reporting the value 5, leading to a service level $M_S = 5$. In this context, the risk alert trust and the behaviour trust have the maximum value.

On the next 10 events, sensors 1 and 3 of $R_S$ become unreliable but continue to report a value as presented in Figure 3(A). In this case they always report value 5 while sensors 1 and 3 of $M_S$ are generated using the following criteria: The difference between the sensor outputs of $R_S$ and $M_S$ is 1,2,3,4 respectively in 0% 5% 5% 90% of the cases. After t=30 minutes, the difference between the sensor outputs returns to 0. In Figure 3(B) the risk alert trust indicator displays this behaviour. The indicator drops when the values become unreliable and gradually starts to grow when the situation is back to normal.

The behaviour trust indicator is evaluated using information gathered on several entities that represent the behaviour of the system. In this case the behaviour of the sensors has been simulated. The normalised values presented in Table 1 are used. After t=25 minutes, two of the sensors stop sending periodic information.

[1] For simplicity reasons we only take one risk indicator into account for simulation. Whenever $R_S$ is mentioned, it represents either C,I or A risk indicator.

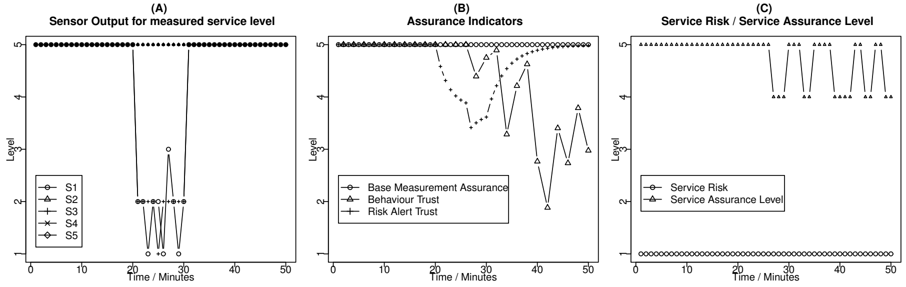


**Fig. 3.** Simulation example

For service risk aggregation, the last information received from the sensors will be used. Only the evaluation of the system behaviour through the behaviour trust reveals that the behaviour of the system is not as he should be, as can be seen in Figure 3(B). It is shown that the behaviour trust indicator changes rapidly, as a result of the unreliable updates of the sensors.

Figure 3(C) shows on the service level that, although the service risk is always 1, the service assurance indicates that the confidence in this risk estimation changes based on the dynamic behaviour observed by the risk alert trust indicator and the behaviour trust indicator. This behaviour could not have been captured by the static base measurement assurance indicator.

## 6 Conclusions and Future Work

In this work, a study of assurance indicators that are used to represent the confidence in the correctness of CI service risks was conducted. The work is based on a CI security model, that represents risk on the level of provided CI services and the risk of the services they depend on. Three assurance indicators were presented. The service assurance level, which represents the confidence in the correctness of the measurements that are used to aggregate the CI service risk. The risk alert trust indicator, a trust-based indicator evaluating the discrepancy between service risk and actually observed service level and the behaviour trust indicator evaluating the dynamic behaviour of the base measurements, for example abnormal behaviour due to sensor failure. An example was given to illustrate the validity of those indicators.

Future work will focus on the validation of the proposed approach. A realistic CI scenario that allows building a CI security model as well as access to dynamic data of CI behaviour (in normal operation as well as during security incidents) is necessary to be able to conduct a realistic validation. A cooperation with a CI provider to get access to such data is intended. In a next step, we want to introduce an approach based on Bayesian networks to use the service risk and the presented assurance indicators as evidence variables to reason about about the service risk. This will allow taking all the evidence gathered from the system into account and enhance the accuracy of the prediction of CI service risk.

## References

1. Artz, D., Gil, Y.: A survey of trust in computer science and the semantic web. In: Web Semantics: Science (January 2007)
2. Aubert, J., Schaberreiter, T., Incoul, C., Khadraoui, D.: Real-time security monitoring of interdependent services in critical infrastructures. Case study of a risk-based approach. In: 21st European Safety and Reliability Conference, ESREL 2010 (September 2010)
3. Aubert, J., Schaberreiter, T., Incoul, C., Khadraoui, D., Gateau, B.: Risk-based methodology for real-time security monitoring of interdependent services in critical infrastructures. In: International Conference on Availability, Reliability, and Security (ARES 2010), pp. 262–267 (February 2010)
4. Baiardi, F., Telmon, C., Sgandurra, D.: Hierarchical, Model-based Risk Management of Critical Infrastructures. In: The 18th European Safety and Reliability Conference, ESREL, vol. 94, pp. 1403–1415 (2009)
5. Caldeira, F., Monteiro, E., Simões, P.: Trust and reputation for information exchange in critical infrastructures. In: Xenakis, C., Wolthusen, S. (eds.) CRITIS 2010. LNCS, vol. 6712, pp. 140–152. Springer, Heidelberg (2011)
6. Caldeira, F., Monteiro, E., Simoes, P.: Trust and reputation management for critical infrastructure protection. Int. J. Electronic Security and Digital Forensics 3(3), 187–203 (2010)
7. Haimes, Y.Y., Kaplan, S., Lambert, J.H.: Risk filtering, ranking, and management framework using hierarchical holographic modeling. Risk Analysis 22(2) (2002)
8. Haslum, K., Arnes, A.: Multisensor real-time risk assessment using continuous-time hidden markov models. In: International Conference on Computational Intelligence and Security, vol. 2, pp. 1536–1540 (2006)
9. Ouedraogo, M., Khadraoui, D., De Remont, B., Dubois, E., Mouratidis, H.: Deployment of a security assurance monitoring framework for telecommunication service infrastructures on a voip service. In: New Technologies, Mobility and Security (NTMS 2008), pp. 1–5 (November 2008)
10. Panzieri, S., Setola, R., Ulivi, G.: An approach to model complex interdependent infrastructures. In: 16th IFAC World Congress (2005)
11. Rinaldi, S.M., Peerenboom, J.P., Kelly, T.K.: Identifying, understanding, and analyzing critical infrastructure interdependencies. IEEE Control Systems Magazine 21, 11–25 (2001)
12. Savola, R.M., Pentikainen, H., Ouedraogo, M.: Towards security effectiveness measurement utilizing risk-based security assurance. In: Information Security for South Africa (ISSA), pp. 1–8 (August 2010)
13. Sokolowski, J., Turnitsa, C., Diallo, S.: A conceptual modeling method for critical infrastructure modeling. In: 41st Annual Simulation Symposium (ANSS 2008), pp. 203–211 (April 2008)

# An Innovative Approach to Emergency Management in Large Infrastructures

Rüdiger Klein

Fraunhofer IAIS,
Schloss Birlinghoven, 53757 Sankt Augustin
ruediger.klein@iais.fraunhofer.de

**Abstract.** Critical Infrastructures as complex technical systems need sophisticated control systems enabling them to be managed effectively under normal, exceptional, and emergency conditions. The EU FP7 project EMILI was launched to develop innovative information technologies for emergency management in large complex Critical Infrastructures. CI are described as cyber-physical systems where physical behaviour and control are tightly integrated. Innovative information technologies as complex event processing and rule based reactions integrated with semantic modelling and physical simulation enable us to describe such cyber-physical systems with their behaviour and control adequately. These techniques provide a new problem oriented layer of description for situation assessment and reactions. They allows us to focus on the "what" of information processing leaving the "how" to an efficient information processing machinery. A comprehensive methodology of emergency modelling based on semantic models allows us to structure the many different kinds of information relevant in CI control. A Simulation and Training Environment SITE was designed which enables us to demonstrate our approach in three realistic but quite different CI use cases: an airport, a metro system, and a power grid.

**Keywords:** Emergency management, SCADA, innovative information technologies, situation assessment and reaction, complex event and action processing.

## 1 Introduction

Critical Infrastructures are complex systems already today, and they will become even more complex in the years ahead [1,2]. They are overwhelmingly important for the normal functioning of modern societies.

They are large, distributed, heterogeneous, mutually dependent, complex technical systems[1]. Control is an essential issue in all of them [3,4]. It's typically done by human operators supported by sophisticated ICT systems with varying degrees of automation. This control has to work adequately under normal,

[1] In some of them like public transport systems, airports, or logistic chains people are also directly involved.

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 42–57, 2013.

exceptional, and emergency conditions. Today, control systems in Critical Infrastructures are traditional large software systems – frequently heterogeneous and historically grown [5]. Their adaptation to new conditions, their extension, their validation and their maintenance are complex endeavours.

A *new generation* of information technologies is needed to support future more complex, more dependent, more heterogeneous, and last but not least more important Critical Infrastructures in all areas of modern societies. These CI will look different in the near future compared with what we have today. More sensors of different kinds will produce more information to be used for control. More information has to be exchanged between dependent CI. Operators need adequate support to manage this large amount of information – especially under exceptional and emergency conditions.

What can these new information technologies look like? How can they be used to support CI control more effectively and efficiently under changing conditions? Which methodology is needed to use them? And which tools? These are the key questions pursued in the EU FP7 research project EMILI [6].

The paper is structured as follows: in the following chapter we outline our methodology to Critical Infrastructures. We characterise CI as cyber-physical systems [7,8] with well defined physical behaviour *and* dedicated control under normal, exceptional, and emergency conditions. In Chapter 3 we describe how this methodology is used as basis for a new generation of information technologies in CI control. Events and actions and the context in which they occur are the main kinds of information to be managed by a CI control system. In EMILI we develop the Core Ontology as an expressive semantic framework for events, actions, and their context. Chapter 4 contains a brief summary of our three use cases (metro, airport, and power grid). In Chapter 5 We briefly describe the Simulation and Training Environment (SITE), that will be developed and implemented as an advanced simulation and training system for emergencies in large infrastructures. The description of SITE is followed by conclusions in Chapter 6.

## 2 Critical Infrastructures as Cyber-Physical Systems

Are there sufficient commonalities between such different Critical Infrastructures as public transportation, airports, or power grids? These commonalities could provide the starting point for a methodology of CI and especially of CI control. Of course, this methodology needs adaptations to the concrete kind of infrastructure. Based on this common CI methodology the information technologies needed for CI control can be specified.

Critical Infrastructures are physical systems which show behaviour according to physical and technical rules. On the other side, they are controlled systems which follow the rules defined in the underlying policy. Both aspects are tightly related to each other: a control which does not take the physical aspects (including environment) into account will not be effective. Physical behaviour without control is useless or dangerous. Both sides depend on each other. Consequently, we describe Critical Infrastructures as *cyber-physical systems* (CPS).

The notion of cyber-physical systems (CPS) was introduced recently [7,9]. It describes complex technical systems which are physical and computational systems. CPS research *as such* is still at its beginning though it can be based on a large amount of previous work in different areas (system theory, control theory, etc.). Recent progress in sensor and communication technology, the increased complexity of technical systems, and dramatically grown computational capabilities prepared the ground for a new generation of technical systems like modern cars with their sophisticated safety, control and driver assistance, new airplane generations, transport systems, industrial plants, and Critical Infrastructures. Previous generations of such systems were mainly *human* controlled systems with *some* control functionality provided by embedded micro-processors and/or traditional software systems (SCADA [10,5] etc.).

Cyber-physical systems in general and CI in particular are complex, heterogeneous *systems of systems.* Each single system or component may possess a more or less complex behaviour depending on the physical and technical mechanisms it is based on. As soon as they interact with other systems in a more complex system of systems these interactions need appropriate design specifications in order to be manageable and controllable. These interactions may be described in different forms, for instance as characteristic parameter curves or as parameter constraints. Frequently, the best way to describe these dependencies and interactions is through *services* which are provided and consumed between such parts in a system of systems.

### 2.1 States, Events, and Actions

For Critical Infrastructures as large networks of interacting systems and components *states* have shown as an appropriate abstraction from physical and technical details [11]. It is not so important what the precise voltage, temperature, or speed is – as long as these values are in a certain state, i.e., within certain limits for normal operational state, for light deviations from normal, or for exceptional and emergency states. Especially the operational modes ("on", "off", "broken", etc.) can be described in this way appropriately. States allow us to describe dependencies between systems and components and their characteristic parameters.

In addition to physical states we need control states as adequate abstraction for control actions. Voltage, temperature, or speed are physical states – instability, exceptional, or emergency are control states. They are based "somehow" on physical aspects, but in a complex way. They allow us to abstract from technical details and to focus on the essential aspects of a situation. Whatever the technical reason for a critical state is – there are control actions to be undertaken independently from the concrete physical side in this situation (stop operation, warn stakeholders, etc.). We can define rules in which control states depend on each other in different parts of a CI. An emergency situation in a part of a CI may cause an exceptional state in the rest of the system.

Closely related to states are events and actions. Events are happenings within the system resulting from state changes, or they happen in the system's environment and change the way it interacts with it including state changes of its components. Actions are the main element of CPS control. They deliberately change the state of a system or of its components.

### 2.2 Complex Events, Complex Actions, and Reactivity

Frequently, events are not isolated happenings but parts of more complex processes: a component failure in a power grid generates a cascade of event messages indicating breaker reactions, other components failing; a fire in a metro station triggers temperature and smoke sensors to generate event messages; etc. The set of events and the related system and component states allow us to describe the complex situations in which a system is. Complex events allow us to describe these relationships in an adequate and comfortable manner. This includes *temporal* aspects between events, states, and actions, spatial relations, functional aspects (within related parts of a complex system), etc.

In a similar way we can describe complex actions as collections of related elementary (or atomic) actions. Time, space, logical relations, and other aspects can be used to describe relationships between actions in a complex action.

A key element in cyber-physical systems is the relationship between situations and actions. In order to keep a complex cyber-physical system in an adequate state the control system has to react to situations according to certain policy rules. Such rules are needed for all situations: for normal operational mode, for different kinds of exceptions, and for emergency situations.

### 2.3 Communication

Communication is an essential element in each cyber-physical system. In a CPS, the physical and the control system communicate with each other through sensors and actuators. Sensors tell the control system in which state the various parts of the physical system are and what happens there, and actuators execute actions initiated by the control system and change the state of physical system components.

This communication is far from trivial. Whereas IT systems are highly reliable physical systems show a significantly higher rate of failure or disturbance. This may result from external/environmental influences, from depending systems, or from component failures in the system itself. Sensors or communication lines may be broken sending erroneous event messages or losing them at all. Similar problems may occur with actuators or their communication links. Consequently it is important for the control system to keep track of action execution including temporal aspects. Also other components in the physical system may be disturbed or broken, or the environment may be in an unintended state and influence system operation other than intended. Both may result in exceptional system behaviour and have to be managed adequately by the control system.

## 2.4 Physical Behaviour and Simulations

Of course, the physical behaviour is essential for a CPS. The functionality resulting from this behaviour is the reason for its existence. The control is needed to get the intended functionality out of the system's behaviour. Even under exceptional and emergency conditions certain rules have to be observed in order to avoid unnecessary damage.

Though this may change to some extend in the future, today physical systems frequently do not provide all information describing their state to the control system. By reasons of practicality or costs only some data are transmitted through appropriate sensors. This may be sufficient for normal operation. The control system knows when a metro train left the platform and "knows" that typically after 2 min. the train will arrive in the next station. It does not know precisely where the train is in between - it can just estimate. In a case of fire in a complex metro station the system knows where the fire is but it does not know how the smoke propagates through the platforms, staircases, etc. This is especially true for future evolutions of situations which are important for decision making. Both aspects – incomplete sensor information and forecast - are important reasons for simulations in CPS control. The control system has to maintain a model of the CPS with all relevant kinds of information: spatial attributes and topological relations, material properties, technical systems with their characteristics and dependencies, etc. In order to take physical behaviour adequately into account an integration of complex events and actions with simulations is needed.

## 2.5 Context and Situations

Critical Infrastructures are complex technical systems. Typically they consist of various sub-systems with special behaviours, dedicated roles and mutual dependencies. They provide and need services "from outside". The physical system and the control system communicate through sensors and actuators exchanging messages about atomic and complex events and actions.

It is extremely important for an adequate description of the behaviour and control of cyber-physical systems that these messages about events and actions carry all information needed to describe the state of the physical system to the control system. Temporal aspects between events and action, spatial relations, functional dependencies between components and sub-systems are necessary in order to enable the control system to maintain all relevant information for control. Complex events allow us to interpret event messages in more general patterns. The control system does not just receive single alarms from sensors – it detects the "rules" behind this set of messages and interprets them in their context.

For this purpose, the control system maintains a *model* of the physical system with all relevant kinds of information: types of components and sub-systems, their technical and spatial attributes and relationships; state dependencies between components and sub-systems; typical behaviours as sequences of states and transitions under normal and exceptional conditions.

### 2.6 Reactivity

There are two key issues in CPS control: to keep the system in a desired state where it can provide its functionality, and to react adequately to internal and external changes including failures, disturbances, etc. Both control issues need reactions to changing situations: in order to provide its functionality the system has to be adapted to changing functional requirements (power to be provided in a power grid, trains arriving or leaving in a metro station, etc.). If something "goes wrong" (a power line is broken, a fire in a metro tunnel) the system has to react to this change in order to avoid larger damages and restore normal operation as soon as possible. Typically, time is a critical factor for reactions. The situation may change rapidly, and the dynamics of the system demands fast reactions (where the time scale of course depends on the kind of the system and the concrete situation).

Consequently, we have to bring these two main aspects of control together:

- situation assessment and
- adequate reactions.

Situations can have many facets. It's not the single event which describes the situation but the event together with other events and the states the system and its parts are in. A fire signal sent by a sensor in a metro station is a serious event but how serious (and what reactions are adequate) depends on many other circumstances: is it a real event or a malfunction of the sensor; are other signals sent by related sensors, what is the number of passengers in that area, the position of trains, are there any obstacles or construction works around in the metro station, etc.

Following the comprehensive situation assessment we have to decide which reactions[2] are adequate in this situation. Reactions have to be assigned to situations in a generic way.

## 3 Innovative Information Technologies for Critical Infrastructures

In the previous chapter we outlined our general approach towards Critical Infrastructures as cyber-physical systems. We showed that states, events and actions are important to describe their behaviour and control. Complex events and actions allow us to interpret complex situations and to describe reactions to them adequately. The context in which events and action happen is described in a model of the physical system the control system maintains about all control relevant aspects.

[2] Frequently, control is done partially automatically (where the reactions are clear and time is short) and to some extend by experienced human operators which are supported by the control system. To find the right work share between the automatic system and the human operators is a great challenge for control system design. How EMILI addresses this issue will be described in a forthcoming publication.

The characteristics of information processing in Critical Infrastructures result in new requirements to information technologies:

- Complex events can be described generically as rules interpreting incoming event messages according to certain temporal, spatial, and context sensitive patterns.
- Complex actions are similarly described as patterns of simpler actions to be performed with their temporal, logical, and other relationships.
- Reactive rules allow us to assign atomic and complex actions to complex events and situations, where context of events and actions can be modelled with all necessary details.
- Where needed physical behaviour can be integrated into event and actions.

In the following we outline the new information technologies needed for Critical Infrastructure control.

### 3.1 Complex Events

Situation assessment is one of the main issues to control cyber-physical systems. For this purpose we have to deal with atomic events, complex events as patterns of (more) elementary events, and states and their changes.

Atomic events are the elementary happenings in the physical system. They are observed by sensors which generate messages sent to the control system. These events are sent by different types of sensors and indicate a certain state change in the physical system. They have a unique identity, they may have types like temperature event or switching event, they are related to sensors which have a position, a type, and other attributes, they may carry values like temperature or voltage, and they are of course characterised temporally by a begin and an end time[3]. We may assign confidentiality values, precision values and other kinds of information to them. The annotation is:

```
person-count{ area{ area1 }, value{ 23 } }
```

describing an event of type "`person-count`" which results for a certain area "`area1`" in a value "`23`". Of course, time is an essential attribute which is assigned automatically to every detected event[4].

Atomic events provide basic information. Complex events allow us to deduce "condensed" information by combining different kinds of more elementary information from different sources including logical and temporal relations into a useful "pattern". We can combine different person count events from different sensors at different times to see the change of a passenger flow pattern through

[3] We focus on occurrence times here though other temporal aspects like detection time, processing time, etc. may be relevant, too.

[4] It may be important to discriminate between different times: detection time, observation time, execution time. In the following we focus on detection time as the point in time where the sensor registered the event and gives it an according "time stamp".

a sensor network. We can deduce complex events by definition rules from (more) atomic events, from states, and from logical and temporal constraints:

```
DETECT
  avg-person-count{ area{ var A }, value{ avg( all var P) } }
ON
  and{
    event e: person-count{ area{ var A } },
    event f: person-count{ area{ var A }, value{ var P } }
  } where { {e,f} within 2 min, f before e }
END
```

with the meaning[5] that a complex event "avg-person-count" is detected whenever its "ingredient events" are – the single "person-count" events are detected and the logical conditions and the temporal conditions in its definition are fulfilled. A complex event starts with its first ingredient, and it ends with the end of its last ingredient.

This logic based approach to complex events gives us a rich and expressive way of event and situation modelling. A more comprehensive description of EMILI's event and action language DEAL can be found in [12].

### 3.2 Complex Actions

Actions can be atomic or complex. Atomic actions change directly the state of objects either in the physical system or in the control system. They are characterised by a starting time and an end time, they have a unique identifier, a state they are going to change, a value, and maybe some other information.

Atomic actions can be commands sent to actuators in the physical system:

```
switch_ventilation{ v23, on }
```

or simply control actions dedicated to stateful objects:

```
inform-station-personel{ area1 }
```

An important issue in cyber-physical systems is that the control system has to keep track of action execution. For this purpose all physical actions are represented as stateful objects in the control system. For each action sent to the physical system the control system introduces a specific stateful object describing this action. These objects occur in four states:

- "sent" for actions just *sent* to the physical system;
- "conformed" for actions which have been *confirmed* by an appropriate event to be successfully executed;
- "failed" for actions where *failure* of execution has been explicitly confirmed; and

[5] Logical variables are treated here as "area{ var A }" indicating that the parameter "area" has a variable value "A" which has to be the same all over the logical formula.

- "unknown" for actions where the control system after a certain specific delay (time out) did neither receive a confirmation nor a denial message.

Actions can be composed to complex actions. Complex actions are just a kind of *macros* allowing us to formulate necessary domain specific relationships between actions:

```
FOR
  counteract-emerging-overcrowding{ area{ var A } }
DO
  action: inform-control-center{ area{ var A } },
  action: inform-station-personel{ area{ var A } }
END
```

The complex action "counteract-emerging-overcrowding", dedicated to a certain area, consists of two different atomic actions "inform-control-center" and "inform-station-personel" responsible for that area have to be executed. Actions can be combined in various ways to complex actions [12]: as concurrent actions, as sequences, or alternatives. Logical conditions may be specified within complex actions. This gives us expressive means to formulate complex activity patterns which can be adapted to concrete scenarios.

### 3.3 Reactive Rules

Complex events allow us to describe generic patterns which can be used to identify and assess concrete situations as combinations of events and states. Reactive rules allow us to assign atomic or complex actions to these situation patterns in order to manage them adequately. Concrete situations which fit a certain situation pattern (complex event definition) are related to actions using the concrete specifications in the situation assessment (through logical variable bindings).

Reactive rules are Event Condition Action (ECA) rules [13,14]: the events part is used to identify the dynamic changes in a system, and the condition part provides the "static" background to assess and interpret the events (spatial relations between events and states, events and states in related systems, etc.). Whenever the concrete situation "matches" with the dynamic (event) and static conditions the ECA rules "fires" – i.e., the actions are initiated. The generic action pattern specified in the ECA rule is instantiated with the concrete situation and static information. The following is an example:

```
ON
  and{
    event e: avg-person-count{ area{ var A }, value{ var P } },
    state s: operation-mode{ area{ var A }, mode{ normal } }
  } where { s at end(e), var P >= 150 }
DO
  action: counteract-emerging-overcrowding{ area{ var A } }
END
```

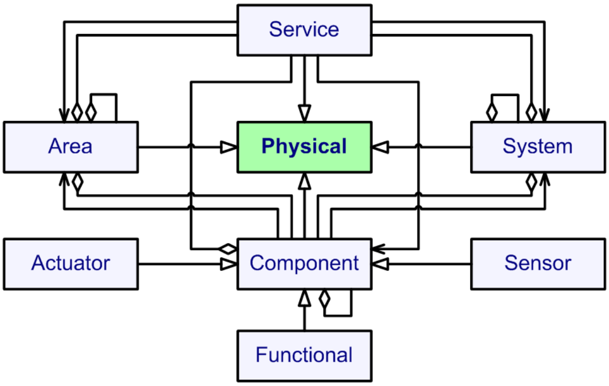


**Fig. 1.** EMILI core ontology: a cut-out of the branch “Physical”

with the meaning that whenever a complex event “`avg-person-count`” results in a count value `P` which exceeds a limit of `150` and the operation mode at the end of this complex event is normal then a complex action denoted as “`counteract-emerging-overcrowding`” is initiated for the area in which the average person count exceeded the limit.

### 3.4 Semantic Models

Critical Infrastructures as complex systems of systems have to process many different kinds of information from different sensors, actuators, and other information sources. Different “pieces of information” may be related in various ways to each other. In order to process this complex information correctly various semantic relations between them have to be taken into account:

- where does the information come from
- how is the information source related to components, systems, areas, etc.
- what are the types of involved components and systems and how are they related to each other, etc.

In order to interpret the many different kinds of information adequately in the respective context a rich semantic information model is needed which provides the necessary information backbone. For this purpose in EMILI various ontologies are created and used as semantic information models:

- ontologies for temporal, spatial, causal, structural and behavioural aspects of large infrastructures;
- a complex event and action ontology as modelling backbone for the event and action rule engine.

The ontologies used in EMILI are built on different levels of granularity: a core ontology, domain specific extensions, and application specific “instance worlds”.

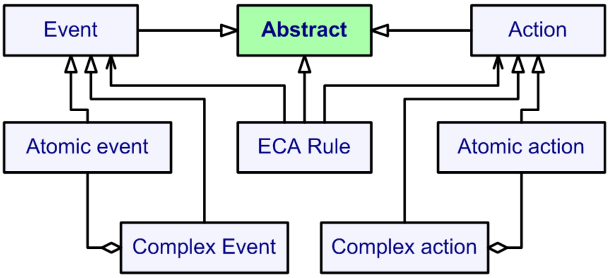


**Fig. 2.** EMILI core ontology: a cut-out of the branch "Abstract"

The core ontology comprises the main kinds of information to be used in CI information modelling. It consists basically of two main categories: physical and abstract. The physical ontology is a coherent set of concepts needed to model complex physical systems (see Fig. 1). It is deliberately restricted to the most elementary and abstract concepts and relations in this area. It mainly contains ontologies for temporal, spatial, causal, structural and behavioural aspects of large infrastructures. The other main part – the abstract ontology – is focused on all event and action concepts needed to describe situation assessment and reactivity in cyber-physical systems and their relationships to the concrete systems, components, and their attributes and relationships (see Fig. 2).

This Core Ontology provides a coherent set of concepts for models of the physical system with all relevant aspects, for the control system with its complex event and state definitions and event condition action rules, and for the relations between the physical and the control system.

One of the main features of cyber-physical systems is the communication between the physical and the control system. This communication goes through two channels: the event channel where sensors and other information sources tell the control system what happens in the physical part, and the action channel allowing the control system to control the situation in the physical system through action messages to actuators, stakeholders, etc. This is one of the main particularities of EMILI's approach with significant requirements to modelling and event and action processing. Sensors may be faulty, events may be lost, false positive events may be sent, actions are sent which can not be executed, etc. – a whole spectrum of specific issues related to event and action processing in cyber-physical systems. The Core Ontology provides the necessary means to model these aspects adequately: complex events can be defined to deal with lost or misinterpreted events, and actions can be defined in relation to confirmation or falsification events.

States play a central role in order to describe the situation of cyber-physical systems. The Core Ontology allows us to relate them to events, to actions and states through various kinds of dependencies.

For each application domain like metro systems, airports, or power grids this Core Ontology is extended to those concepts which are needed to model them adequately. This extension can be done in different ways: through concrete classes introduced as subclasses of the core ontology concepts, or through additional attributes and relations for them. This gives us a great flexibility of information modelling in conjunction with a clear basic structure provided by the Core Ontology.

Finally, the domain ontology is used to model the concrete use cases: a concrete metro system, a concrete airport, power grid, etc. All entities populating such a use case are modelled as instances of domain specific concepts with their attributes and relations.

## 4 Use Cases

EMILI has three quite different but representative use cases: airport, metro, and power grid. With the broad spectrum of issues related to emergency management in these three different domains it was a challenge to develop a generic methodology which can be adapted to concrete requirements. Now we outline the main issues in each use case and show how they are dealt with in EMILI.

### 4.1 The Airport Use Case

An airport is not just a complex building with many areas of different kinds and roles but also a systems with different technical systems for managing people, luggage, airplanes, other resources, etc. This has to be done under various normal, exceptional, and emergency conditions. A network of interacting control systems is operated in order to enable this functionality.

There are many challenges for safety and security in airports. A comprehensive analysis can be found in [15]. For the EMILI airport use case we decided to consider a fire scenario as one of the most important cases.

In order to manage an airport (including such an emergency situation) different kinds of information have to be collected, analysed, and processed for appropriate reactions. The building structure, the kinds, dependencies, and functionalities of various technical systems, the number of passengers in various areas and their movements have to be taken into account. The airport ontology [15] was designed as use case specific extension of the EMILI Core Ontology allowing us to collect and process all relevant information.

The fire scenario is used to demonstrate how events, actions, and simulations work together. The incoming sensor information is analysed by complex event rules to provide a comprehensive situation assessment. This situation “triggers” ECA rules which specify appropriate reactions. Simulations are used to calculate fire and smoke propagation as part of the decision process.

### 4.2 The Metro Use Case

Though metro systems are in some sense similar to airports as infrastructures for the transport of people there are also a couple of differences between them: metro systems are more distributes, they are more open, and not so highly protected. As in airports we need complex information to manage normal and emergency situations in them: the number of people in certain parts, the train positions, building structures, technical systems with their topology, their various dependencies, etc. The EMILI Core Ontology has been extended into a metro domain ontology providing all these kinds of information in a structured way.

The scenario is again focused on the most critical situation: a fire in a metro station or in a metro train. Depending on the whole situation different options for reactions are available. Which size does the fire have and what evolution can be expected? How does the smoke propagate? Which evacuation paths are available due to the size and position of the fire, the smoke propagation, people density, etc.? How long does it take to evacuate people along these paths?

Complex event definitions enable us to bring these different kinds of information together in a coherent way. Positions of fire sensors, positions of trains, data about people density, etc. allow us to assess the overall situation. Appropriate reactions can be specified in ECA rules. Smoke propagation simulation and simulation of people movements allow us to predict scenario evolutions as basis for decision making.

### 4.3 The Power Grid Use Case

Power grids are of course quite different from metros and airports. Today, they are managed through (more or less) sophisticated control systems which get their data from different kinds of sensors in various network components (see [16]). Frequently, these sensor networks do not allow the operators to get immediately a complete picture of what happens in the grid. In the case of component failures a sequence of alarms is initiated by different sensors which can not directly be mapped onto the real network situation. They have to be interpreted by experts in order to find out what really happened. The SCADA telecommunications have partial failures during disturbances. The visibility is degraded, State Estimators have temporary failures (fails to "converge"). Alarms arrive in avalanches that flood the operator so that the operator is pressed to assess the situation as fast as possible, and to take immediate action. But he knows that some errors may make the problem much worse.

*We need intelligent processing of alarms*: not just a filter, but actual interpretation of the possible scenarios. This includes more holistic *situational awareness*: integrating SCADA, alarms, and other contextual information into a quick, effective visualization system.

Complex event definitions allow us to formulate the rules needed for this interpretation as patterns of situations in power grids. They combine sensor information including temporal, topological, and other kinds of data. In this way we can discriminate different kinds of events: breaker events (caused by breakers and their protections), line events (combinations of breaker events and other line events) and link events (produced by line faults between two elements).

Currently, we do not draw conclusions for reactions from these interpretations. This may follow later.

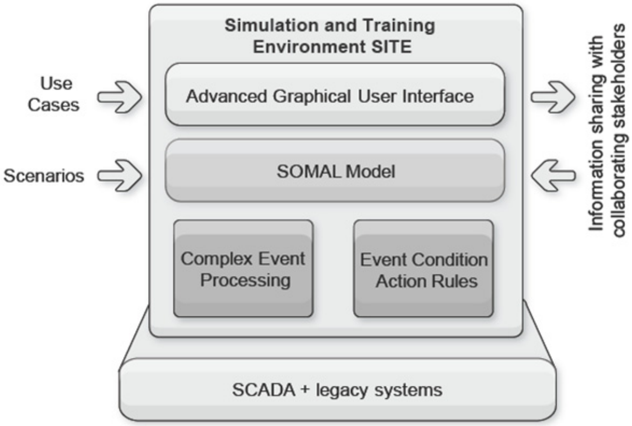

Simulation and Training Environment SITE
Use Cases
Scenarios
Advanced Graphical User Interface
SOMAL Model
Complex Event Processing
Event Condition Action Rules
Information sharing with collaborating stakeholders
SCADA + legacy systems

**Fig. 3.** Overview on the EMILI SITE Simulation and Training Environment

## 5 The Simulation and Training Environment (SITE)

The EMILI Simulation and Training Environment (SITE) will be developed and implemented as an advanced simulation and training system for emergencies in large infrastructures. Using SOMAL, SITE will allow us to model large Critical Infrastructures, their dependencies, and emergency scenarios of any kind with all aspects relevant for crisis and emergency management. It will facilitate the *simulation* of these models using the related emergency scenarios. This simulation will be enabled by our next generation Web technologies (Complex Event Processing and Event Condition Action rules) in conjunction with special purpose simulators (like smoke propagation or load distribution on power grids).

SITE will primarily be used as an evaluation and training environment, but it could be also used as a platform for real applications of our methodology and technology integrated into Critical Infrastructures. Different scenarios can be simulated, and the user can apply actions and reactions to manage emergency situations in order to find the optimal solution.

SITE will allow us to model large Critical Infrastructures, their dependencies, and emergency scenarios of any kind with all aspects relevant for crisis and emergency management. It allows us to *simulate* these models of large infrastructures and of related emergency scenarios. This simulation will be enabled by our next generation Web technologies (Complex Event Processing and Event Condition Action rules) in conjunction with special purpose simulators (like fire and smoke propagation or load distribution on power grids).

SITE can be used in two different ways: as decision support tool integrated into Critical Infrastructure SCADA systems, or as evaluation and training environment.

## 6 Conclusions and Outlook

Critical Infrastructures are complex cyber-physical systems. In order to manage their growing complexity, heterogeneity, and mutual dependencies new control strategies are needed which enable control systems to react adequately under normal, exceptional, and emergency conditions. With traditional software technologies such complex control systems are hard to specify, to implement, to validate, and to maintain.

The EMILI project is dedicated to next generation information technologies for Critical Infrastructure control. These new technologies provide a new level of abstraction to information processing. They enable a descriptive approach to information processing focused on the "what" instead of the "how". Complex event definitions allow us to specify patterns of events and states as expressive means for situation assessment. Complex actions and reactive rules can be used to assign reactions to these situations. Semantic models enable us to describe the many different kinds of information to be used in CI control in a clear and transparent manner. Simulations can be integrated into complex event and action processing whenever detailed information about physical effects is needed for comprehensive situation assessment and forecast as part of decision making.

Describing Critical Infrastructures as complex cyber-physical systems has the advantage that the relationships between the physical and the control part can be defined in a clear manner. Communication between the physical part with its sensors and actuators and the control part is well described through event and action messages exchanged between these two subsystems. Compared with event and action technologies used in other domains like business process modelling or active databases a couple of special effects have to be taken into account. Loss of communication or failures of sensors and actuators are important issues in CI control and need explicit consideration.

In its first 18 months the EMILI project produced the basic concepts for emergency management in CI, for event and action modelling and processing including semantic issues, and went through comprehensive use case modelling. The Simulation and Training platform SITE and the graphical user interface were designed. These results will now be brought into a coherent integrated framework allowing us to demonstrate and evaluate our approach.

**Acknowledgements.** The research leading to these results was funded by the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement n° 242438.

## References

1. Rinaldi, S., Peerenboom, J., Kelly, T.: Identifying, understanding, and analyzing critical infrastructure interdependencies. IEEE Control Systems Magazine 21(6), 11–25 (2001)
2. Rinaldi, S.M.: Modeling and simulating critical infrastructures and their interdependencies. In: Hawaii International Conference on System Sciences, vol. 2 (2004)

3. Franklin, G.F.: Feedback Control of Dynamic Systems, 6th edn. Addison-Wesley Longman Publishing Co., Inc., Boston (1993)
4. Murray, R.M.: Control in an Information Rich World: Report of the Panel on Future Directions in Control, Dynamics, and Systems. Society for Industrial and Applied Mathematics, Philadelphia (2002)
5. Bailey, D., Wright, E.: Practical SCADA for Industry (IDC Technology). Elsevier (2003)
6. EMILI: Emergency Management in Large Infrastructures, EU FP7 project (2010), http://www.emili-project.eu
7. Lee, E.A.: Cyber Physical Systems: Design Challenges. In: Proceedings of the 2008 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing, pp. 363–369. IEEE Computer Society, Washington, DC (2008)
8. Sha, L., Gopalakrishnan, S., Liu, X., Wang, Q.: Cyber-Physical Systems: A New Frontier, pp. 3–13. Springer, US (2009)
9. Lee, E.A., Seshia, S.A.: Introduction to Embedded Systems, A Cyber-Physical Systems Approach (2011), http://LeeSeshia.org
10. Daneels, A., Salter, W.: What is SCADA? In: Proceedings on the International Conference on Accelerator and Large Experimental Physics Control System, Trieste, Italy, pp. 339–343 (1999)
11. Klein, R., Rome, E., Beyel, C., Linnemann, R., Reinhardt, W., Usov, A.: Information modelling and simulation in large interdependent critical infrastructures in IRRIIS. In: Setola, R., Geretshuber, S. (eds.) CRITIS 2008. LNCS, vol. 5508, pp. 36–47. Springer, Heidelberg (2009)
12. Hausmann, S., Brodt, S., Bry, F.: EMILI Deliverable D4.3. DEAL - Concepts and Examples. Technical report, Ludwig-Maximilians University Munich (2011)
13. Alferes, J.J., May, W.: Evolution and Reactivity for the Web. In: Eisinger, N., Małuszyński, J. (eds.) Reasoning Web 2005. LNCS, vol. 3564, pp. 134–172. Springer, Heidelberg (2005)
14. Behrends, E., Fritzen, O., May, W., Schenk, F.: Embedding Event Algebras and Process Algebras in a Framework for ECA Rules for the Semantic Web. Fundamental Information 82(3), 237–263 (2008)
15. Vraneš, S., Mijović, V., Tomašević, N., Konečni, G., Janev, V., Kraus, L.: EMILI Deliverable D3.1 Annexe A. Specific report for use case I, Airport. Technical report (2010)
16. Español, J.L.M.: EMILI Deliverable D3.1 Annexe C. Specific report for use case III, Power Networks. Technical report (2010)

# Dynamic Evacuation Guidance as Safety Critical Application in Building Automation

Armin Veichtlbauer and Thomas Pfeiffenberger

Salzburg Research Forschungsgesellschaft m.b.H.
Jakob Haringer-Str. 5/III, 5020 Salzburg, Austria
{aveichtl,tpfeiff}@salzburgresearch.at

**Abstract.** In building automation systems a bunch of very diverse applications have to be executed at the same time. This includes applications with special safety and security requirements. One of these applications is the dynamic evacuation guidance, i.e. electronic signs which show the direction of the most fortunate evacuation path from public buildings like schools, universities, office buildings or buildings of public authorities during emergency situations like fire or also terrorist attacks. In this context we explored the new and innovative solution Flexit, focussing on safety issues of the communication infrastructure.

**Keywords:** Building Automation, Evacuation Guidance, Communication Infrastructure, Safety Life Cycle, Availability Analysis.

## 1 Introduction

An essential measure in case of natural disasters, terrorist attacks, industrial accidents, or at presence of significant threats is to guide people out of the dangerous areas. To support the evacuation, public buildings have to have an evacuation scheme (i.e. defined evacuation paths from every position in the building), and appropriate signs [1] to mark these paths.

Yet in the past 50 years there have been almost no innovations to these evacuation schemes. For catastrophes static evacuation schemes may cause a massive threat to human lives [2]. An evacuation scheme which adopts dynamically to the current situation leads to a considerable safety gain for the people within a building.

The “Flexit” (flexible exit) system [3] aims to provide a situation aware calculation of best evacuation paths, taking into account sensory information from the fire alarm system, which is mandatory for public buildings of a certain size. The sensors include temperature, smoke, and gas detectors. With these data, dangerous zones in the building can be identified and thus avoided for the evacuation path calculation.

More concrete, a graph representation consisting of elementary paths can be defined and associated with a “cost function” for all of the elementary paths. With that, the paths with the lowest total price from any starting point (sources) to defined exit points (sinks) can be calculated based on the well-known Dijkstra algorithm.

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 58–69, 2013.

These paths should then provide the highest probability to get out of the building without harm.

During the last three years, our research group tested and evaluated the prototypical Flexit solution in technical, judicial, economical, and social respect and worked on several enhancements in order to provide additional required functionalities. The requirements analysis has been made in cooperation with the municipal fire brigade of the city of Salzburg.

The research work included also questions of responsibilities and liabilities. The safety characteristics of a system are described by its safety functions. In case of (partly) outages of the system it must be clear to determine, whether the safety functions have worked properly to exclude liabilities of the operator or the technology vendor. Thus the conformity to all existing national and international norms and standards has to be ensured.

Finally we included questions about the social aspects of the proposed technical solutions, like acceptance or impacts of the regarded tools.

The technological questions we addressed in our work covered the following:

- Is the Flexit system able to fulfil the technical requirements for a safe and highly available evacuation guidance system? If not, where are the problems and how can they be assessed?
- Which additional functions of the system are imaginable / desirable? Which measures have to be taken to be able to provide these functions in future versions?
- Based on which technologies a seamless integration of external systems into existing building automation systems can be effected? How could a generic, standardised interface be realised?
- How could a mobile command and control centre (“CCC”) for action forces be realised? Which functions does it have to provide and how can these be implemented? How can the integration of sensors and actuators be effected?

To answer these questions a prototypical solution (“proof of concept”) based on the Flexit system has been designed and implemented. This solution was tested first in the lab and then at the University of Applied Sciences in Salzburg with two show cases. The test results have been evaluated including technical (functionality and quality) as well as non-technical (acceptance) factors.

## 2 Requirements

Unlike with other solutions for dynamic evacuation guidance, the Flexit system works distributed: A network of “LIENs” (“Local Intelligent Emergency Nodes”) is responsible to share all necessary information to calculate evacuation paths, and the calculation is done independently for all LIENs (see chapter 5). It is obvious, that this approach demands a dependable communication infrastructure [4].

Furthermore, the question of interoperability of the Flexit system not only with the fire alarm system, but also with third party systems is essential in order to collect and analyse a comprehensive representation of the environmental situation in the building with all accessible sensors.

For the fire brigade an additional goal was to get manual access to the evacuation control. This can be used to provide the officer in charge with all of the systems knowledge, functioning as decision support system. It can also be integrated in a mobile command and control centre to allow the officer in charge to supervise and if necessary manually overrule all current settings of the evacuation control system (e.g. to open or close remote controlled doors, or to change the preferred evacuation path, etc.).

In the following we list the requirements which we identified in this analysis, sorted by the area of origin [5].

### 2.1 Functional Requirements

First the basic functional requirements to an IT supported evacuation system have been investigated:

- **Situation Awareness**: Automated (sensor supported) ascertainment of the current situation in an emergency situation
- **Decision Support**: Algorithms for calculation of appropriate evacuation paths or attack routes for action forces
- **System Integration**: Connection of the mobile command and control centre to the evacuation control system via a generic interface
- **Communication Infrastructure**: Generation of an infrastructure for the communication of the officer in charge with the action forces and the evacuation control

### 2.2 Non-functional Requirements

Second the relevant non-functional requirements (quality requirements) have been identified:

- **Usability**: Easy to use interface without laborious setup in the field
- **Performance**: Sufficient quality of the communication infrastructure
- **Flexibility**: Generation of a universal system interface which allows the integration of third-party systems; usage of generic data transmission protocols
- **Mobility**: Long-term independence from fixed energy infrastructures and usage of handy (small and lightweight) devices
- **Robustness**: Maximum effectiveness of the system also for adverse operating conditions
- **Security**: Protection against attacks from non-authorised persons and protection of access rights and system information

### 2.3 Economic Requirements

Besides the technical requirements also economic and commercial requirements have been considered, since we aimed to provide a commercially available product at the end of the development chain:

- **Development costs**: The product life cycles in building installations range around 3 to 5 years. In order to avoid a complete new development at latest all 5 years the system shall be set up in a modular manner having the possibility to reuse the modules to a great extent.
- **Production costs**: Due to the decentralised architecture of the Flexit system every evacuation path display ("Fluchtwegsanzeige", "FWA") has its own evacuation path calculation, network interface, power supply, etc. Thus the production costs are relatively high, consequently also the price of a single FWA is rather high (about € 2.000,-). It has to be checked whether some functionalities can be centralised without losing the advantage of redundancy.
- **Installation costs**: The most important cost factor for the installation is the wiring of the total arrangement. Thus the usage of alternative communication technologies (radio connections, power line) has to be checked; eventually these technologies have to be integrated in future versions of the evacuation control system.

### 2.4 Social Requirements

In order to ensure an orderly evacuation of civilians in emergency cases the impact of the evacuation guidance system has to be guaranteed. For that purpose the following items have to be minded [6], [7]:

- **Perceptibility**: Above all, the evacuation guidance system has to be easily perceptible, i.e. it has to prevail over the local stimulus background. This can be broken down to several aspects: Visual cue, legibility, aesthetic effect, attention density.
- **Context sensitivity**: The evacuation guidance system has to account for location contexts (e.g. display the right evacuation paths in zones of danger and warnings outside) or for target group contexts (display evacuation paths for civilians and attack paths for action forces).
- **User acceptance**: Finally, the system has to be known to the target groups, i.e. it has to be identified correctly as an evacuation guidance system by the civilians. Also, the target group has to trust the displayed evacuation paths ("the displayed path is the best way to leave the building").

## 3 Communication Infrastructure

As a result of the requirements analysis, one of the main goals in our research work was to provide a **dependable communication infrastructure** as defined in [4]. Accordingly, the communication subsystem of the prototype had to ensure certain reliability, availability, safety, security and robustness features.

The key feature for communication infrastructures is availability. The design of the networks has to guarantee a minimum outage percentage; only with available base network services the higher layer questions of security and reliability can be addressed in a reasonable manner.

Furthermore, availability is strongly connected to safety. The safety lifecycle begins with a risk analysis. According to the "IEC 61508" standard [8], the risks are classified dependent on the risk parameters probability, avoidance possibility, frequency, and consequences. For each class ("Safety Integrity Level", "SIL") a maximum outage limit is defined. There are four SIL for two different operating modes (occasional vs. continuous), with failure limits up to < $10^{-8}$ for the highest level, i.e. an average outage time below 0.4 seconds per year [9].

As indicated, the failure limits are assessed by the presence of safety functions, i.e. a system is called safe, if the safety functions work according to the required outage limits. The definition of the safety functions has to be done very carefully, especially for applications which are critical to human lives, which is doubtlessly the case for evacuation guidance.

### 3.1 Availability Calculation

For communication networks, the failure percentage correlates with availabilities of the network components, i.e. sensors, actuators, concentrators, routers, gateways, etc. as network nodes and wired or wireless links as network edges.

For a fixed route from node A to node B the probability of a correct data transmission (i.e. the availability of this route) depends on the single probabilities of all links on this route and of the routers which are connecting these links, provided that A and B themselves are up and running. A transmission is successful if and only if all components are working correctly, thus the total availability can be derived by serial combination of single availabilities [4].

$$A_t = \prod_{i=1}^{n} A_i = A_1 * \ldots * A_n \tag{1}$$

It can be seen easily that this calculation leads to a rather low total availability $A_t$, as all $A_i$ are < 1:

$$\mathrm{A_t} < \min(\mathrm{A_1}, \ldots, \mathrm{A_n}) \tag{2}$$

Here the only solution to overcome this problem is the use of redundancy. The more (disjunctive) routes exist in a network, the higher is the probability that at least one of them is working. Thus we have to build the product of the counter probabilities to calculate the total availability:

$$\mathrm{A_t} = 1 - \prod_{\mathrm{i}=1}^{\mathrm{n}} (1 - \mathrm{A_i}) \tag{3}$$

In meshed networks, both effects will take place: In many cases there will be more than one possible route from A to B, yet these routes are not necessarily disjunctive.

The total availability can thus be calculated by a nested application of the two basic formulas, dependent on the network's topology.

For given target values for the total availability $A_t$ (according to the outcome of the risk analysis) it can be tested whether a network fulfils the availability requirements or not, and in case of failing the network topology can be optimised, e.g. by adding more redundancy, or by using more reliable hardware components.

### 3.2 Network Technologies

For the network's nodes, i.e. the routers, gateways and end systems, the availability is in most cases stated by the vendors, as a result of the stress tests they conducted with the hardware. For the network's links the availability depends very much on the chosen technology, especially the decision whether to use wired or wireless data transmission has an important impact. Table 1 gives an overview of some interesting properties of selected wired and wireless technologies:

**Table 1.** Properties of selected data transmission technologies

| | **Ethernet** | **WiFi** | **ZigBee/IEEE 802.15.4** | **Power Line** |
|---|---|---|---|---|
| Bit Error Rate (BER) | very low | very high | high | low |
| Bandwidth [Mbit/s] | 10 - 1000 | 11 – 54 | 0,02 – 0,25 | 10 - 200 |
| Mounting Effort | very high | very low | very low | high |
| Attack Vulnerability | low | very high | very high | very low |
| Catastrophe Vulnerability | very high | low | low | high |

Here, the BER is the basic property that describes the availability of the data link. Additionally, the hardware availability of cables and sockets has to be taken into account, which is especially important in the case of natural catastrophes.

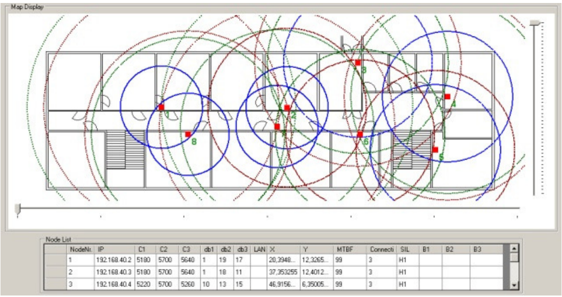
The BER turns out to have much lower values for wired technologies. Also in case of terrorist attacks the wired technologies have advantages over wireless solutions, as the wires have to be physically damaged to prevent their functioning, whereas wireless transmission can efficiently be supressed by jamming.

Yet wireless solutions have the advantage of a much easier and cheaper mounting, especially in case of modifying already existing systems. Thus, as a backup solution which provides some additional redundancy, they may be very useful also for safety critical environments, especially in cases of grievous physical destruction.

### 3.3 Connectivity Simulation

To find the best positions for radio network components (like our LIENs) in order to ensure the continuous availability of radio connections the simulation tool Wplan [9] was used. The tool calculates the signal strength of simulated radio senders at every position in a testing area denoted by a map of the testbed. The calculation is based on the distance of the examined positions to the senders and the attenuation resulted from walls or obstacles, yet other possible influence factors like the presence of persons or interference with other technological equipment are not considered.

Using this tool it can be evaluated whether each place of the testing area has a minimum of three radio connections available, according to the results of the risk analysis after IEC 61508. These are used for redundant connections between LIENs and for redundant accessibility to "Mobile Field Devices" ("MFDs"), carried by the action forces. If the triple connectivity is not obtained, manual changes of the LIENs' positions can be tried, or the adding of further LIENs. This manual optimisation process guarantees the required redundancy for high availability communication infrastructures in case of emergencies. Fig. 1 shows a sample simulation with Wplan.



| NodeNr. | IP | C1 | C2 | C3 | db1 | db2 | db3 | LAN | X | Y | MTBF | Connecti | SIL | B1 | B2 | B3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 192.168.40.2 | 5180 | 5700 | 5640 | 1 | 19 | 17 | | 20,3948... | 12,3265... | 99 | 3 | H1 | | | |
| 2 | 192.168.40.3 | 5180 | 5700 | 5640 | 1 | 18 | 11 | | 37,353255 | 12,4012... | 99 | 3 | H1 | | | |
| 3 | 192.168.40.4 | 5220 | 5700 | 5260 | 10 | 13 | 15 | | 46,9156... | 6,35005... | 99 | 3 | H1 | | | |

**Fig. 1.** Connectivity simulation with the tool Wplan

## 4 Robustness Middleware

In this part we present our work in analysing different protocols which can be used to establish a robust layered architecture, and their deployment in our prototype. The most important requirements in order to support robustness of the evacuation system can be bundled to the following categories:

- Data Representation: includes a standardised representation of the address space (related information of specified objects should be: description, attributes, references, and methods)
- Security Model: includes authentication, authorisation, confidentiality (signing and encryption), integrity, auditability, and availability
- Configuration Service: includes address auto configuration, service registration, and service discovery
- Redundancy: includes keep alive channels, redundant servers, and detecting of server failures

Table 2 shows the collection of analysed protocols and their conformance to the robustness requirements. According to this analysis, two technologies could be identified which comply fully with the required functionalities to establish a robust middleware: “BACnet” (“Building Automation and Control Network“) [10] and “OPC-UA” (“OLE for Process Control Unified Architecture“) [11].

**Table 2.** Robustness requirements and their mapping with selected technologies

| Robustness Requirements | OPC-UA | BAC net | Modbus/ TCP | SIP | Soap/WS Security |
|---|---|---|---|---|---|
| Data Representation | ✓ | ✓ | - | - | - |
| Signing / Encryption | ✓ | ✓ | - | ✓ | ✓ |
| Authentication / Authorisation | ✓ | ✓ | - | ✓ | ✓ |
| Notifications / Alarms | ✓ | ✓ | - | ✓ | ✓ |
| Registration / Discovery | ✓ | ✓ | - | ✓ | ✓ |
| Abstract Address Scheme | ✓ | ✓ | - | ✓ | ✓ |
| Keep-alive / Heartbeat | ✓ | ✓ | - | - | ✓ |

### 4.1 System Architecture

The system architecture of the prototype is based on the features of the used protocols. The generic interface of the robustness middleware guarantees an easy and simple integration to other systems, like fire alarm systems and other building service management systems. Fig. 2 shows the overall system architecture including the generic application interface of the robustness middleware.

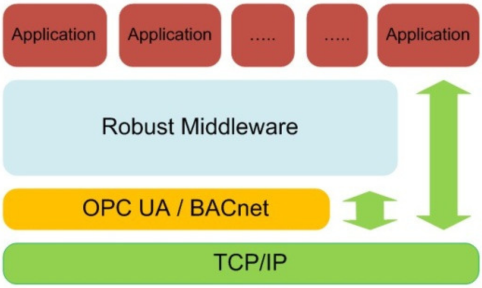


**Fig. 2.** System architecture of the prototype

The following list gives an overview about the roles which are defined within the prototype:

- Client: reads or writes data on a server
- Server: provides data architecture
- Information logger: collects information from the system

- Registrar: provides access control management
- Mediator: translates information into other protocols

A focus of the prototype is to use standard network components. Thus we use TCP/IP for the underlying transport protocol. To connect devices to the system, a registrar device is used. To facilitate the communication with the prototype for external devices, the mediator device translates the inputs from these devices into BACnet respectively OPC-UA messages.

## 4.2 Network Scheme

Fig. 3 shows the components of the implemented prototype, which incorporate the roles mentioned above, albeit not their full functionality. Basically the prototype consists of two parts: The "Dynamic Evacuation System" ("DES") based on the Flexit system and the "Mobile Command and Control Station" ("MCCS").

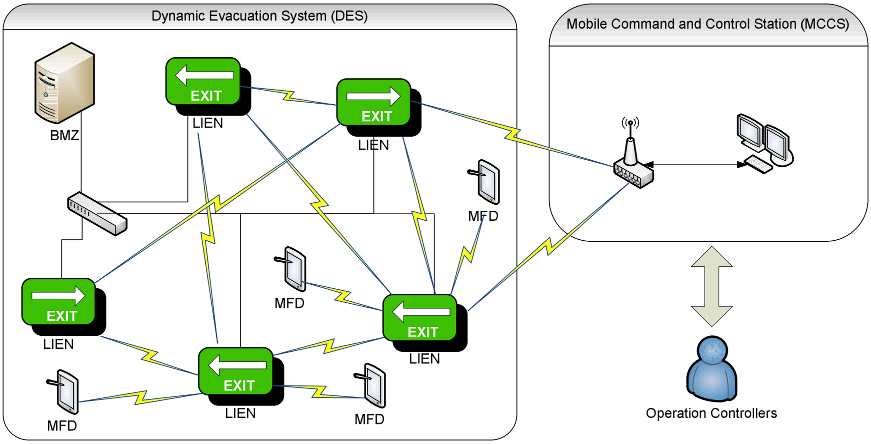

Dynamic Evacuation System (DES)
Mobile Command and Control Station (MCCS)
BMZ
EXIT
LIEN
MFD
Operation Controllers

**Fig. 3.** Network scheme of the prototype

The DES consists of several LIENs (see chapter 2), which serve as communication and displaying entities, as well as MFDs (see chapter 3), which provide personalised information (relevant to the respective evacuation scenario) to action forces. The communication infrastructure described in chapter 3 is used to connect the LIENs and MFDs among each other and to the MCCS. Here, the LIENs are responsible to establish the redundant connection to the MCCS and the MFDs.

It is possible to establish multiple communication paths via wired or radio networks. Also the communication with the fire alarm system can be established by using different physical communication standards, like power line, Ethernet or radio standards.

# 5 Validation

Based on the requirements and the system architecture we built a prototype, consisting of selected functions (see chapter 4), which should allow us to evaluate the feasibility of our approach (proof of concept). The validation of the prototype was based on distinct parts:

- The validation of the infrastructure, i.e. the physical communication network, including availability of devices and links
- The validation of the software, i.e. the robustness middleware, including features of the software development process
- The validation of the whole system in the field, i.e. the planning, conduction, and evaluation of real world show cases

## 5.1 Infrastructure Validation

The infrastructure validation consisted basically of a set of lab tests, in which we tested the basic implemented features of our communication infrastructure, which was a meshed network consisting of wireless links (IEEE 802.11 a/b/g standard) combined with Ethernet links. The hardware we used consisted of the “MikroTik Routerboard 532A” and its add-on “Routerboard Interface R52” with 3 Ethernet and 3 wireless connections for each node (LIEN).

As the network is dependent on the proper working of lower layer protocols, this includes the testing of some software features also, yet these features were not implemented by our group. We used the commonly known “Spanning Tree Protocol” (“STP”) and its faster variant “Rapid Spanning Tree Protocol” (“RSTP”) [12] for identifying the routes in the meshed network.

For the tests we made in the lab, we first studied the switch times in case of broken links. For the STP we measured switch times from over 30 s until the new route was established and all routing entries have been changed. As this is considered too long for safety relevant applications, we decided to use RSTP for all following activities in order to obtain switching times of below 10 s [9].

The actual lab tests consisted of different scenarios with outages of components. We used 4 LIENs interconnected with a wireless full mesh, i.e. one link between each of the LIENs, and a control PC connected via Ethernet to LIEN 1. All the wireless links used different frequencies, thus we avoided that more than one device could receive the same message. This gave us the possibility to test the links separately by switching them on and off.

## 5.2 Software Validation

During the software development process different quality limits have been set. To reach this quality limits test scenarios in different scales have been defined: Unit Tests, Module Tests, Integration Tests, and System Tests.

The first three were established in a “Continuous Integration” (“CI”) environment. The CI consists of a full automated build, test and deploy system. If a new code version is checked into the version control system the CI starts automatically. To raise the code quality and to give feedback to the programmers some statistical code review tools are included to the CI system. The documentation of the CI process is available via a web interface.

### 5.3 Show Cases

To validate that the DES fulfils the defined requirements (see chapter 2) of all involved partners we had to perform system tests in real world scenarios. For that purpose we established a testbed for two show cases at the site of the University of Applied Sciences in order to validate the functionality of the installed DES.

Based on the results of the simulations (see chapter 3) 16 LIENs were installed in the 4th floor of the building. As a first step the availability of the redundant network connections and the basic functions of the installed system were tested with distributed tests on all LIENs. Then all functions which had to be used by the involved participants (the fire brigade, the facility management stuff, the project researchers and technicians) were prepared for the fire drill.

The goal for the first show case was to validate the communication interface and the installation and configuration of the LIENs via a provisional control interface. With that pre-prototype an evacuation of the 4th floor of the building was performed. In the second show case a full fire drill including a complete evacuation of the building was conducted. Here we used the more advanced prototype, with a MCCS as control interface, which was communicating with the LIENs over the BACnet protocol. Both show cases were monitored by different cameras and involved persons. All evacuated persons and involved persons were interviewed about the fire drill later on, as a basis for the impact evaluation of the prototype technology.

## 6 Results and Further Work

### 6.1 Test Results

As a result of the lab tests, we were able to show that the system was able to transmit data between all nodes, if both nodes and at least one possible route between them were up. The rerouting took a maximum of around 4.5 s, which is far below the required 10 s. Yet the scaling properties could not be tested in the lab tests. Thus these properties had to be assessed in the final system tests, i.e. the show cases we conducted at the University.

The two show cases demonstrated the correct functionality of the installed DES. It was possible to evacuate all persons in the area. All defined requirements for the DES were fulfilled correctly, and all considered standards were satisfied. As a result of the interviews with the evacuated persons, we observed a higher cognition of the displays of the DES compared to conventional signs.

### 6.2 Further Work

Further research efforts should be made to investigate the possibility of using the DES also as a guiding system for fire fighters in an emergency scenario. The possibility of using sensors and actuators to analyse the local situation in an emergency case (situation awareness) opens new research questions.

Finally, the Flexit system's conformance to all relevant standards has to be approved in order to allow for a wider deployment.

## References

1. ÖVE/ÖNORM E 8002-1: Starkstromanlagen und Sicherheitsstromversorgung in baulichen Anlagen für Menschenansammlungen, `http://www.kfe.at/empfehlungen/verbindlbest/befristungen/OEVE_OENORM%20E%208002-1.pdf`
2. Atemschutzunfaelle.eu: 11. April 1996 - Flughafenbrand Düsseldorf –Orientierungsprobleme. 112-Sonderausgabe "Flughafenbrand Düsseldorf", `http://www.atemschutzunfaelle.de/ausbildungsunfaelle/1996/b19960411-duesseldorf.html`
3. Flexit Sicherheitstechnik: Evakuierung - Der neue Stand der Technik, `http://www.flexit.at/index.php?id=12`
4. Panholzer, G., Veichtlbauer, A., Dorfinger, P., Schrittesser, U.: Simulation of a Robust Communication Protocol for Sensor Data Acquisition. In: Proceedings of the Sixth International Conference on Wireless and Mobile Communications, Valencia (2010)
5. Veichtlbauer, A., Hofmann, U.: RescueNet und CaR. In: Wissenschaf(f)t Sicherheit, Tagungsband der Fachtagung Sicherheitsforschung 2011, Vienna, pp. 35–42 (2011)
6. Peek, L.A., Mileti, D.S.: The history and future of disaster research. In: Bechtel, R.B., Churchman, A. (eds.) Handbook of Environmental Psychology, pp. 511–524. Wiley (2002)
7. Proulx, G., Reid, I.: Occupant behavior and evacuation during the Chicago Cook County Administration Building fire. Journal of Fire Protection Engineering 16, 283–309 (2006)
8. International Electrotechnical Commission (IEC): IEC 61508: Functional safety of electrical/electronic/programmable electronic safety-related systems (2005)
9. Schrittesser, U.: Synthese von redundanten vermaschten WLAN. Diploma thesis, Polytechnical University of Salzburg (2008)
10. ANSI/ASHRAE Standard 135-2004: BACnet – A Data Communication Protocol for Building Automation and Control Networks. American Society of Heating, Refrigerating and Air-Conditioning Engineers Inc. (2004)
11. Mahnke, W., Leitner, S.H., Damm, M.: OPC Unified Architecture. Springer (2009)
12. Galea, M., Pozzuoli, M.: Redundancy in Substation LANs with the Rapid Spanning Tree Protocol. Electric Energy T&D Magazine (2003)

# Defeating Node Based Attacks on SCADA Systems Using Probabilistic Packet Observation

Thomas Richard McEvoy[2] and Stephen D. Wolthusen[1,2,*]

[1] Norwegian Information Security Laboratory,
Department of Computer Science,
Gjøvik University College, Norway
[2] Information Security Group,
Department of Mathematics,
Royal Holloway, University of London, UK
{T.R.McEvoy,stephen.wolthusen}@rhul.ac.uk

**Abstract** Supervisory control and data acquisition (SCADA) systems form a vital part of the critical infrastructure. Such systems are subject to sophisticated attacks by subverted processes which can manipulate message content or forge authentic messages, undermining the action of the plant, whilst hiding the effects from operators. In this paper, we propose a novel network protocol which, using techniques related to IP Traceback, enables the efficient discovery of subverted nodes, assuming an initial detection event. We discuss its advantages over previous techniques in this area and provide a formal model.

## 1 Introduction

Supervisory control and data acquisition (SCADA) systems are subject to sophisticated and persistent attacks, e.g. [1] based on the subversion of network nodes or control units. Attack aims are not confined to creating short-lived, albeit disastrous, effects, but based on medium- to long-term strategies to disrupt plant operation, and necessarily make use of advanced concealment techniques, which require the concomitant development of detection methods to uncover and remove them. The concept of utilising network protocols to aid intrusion recovery on SCADA systems has been proposed previously [2]. A detection event is assumed[1] [3], indicating a process under attack. By combining packet tracing methods with a knowledge of control system responses, it is possible to reason over which network nodes, or control units, are under adversary control and initiate appropriate recovery strategies to eliminate subverted nodes from participation in plant operations, but can be made efficient based on topology and causality side information.

In this paper, we propose a simpler, but more robust approach to tackling this problem. Setting aside momentarily the consideration of whether control units are subverted, we concentrate on observing (probabilistically) packet *behaviour* along network routes, i.e., considering both packet routing and content alteration. This enables us to detect efficiently whether or not network nodes are forging or manipulating packages. If no such

[*] Corresponding author.

[1] Based on detecting inconsistencies in plant signals.

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 70–80, 2013.

activity is discovered, in the presence of anomalous plant conditions, we may assume that the control unit is subject to adversary subversion [2]. We subsequently prevent subverted nodes from participating further in system communications, while repair and recovery procedures are carried out.

We assume an adversary is capable of subverting network nodes which act as *agents* on his behalf in the network. Such agents can forge or manipulate individual system instructions perfectly, while concealing the result from the operator. These are well-formed and hence not normally detectable by protocol analysis. However, their overall control of the system is assumed to be incomplete, so inconsistencies with expected *plant* behaviour may arise. Under such conditions, we initiate the proposed protocol which causes network nodes to probabilistically create and hash copies of packets, both input and output, and return these to the operator. These can be checked against the original for validity, or declared invalid if no original exists. The information generated allows the operator to determine along a given route the point at which manipulation took place. Hashing the packets – using an approach based on dynamic, delayed key-release hashing – protects the protocol against agent, or adversary, subversion. Our approach is computationally efficient compared with previous uses of IP tracing techniques. The method can also be combined with dynamic re-routing protocols to divert network messages away from subverted nodes, assuming the existence of redundant, normally unemployed network routes. This enables attacks to be dealt with without disrupting production, a key advantage on many SCADA systems.

In section 2, we review related work. In sections 3, we outline our problem and approach. We model the topology and operations of SCADA system in section 4. We provide details of our probabilistic observation protocol in section 5. We discuss design implications, storage and space requirements in section 6, concluding and outlining future work in section 7.

## 2 Related Work

SCADA systems are publicly known to been subject to attacks characterised by a *persistence* capability incorporating the use of concealment techniques with the capability to manipulate process signals on control units [1], or on communication routes [4]. Such attacks have been discussed extensively [4,5,6]. Detective strategies based on protocol analysis, or statistical signal analysis by themselves can be shown to have weaknesses in uncovering such attacks [7], particularly in the face of an adversary with the ability to subvert network nodes in the system [3]. Hence conjoint reasoning over both communication and control functionality, making use of advanced state estimation techniques, is necessary to detect such attacks [2] [8]. This can be split into the analysis of the integrity of network nodes on the one hand and the analysis of the integrity of process control units on the other. Techniques related to IP traceback [9,10,11,12,13,14] have been proposed to help solve this kind of problem in the context of denial of service attacks and such techniques are easily adaptable to this problem, particularly for probabilistic cases [15]. In [2], this requires demonstrating the existence of independent routes and then considering their consistency in terms of the model to determine route

[2] Subject to further investigation.

integrity. However, this approach relies on the knowledge of routes and system operations in a highly stable topology. The behaviour of packets *en route* is not considered, nor is adversary ability to re-route messages, and it is not clear that operator inputs are consistently delivered (except by means of complex inductive reasoning).

In this paper we argue that we need to simplify the reasoning process involved by observing packets behaviour during communication. We also avoid the implicit assumption that the subverted nodes which manipulate output data are the same as those manipulating input data. If we can determine where node subversion has taken place, the use of state estimation techniques [16], can be confined (except for initial detection) to consideration of the integrity of control systems, hence lowering the computational burden of the approach. We base our technique on IP Traceback methods (noting that we are not limited to a particular protocol) which were originally devised to deal with denial of service attacks. Similar techniques are also used in other domains, e.g. the analysis of legitimate traffic in a network, congestion control, robust routing algorithms, or dynamic network re-configuration [12]. We also follow the assumptions established in [13] for DoS attacks regardless of a specific target domain:

1. Packets may be lost or reordered,
2. Attackers send numerous packets,
3. The route between attacker and victim is fairly stable,
4. Network nodes are both CPU and memory limited, and
5. Network nodes are not widely compromised.
6. An attacker may generate any packet,
7. Multiple attackers may conspire,
8. Attackers may be aware they are being traced,
9. A compromised router can overwrite any upstream information

Clearly, assumption 2 is not relevant with regard to packet manipulation in a SCADA environment, but other assumptions continue to hold and are augmented in the adversary model we assume – see section 3 and [3].

Various approaches to packet marking were proposed with differing payoffs in the context of denial of service attacks such as link testing, router logging, behavioural monitoring, packet-based traceback or by creating information packets separate from data packets. Each method had different overheads and payoffs. Probabilistic packet marking was proposed both to minimise message overheads and router logging requirements, but at the expense of introducing uncertainty due to the probabilistic sampling of the flow path and the ability of the attacker to inject false packets [10]. Other hybrid methods were also introduced to cut on message and storage overheads further, without solving the injection problem [14]. Our approach is based on the strategy proposed for the iTrace protocol and other techniques discussed in [14], but using probabilistic observation of packet contents.

The advantage of our approach is that it enables us to efficiently pinpoint agent processes. It also identifies packet injection as well as manipulation. Compared with probabilistic packet marking where the expectation of the number of packets required to trace an attack is given by a hypergeometric probability distribution and is in the order $O(nlogn)$ [13], we are able to use knowledge of network routing and operations (defined algebraically) to rapidly eliminate valid nodes. This elimination can take place along multiple routes simultaneously depending on edge-disjointedness of the network graph. In the remainder of the paper we rely on an algebraic formulation of the protocol [2] [17] as this offers a means of designing and proving protocol behaviour abstracted from specific SCADA environments.

## 3 Probabilistic Tracing of Subverted SCADA Nodes

We retain the assumptions outlined in section 2. Our adversary model is based on [3], permitting subverted nodes to act as *agents* on behalf of the adversary and able to manipulate process units by sending false instructions whilst simulating normal operation to system operators, or else they can mimic negative conditions which cause operators to enter inappropriate commands. In this paper, we do not directly consider the problem of subverted control units but seek to detect agent nodes which are subverting communications. However, we assume that if no such agents are detected that control units will become subject to investigation.

On detecting a system anomaly, we initiate the protocol by setting a flag and providing network packets with unique identifiers based on packet characteristics. Network nodes probabilistically copy certain packets and apply a cryptographic hash using a randomly generated time-released key (to avoid adversary forgery). The resulting packet is sent to the operator and may be compared with the original packet and any subsequent copies. The protocol applies to both input and output messages. This enables the discovery of agent nodes on routes. Subsequently, dynamic re-routing techniques using redundant, initially unutiliseed routes (whose *planned* existence we assume) may enable the operator to bypass the subverted nodes where topology is at least partially known.

## 4 SCADA System Model

A SCADA (supervisory control and data acquisition) system consists of an operations center, manned by one or more operators who use an HMI (human-machine interface) to view the state of the system as determined by signals from a set of control units and associated sensors. The operators in response to changes in the system state, or as the result of management decisions, send signals to control units which alter the operational parameters of those units. In fact, large parts of this process may be automated, or under the control of expert systems. Such control is “supervisory” because the control units take responsibility for enacting changes and subsequent real-time local adjustments to actuators operating e.g. valves, pumps, and solenoids. All signals are communicated by network or other (e.g., satellite, point to point, PSTN, broadband, wireless) communications routes. The history of the system is recorded in a database (“the historian”) and may also be communicated to management information systems on corporate networks in near real-time. Here, we model a SCADA system $G$ as a directed multi-graph $\overrightarrow{G}$ which we call the *initial network model.*

**Definition 1.** ***Initial Network Model*** *Let $G$ be a SCADA network. Let $\overrightarrow{G} = V(\overrightarrow{E})$ be a directed multi-graph such that each $v \in V$ represents a network node which we also label a* process *and each $\overrightarrow{e} \in \overrightarrow{E}$ is a directed edge which represents* potential communication *means between processes $u, v \in V$ which we also label a* channel. *We call $\overrightarrow{G}$ the* initial network model *of G. We assume that the topology of $\overrightarrow{G}$ is fixed w.r.t channels and processes.*

We note, at this point, that we have not yet defined operations over $\overrightarrow{G}$. For example, while all channels are potential means of communication, we have not specified which ones are used by processes, or in what order, or under what conditions. We use the terminology of *channels* and *processes* to aid a discussion of operations which we subsequently define algebraically. We define a potential network route $R$ to be a non-empty, non-cyclical path, consisting of directed edges which are channels between two vertices $U$ and $W$ that are not necessarily adjacent, which represent processes. A given $R$ is uniquely identifiable by its edges.

Let $\mathcal{R}_{U,W}$ be a subgraph of $\overrightarrow{G}$ which is the set of potential routes between $U$ and $W$. Where $\mathcal{R} = \emptyset$, we say that $U$ and $W$ are *disjoint* and no potential communication exists between them. Where $|\mathcal{R}| > 1$ we say that potential communication between $U$ and $W$ is *redundant*. Where two routes, say $S$ and $R$ in $\mathcal{R}_U, W$ are not edge disjoint, we say that the potential routes $S$ and $R$ are *dependent*. Where two routes are edge disjoint, we say that the potential routes $S$ and $R$ are *independent*. We can extend these notions to redundancy, dependence and independence to the set of all subgraphs $\overrightarrow{G}(\mathcal{R})$ of $\overrightarrow{G}$.

A network operator $U$ is represented as a process in $\overrightarrow{G}$. Likewise a control unit $W_i$, belonging to a set of control units $\mathcal{W}$, is represented by a process in $\overrightarrow{G}$. We assume, during normal operations, that $U$ sends and receives message packets from each $W_i$ and that all other vertices $V_i \in \mathcal{V}$ act as forwarding links for these messages. We also assume that potential communication between $U$ and $\mathcal{W}$ is redundant for each $W_i \in \mathcal{W}$ [18] and that there exists planned non-utilisation of redundant channels and processes during normal network operations, hence they are available for other purposes such as contingent actions by the operator or other system components. We subsequently model network operations over the initial network model using an algebraic representation based on the $\pi$-calculus [17]. For example, let $U, V$ be a process and $C$ be a control system, we define $C$ as

$$\begin{aligned} V &:= \nu\, z\, y(u_{\langle \bar{c} \rangle}).f(u) \rightarrow \bar{x} z_{\langle \bar{c} \rangle} \oplus \lambda \\ U &:= \nu\, z[TRUE]\bar{y} a_{\langle \bar{c} \rangle} + x(z_{\langle \bar{c} \rangle}) \\ C &:= !U|!V. \end{aligned} \tag{1}$$

Period (".") indicates an ordered sequence, $+$ an arbitrary sequence, $\oplus$ represents a choice, $f() \rightarrow \nu z, z, \bar{x}z$ is a function which creates/updates a name and may send it, $\bar{x}z$ indicates the capability *send* the name $z$ by channel $x$, $x(z)$ indicates the capability *receive* the name $z$ by channel $x$ and (purely for convenience) the tuple $\langle \bar{c} \rangle$ is a set of characteristics associated with a name. $[TRUE]$ represents some condition which must be fulfilled before a capability is exercised. Processes may replicate $!P$ on exhausting their capabilities. We freely use $\prod$ and $\sum$ to deal with multiple concurrent processes

$P|Q|R|\ldots$ and sums of capabilities $\bar{x}_1 a + \bar{x}_2 a + \ldots$ respectively. Labels (e.g. $\lambda$) are used to indicate some inaction, which may or may not be observable (e.g., decision making, message loss). See [2] for a fuller discussion of this approach.

Modelling network operations enables us to specify which potential routes in the initial network model will be used during operations and under what conditions. This knowledge which can be captured economically using our algebraic specification which encapsulates both the network topology and operations over it and can consequently be used by operators for detection and recovery purposes as discussed in section 6.

# 5 Probabilistic Observation Protocol

We define the protocol and subsequently discuss complexity requirements in terms of number of observations required to locate a subverted node and space requirements.

## 5.1 Probabilistic Packet Observation Protocol

Let $V$ be any network node which is not an operator, nor a control unit. We can define the action of $V$ algebraically by

$$\begin{aligned} V :=\nu\, a\tilde{k}t\, &\sum x_i(u_{\langle r,f,d,0\rangle}).Observe(f,u) \rightarrow (\bar{x}w_{\langle r,0,d,a\rangle}.\mathbf{0} \oplus \lambda) \\ &.NewKey(t,k\tilde{k}) \rightarrow (\bar{x}k^{t-\delta}{}_{\langle r,0,0,0\rangle}.\mathbf{0} \oplus \lambda) \\ &+ \sum \bar{x}_j u_{\langle r,f,d,0\rangle} |!V \end{aligned} \tag{2}$$

where $a$ is the node address, $f$ is the packet flag, $d$ is the packet identity, $r$ is the address to which the packet is routed, $k$ is a key from the vector of keys $\tilde{k}$. $x_i$ and $x_j$ are sets of channels over which we sum the send and receive capabilities and $t$ is a time signal. Initially, for all packets, we set a flag $f = 0$ and observation is a null event. But we assume that the set $\mathcal{W}$ of control units – see section 4 – contains redundant sensors or control units which the operator can use for intrusion detection purposes by considering contextual information – i.e., determining whether a given signal from a control unit is consistent with other (redundant or related) signals causally linked to that unit's operation[3]. A break in context results in a detection event.

As the result of a detection event, the operator initiates the protocol by setting $f = 1$ and packets are subsequently identified by originating nodes (i.e., control units or the operator). On receiving flagged packets, whether inputs or outputs, routers determine based on some probability $q$ to "observe" identified packets. The $Observe()$ function copies the packet and hashes its data contents, the packet identity and its own IP address which it stores as a characteristic. Then it sends the copied packet to the operator. We note the potential for packets to be dropped by $\lambda$, but, in general, minor losses should not affect protocol operation. Finally, after a period of time indicated by $t$, routers publish the current key to the operator and generate a new key in such a way that subsequent (and previous, unrevealed) keys cannot be predicted, cf. section 5.4. This action is controlled by the function $NewKey()$ and it should be noted that the names used by this function for keys and time are restricted to the scope of the function (i.e., they are local variables). Hence these values can only be known to the adversary in agent nodes.

On receiving packets whose identities match, the operator can compare the data contents of copied packets with each other and the original, noting its origin and potential routing (based on the initial network model $\overrightarrow{G}$). Hence where packet contents differ due to adversary manipulation, it can be determined from several packets which routes may have been subverted by the adversary, simply by observing the hashed address of observed packets and whether or not they have been manipulated. A process of elimination over each route leads to a determination of which nodes have been subverted on which routes – see section 5.2. In addition, if the adversary forges packets, these are trivially detected because these have no corresponding original packet or because they are not hashed with the correct control unit hash key. Once sufficient subverted routes are identified (i.e., such that they can effectively be bypassed using redundant, unused channels), the operator can (other than for further detective purposes) re-direct traffic away from affected routes using a dynamic re-routing protocol [3].

Proof of the protocol's correctness requires reduction of the algebraic statement along with the construction of suitable algorithms for the functions, which we have only defined informally here. The method of subverted node location is outlined during a discussion of the message complexity requirements in section 5.2

### 5.2 Message Complexity

We now consider the average number of observation packets required for the detection of a subverted process. Let $R$ be a single route which is part of a larger subgraph of routes $\mathcal{R}$ and consider the detection of a single subverted process. We also limit ourselves to considering the manipulation of a single output signal[4]. First, we define the *observation range* of $R$:

**Definition 2.** ***Observation Range*** *On $R$, we label the process directly adjacent to the operator $S$ and the control unit $T$ respectively. Let $<$ be the relation "follows" in order of communication, e.g., for output signals, $S < T$. We require observation packets to be sent by the processes between $T$ and $S$, probabilistically. Hence we call the set of processes from $S$ to $T$, but not including $T$ and $S$, the* observation range *of $R$ and we write $Obs(R)$. If we determine on a new $S'$ or a new $T'$ then we have a new observation range for $R$ which is $Obs(R')$.*

For convenience, we label the processes in $Obs(R)$ which are not $S$ or $T$ as a set $V$ with members $\{V_1, V_2, \ldots, V_{n-1}\}$ in order of communication, e.g., $V_2 < V_1$. We also note that if the number of processes from $S$ to $T$ is $n$ then the number of processes in $Obs(R)$ is $n - 2$. If we can see mismatches between observation packets and original packets, we assume that, at least, process $S$ is producing a manipulated packet. We overload the process label $S$ to designate the set of processes which communicate invalid packets. Likewise, we assume that $T$ (for the moment) is producing valid packets

[3] This is achieved using another protocol which we do not define here. But, for example, TCP/IP and similar protocols have fields for setting required routes, though our results do not rely on having IP based protocols.

[4] Applying the equivalent argument to input commands originating with the operator is omitted here.

and overload the process label $T$ to designate the set of nodes which communicate valid packets. To find out how many processes are in $S$ and how many are in $T$ we need to collect information about every process in $R$. This problem is related to the card collector's problem which is encountered in IP traceback for detecting DOS attacks [10], but with strong efficiencies. These arise because each packet observation inside $Obs(R)$ may allow the elimination of a random number of packets in order of communication along $Obs(R)$. For example, if we observe a process $V_i < T$ and $V_i \in T$ of valid processes then all processes $V_i < V_{i-1} < V_{i-2} < \ldots < T$ may likewise be assumed to be valid. Similarly, if $V_i \in S$ all subsequent processes in order of communication $\ldots < V_{i+1} < V_i$ are assumed to be in $S$. Hence we can designate the process $V_i$ to be the new endpoint for the search (either the new $S'$ or $T'$). This means that the next valid observation operation will take place in the new *observation range* $Obs(R')$.

We assume the observation probability $p$ for each process is uniform. Let $X_i$ be a random variable which is the number of observations required to observe a packet in $Obs(R)_i$ where $i = 1, 2, 3, \ldots$ is the number of previous observations inside each successive $Obs(R)$. On each observation, the number of processes in $Obs(R)$ shrinks by a random amount $d_i$. Since the probability of a process making an observation is uniform, it follows that the (independent) probability of making an observation inside $Obs(R)_i$ is $\frac{n-\sum_i d_i}{n}$ [5]. Let $Y_j$ be the number of observations required to acquire total knowledge of the processes in $R_i \in \mathcal{R}$ then the expectation of $Y_J$ which is $E(Y_j)$ is calculated as shown in equation 3 since the observation range shrinks randomly by a distance of $d_i$ nodes for which a determination has been made on each observation and $d_0 = 2$ and the sum $\sum_j d_j = n - 1$ where $j \geq 0$ (because we exclude $S$ and $T$ from $Obs(R)$).

$$\begin{aligned} E(Y_j) &= \sum_i E(X_i) \\ &= \frac{n}{n-d_0} + \frac{n}{n-(d_0+d_1)} + \frac{n}{n-(d_0+d_1+d_2)} + \ldots + \frac{n}{n-(\sum_j d_j)} \end{aligned} \tag{3}$$

The sum $\sum_j' d_j' = n - 3$ where is $j' > 0$ (i.e., excluding the constant $d_0 = 2$) implies that we have positive integer solutions $d_1 + d_2 + \ldots + d_r = n - 3$ for each $r \in \{1..n-3\}$. Hence there are $Q = \sum_{r=1}^{n-3} \binom{n-4}{r-1}$ possible sums each of which is equally likely to occur. Let $Y$ be a random variable which indicates the average total number of packets required to take complete observation of the set of nodes $R$, then we have

$$E(Y) = \sum_{j=1}^{Q} \frac{1}{Q} \left[E(Y_j)\right] \tag{4}$$

Considering detection for all routes $R_i$ in a subgraph $\mathcal{R}$, further efficiencies are gained where routes are dependent since node validity may be determined along several routes simultaneously, hence the largest average number of packets $\sum_{k=1}^{m} [E(Y)_{max}]_k$

[5] The number of actual packets observed will be a ratio $\frac{1}{p}$ of the observation probability.

needed for observation occur when each route $R_i \in \mathcal{R}$ is independent. This finding has implications for the design of SCADA systems which we discuss in section 6.

### 5.3 Space and Storage Requirements

An important advantage of probabilistic marking protocols is the constant space requirement for packet header size and that storage requirements on routers were not overburdened [13]. These advantages are retained for this protocol for recording packet identity and marking process, but with the overhead of creating a new packet on each observation, similar to the iTrace protocol. Further economies can be gained in space requirements by utilising a knowledge of the network topology – cf. [14].

Online storage requirements, aside from space required for storing hash value key chains (see section 5.4), are assumed to be met by the operational control centre capacity and not to affect network node storage. Storage requirements are anyway minimal set by the observation probability where $X$ is the number of packets per route (or routing subgraph) required before an observation takes place and $E(X) = \frac{1}{p}$. Once an observation has taken place, unobserved packets can be discarded and observed packets archived for forensic purposes.

### 5.4 Time-Sequenced Key Value Release

A network node $V$ generates an initial sequence of keys $\tilde{k}$ and subsequently generates a fresh key based on a nonce using a randomly selected key from its key chain, using a suitable one-way function. The fresh key becomes part of its key chain, randomly replacing another key, and is used for packet marking from that point. Depending on some time period, or set of discrete events (e.g., observing $n$ packets), the network node generates another fresh key by hashing a randomly selected key from its key chain with the current key and replacing one of the keys, again at random. It subsequently publishes the previously used key to the operator. Subsequent keys are released in the order in which they are used. This scheme is similar to the one proposed in [14]. This approach is resilient to attack since insufficient time exists for guessing keys while both the packet identity and its contents are hidden from the attacker c.f. [19], hence packets cannot be directly manipulated and may only be arbitrarily delayed, dropped, re-routed or vandalised. However, these forms of attack only delay rather than disrupt discovery. In some cases, they may accelerate it by providing further packet-based anomalies, hence detracting from agent capability to conceal their presence using protocol forgery. The approach is also low cost in computational terms because the rapid turnover in keys means even a weak encryption method suffices to implement this approach.

## 6 Implications for SCADA System Design

We have a proposed a low cost approach to the location of subverted network nodes in SCADA networks under attack, assuming a detection event. The location method has obvious efficiencies in detective terms over other packet tracing methods and minimises space complexity and storage requirements. Overheads derive from the requirement to create additional packets and the use of a dynamic encryption protocol, but these are kept to a minimum. The proposed approach has implications for SCADA system design. It is trivial using the algebraic definition of the system to enumerate both potential and utilized routes for any subgraph $\mathcal{R}$ which is a set of routes on the system. Hence the operator can determine an operational solution which switches the utilization of potential routes to bypass subverted. However, this does not require the (costly) creation of multiple independent routes (i.e. lengthwise edge-disjoint paths) between an operator and a control unit. Instead it suggests a design based on creating railway-like “junction switches” along SCADA network routes to allow traffic to be diverted to alternative, currently unused routes at multiple points along the “track”. This opens an area of research into efficient means of dynamic re-routing, including route calculation, in such networks under attack circumstances.

There are also other approaches to utilising this approach to be explored. For example, instead of observing single nodes, we could use edge observation, probabilistically observing two adjacent nodes at a time. The use of dynamic re-routing combined with deterministic observation might also, under certain circmstances, prove more efficient than probabilistic observation. Finally, we could also partially implement probabilistic (or deterministic) node observation on certain nodes, utilizing the protocol as a detection method in its own right, which we aim to investigate in subsequent work.

## 7 Conclusion and Future Work

In this paper we have proposed a novel, low cost protocol for use on SCADA systems which enables the operator to observe packet behavior with regard to content and routing which will allow us to locate and counter such subverted nodes on known or partially known topologies with compromising adversaries and agents. The protocol is based on previous work in IP Traceback, originally used for detection and prevention of DOS attacks. Combined with the ability to route packets dynamically, this approach represents a cost-effective approach to defeating node based attacks from an engineering point of view by making use of system redundancy and planned underutilization. Future work will consist of determining, for various SCADA protocols, practical implementations of this approach and testing them by simulation with regards to performance. Further work remains to be done on other related protocols or variants and extending the work to other kinds of network. We will also consider further options for the utilization of this protocol in the context of SCADA systems.

## References

1. Chen, T.M., Abu-Nimeh, S.: Lessons from Stuxnet. IEEE Computer 44(4), 91–93 (2011)
2. McEvoy, T.R., Wolthusen, S.: A Plant-Wide Industrial Process Control Security Problem. In: Butts, J., Shenoi, S. (eds.) Critical Infrastructure Protection V. IFIP AICT, vol. 367, pp. 47–56. Springer, Heidelberg (2011)
3. McEvoy, T.R., Wolthusen, S.D.: A Formal Adversary Capability Model for SCADA Environments. In: Xenakis, C., Wolthusen, S. (eds.) CRITIS 2010. LNCS, vol. 6712, pp. 93–103. Springer, Heidelberg (2011)

4. Verba, J., Milvich, M.: Idaho National Laboratory Supervisory Control and Data Acquisition Intrusion Detection System (SCADA IDS). In: IEEE Conference on Technologies for Homeland Security, pp. 469–473 (2008)
5. Gamez, D., Nadjm-tehrani, S., Bigham, J., Balducelli, C., Burbeck, K., Chyssler, T.: Safeguarding Critical Infrastructures. In: Dependable Computing Systems: Paradigms, Performance Issues, and Applications. Wiley[Imprint], Inc. (2000)
6. McEvoy, T.R., Wolthusen, S.D.: Trouble Brewing: Using Observations of Invariant Behavior to Detect Malicious Agency in Distributed Control Systems. In: Rome, E., Bloomfield, R. (eds.) CRITIS 2009. LNCS, vol. 6027, pp. 62–72. Springer, Heidelberg (2010)
7. Svendsen, N., Wolthusen, S.: Using Physical Models for Anomaly Detection in Control Systems. In: Palmer, C., Shenoi, S. (eds.) Critical Infrastructure Protection III. IFIP AICT, vol. 311, pp. 139–149. Springer, Heidelberg (2009)
8. Sheng, S., Chan, W., Li, K., Xianzhong, D., Xiangjun, Z.: Context Information-based Cyber Security Defense of Protection System. IEEE Transactions on Power Delivery 22(3), 1477–1481 (2007)
9. Al-Duwairi, B., Govindarasu, M.: Novel Hybrid Schemes Employing Packet Marking and Logging for IP Traceback. IEEE Transactions on Parallel and Distributed Systems 17(5), 403–418 (2006)
10. Park, K., Lee, H.: On the Effectiveness of Probabilistic Packet Marking for IP Traceback Under Denial of Service Attack. In: INFOCOM 2001: Proceedings of the Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 1, pp. 338–347 (2001)
11. Benetti, D., Merro, M., Viganò, L.: Model Checking Ad Hoc Network Routing Protocols: ARAN vs. endairA. In: SEFM, pp. 191–202 (2010)
12. Dean, D., Franklin, M., Stubblefield, A.: An Algebraic Approach to IP Traceback. ACM Transactions on Information System Security 5, 119–137 (2002)
13. Savage, S., Wetherall, D., Karlin, A., Anderson, T.: Network Support for IP Traceback. IEEE/ACM Transactions on Networking 9(3), 226–237 (2001)
14. Song, D.X., Perrig, A.: Advanced and Authenticated Marking Schemes for IP Traceback. In: INFOCOM 2001: Proceedings of the Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 2, pp. 878–886 (2001)
15. Wong, T.Y., Wong, M.H., Lui, C.S.: A Precise Termination Condition of the Probabilistic Packet Marking Algorithm. IEEE Transactions on Dependable and Secure Computing 5(1), 6–21 (2008)
16. Simon, D.: Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches, 1. auflage edn. Wiley & Sons (August 2006)
17. Sangiorgi, D., Walker, D.: $\pi$-Calculus: A Theory of Mobile Processes. Cambridge University Press, New York (2001)
18. Cardenas, A.A., Roosta, T., Sastry, S.: Rethinking Security Properties, Threat Models, and the Design Space in Sensor Networks: A Case Study in SCADA Systems. Ad Hoc Networks 7(8), 1434–1447 (2009), Privacy and Security in Wireless Sensor and Ad Hoc Networks
19. Ye, F., Yang, H., Liu, Z.: Catching "Moles" in Sensor Networks. In: ICDCS, p. 69 (2007)

# Sub-optimal Topological Protection Strategy from Advanced Malware

Andrea Arbore and Vincenzo Fioriti

AIIC, Via Isonzo 32, 00198 Roma, Italy
arborean@hotmail.com, vincenzo.fioriti@gmail.com

**Abstract.** The spreading of dangerous malware in inter-dependent networks of electronics devices has raised deep concern, because from the ICT networks infections may propagate to other Critical Infrastructures producing the well-known domino effect. Researchers are attempting to develop a high level analysis of malware propagation, discarding software details, in order to generalize to the maximum extent the defensive strategies. It has been suggested that the maximum eigenvalue could act as a threshold for the malware's spreading. In this paper we study the Italian Internet Autonomous System simulating the diffusion of a worm, verifying the theoretical threshold and showing how to choose in a sub-optimal way the set of most influential nodes to protect with respect to the spectral paradigm. Our algorithm is fast and outperforms measures as degree, closeness, betweenness, and dynamical importance.

**Keywords:** malware, epidemic spreading, threshold, interdependencies PACS number: 9.75.Fb - Structures and organization in complex systems.

## 1 Introduction

The malicious software (mal-ware) is a program code designed to produce undesired effects on a computer. Once malware was specialized: viruses, worms, trojan horses, spyware, backdoors, were among the most common forms of malware. Today, the trend is reversed toward an unification of these different dangerous codes and towards a very high technical level. The usage of zero-days attacks (exploits targeting specific vulnerabilities for which a software update is not yet available), botnets (nets of slave PC), net-awareness (smart malware exploiting the network features), make the protection of large computer networks a major problem. Another issue is the capability to influence not only the ICT network but also various Critical Infrastructures dependent from ICT [4,5,6,7,9,11,13]. In order to obtain such a result, there are basically two strategies: the targeted intrusion and the cooperative search. The first foresees a direct conventional approach to the actual target, while the second one demands a distributed control system, a complex communication scheme and a consensus-like decision making process. As a side effect of the cooperative search, the malware will spread in the network like a disease (the "epidemic" spreading). Actually, any kind of worm follows the epidemic spreading, but a standard worm will attempt to invade

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 81–92, 2013.

the maximum number of machines as quickly as possible, instead a sophisticated malware adopting a cooperative search strategy or even a simpler network aware strategy, will infect (relatively) few machines during a long period of time. In any case, both seem to propagate following the epidemic spreading model, at least during the initial phase of the attack. Thus, understanding this model may help in countering the spreading at the very beginning of it, when the cost of the defence is more affordable. In this paper we will apply important epidemiologic results to the Italian Internet Autonomous System (AS) and propose a sub-optimal defensive strategy. Important results on the threshold to the spreading are those of Pastor-Satorras and Vespignani [3] for the scale-free graphs, and subsequently by Wang et al. [1] and Chakrabarti et al. [2] for a generic graph. "Generic graph" means no assumption is made on the graph (scale-free, random, small-word, degree distribution, etc) or on the modelling Susceptible-Infected-Susceptible, Susceptible-Infected-Refractory, Infected-Refractory, etc (SIS, SIR, IR). The threshold is related to two parameters, namely the infection rate $\beta$ (average number of machines that can be infected per time unit by an already infected machine) and the cure rate $\delta$ (average number of machines that can be restored per time unit). Above the threshold the malware will propagate, otherwise it will end quickly. In the epidemic approach the design details of the malicious code are discarded or simply represented by the infection rate intended as a probabilistic parameter, in order to guarantee a more general analysis. We test these claims when $\beta$ and $\delta$ vary over links and nodes in a real graph and introduce the AV11 algorithm to choose the most influential subset of nodes with respect to the maximum eigenvalue. Note that we need to pick this subset as a unique group, thus we have to face a NP-complete knapsack problem.

The rest of the paper is organized as follows: Section 2 gives a survey on the epidemic threshold in the Peng's framework. Section 3 introduces the Italian Internet Autonomous System (AS) and the section 4 reports the spreading simulations modeling AS net as a Markov chain. Section 5 gives a brief discussion of the results. We propose the AV11 algorithm in section 6 and testing it on simple graphs in the section 7. Section 8 gives the conclusions. The Appendix A deals the AV11 principles and its computational issues.

## 2 Determining the Threshold

We will sketch the calculation of the threshold using the Peng's framework. Peng et al. [7] have provided an analytical treatment when $\beta$ and $\delta$ vary but have tested their claim only on a random Erdős-Rényi artificial graph. Here we outline briefly the formalism to derive the stability conditions for the dynamic system of the spreading (for details see [7]). The homogeneous models [1,2,3] assume that every machine has equal contact to others in the population, thus the infection and cure rates are constant; instead, in this paper we consider a different infection rate for each link $\beta_{ij}$ and a different cure rate for each node $\delta_i$ of the (directed or undirected) graph $G$ representing the Italian AS network. $\beta_{ij}$ and $\delta_i$ are extracted from a uniform distribution. The first step [7] is the modified adjacency matrix $\mathbf{M}$ obtained from the standard adjacency matrix $\mathbf{A} = (a_{ij})$ of the undirected graph $G$, whose entries $m_{ij}$ are modified according to:

$$m_{ij} = \begin{cases} a_{ij}\beta_{ij} & \text{if } i \neq j, \qquad 0 \leq \beta_{ij} \leq 1 \\ 1 - \delta_i & \text{if } i = j, \qquad 0 \leq \delta_i \leq 1 \end{cases} \tag{1}$$

Note that we allow $G$ to be directed, i.e. $\beta_{ij} \neq \beta_{ji}$, with no loss of generality. The difference system representing the infection dynamics on $G$ is:

$$p_i\left(t\right) = 1 - \prod_k \left(1 - m_{ik} \cdot p_k\left(t-1\right)\right) \,, \quad i,k = 1,\ldots,N \tag{2}$$

where $p_i\left(t\right)$ is the probability that the node $i$ at the discrete time $t$ is infected from the node $k$, $N$ is the number of nodes. Note that the $p_k\left(t-1\right)$ should be mutually independent; if this is not the case, the threshold value cannot be calculated exactly within the Peng's framework [7]. Now, since

$$1 - \prod_k \left(1 - m_{ik} \cdot p_k\left(t-1\right)\right) \leq \prod_k \left(m_{ik} \cdot p_k\left(t-1\right)\right),$$

the system (2) converges to zero if the difference system (3)

$$p_i\left(t\right) = \prod_k \left(m_{ik} \cdot p_k\left(t-1\right)\right) \,, \tag{3}$$

converges to zero. On the other hand, if (3) converges to 0, it can be proved that also (2) converges to 0. In compact notation the (3) is:

$$\mathbf{P}\left(t\right) = \mathbf{M} \cdot \mathbf{P}\left(t-1\right) \tag{4}$$

The system (3) is stable iff the largest eigenvalue of $\mathbf{M}$ is [9]:

$$\lambda_{\mathbf{M}} < 1$$

In that case

$$\lim_{t\to\infty} \mathbf{P}\left(t\right) = \mathbf{0}$$

or

$$\lim_{t\to\infty} p_i\left(t\right) = 0 \,, \quad i = 1,\ldots,N$$

and the epidemic spreading disappears. Since $\mathbf{M}$ is non-negative, its largest eigenvalue is a positive real number and the analytical threshold can be set to

$$\lambda_{\mathbf{M}}^{thr} = 1 \tag{5}$$

Anyway, in this paper we do not use this threshold due to the unrealistic [7] independence assumptions for (2) when the size of the graph is small; the (5) should be regarded as a lower bound of the actual threshold. Note that (5) says nothing about the actual spreading above the threshold: it states only a stop below the threshold. The significance of (5) is that the spreading depends on graph topology: adding or eliminating nodes/links affects strongly the diffusion of malware; on the other hand, the largest eigenvalue may be used to estimate/improve the network resilience modifying properly the topology [8].

## 3 The Italian AS Network

In this section we present the graph of the Italian internet AS network [10]. An internet Autonomous System (AS) is a set of connected Internet Protocol routing prefixes, responding to one operator and presenting a common defined routing policy. Our net is of $N = 611$ nodes, 5974 links, maximum degree 212, average degree 9.8; in Figures 1, 2 the node degrees and their histogram are shown.

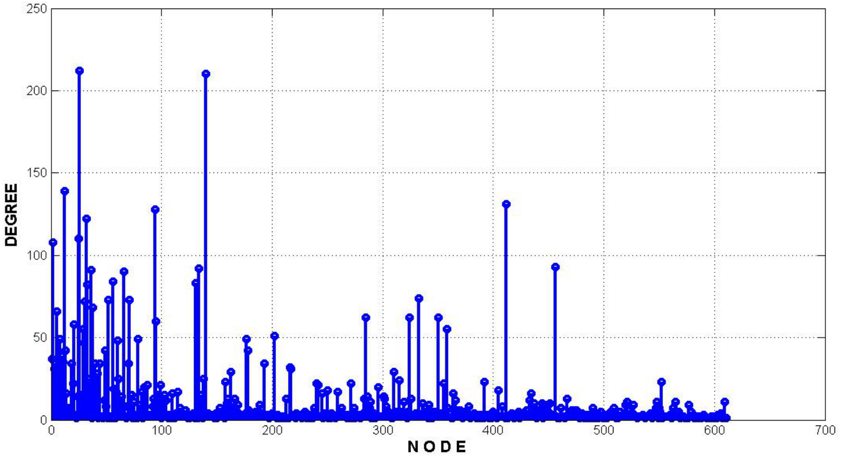


**Fig. 1.** Node degrees (not sorted)

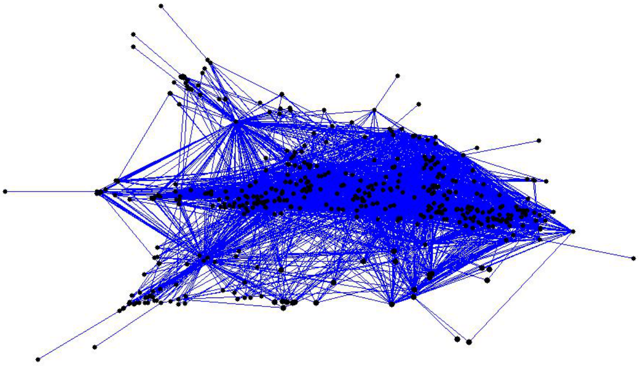


**Fig. 2.** Degrees histogram

In Figure 3 the Italian internet AS according to the Fruchterman-Rheingold graph drawing algorithm [12] is shown (the algorithm tries to minimize crossing links). Clearly the network is changing continuously, thus these numbers are only estimations. The maximum eigenvalue of the adjacency matrix $\mathbf{A}$ is $\lambda_{\mathbf{A}} = 41.5$,

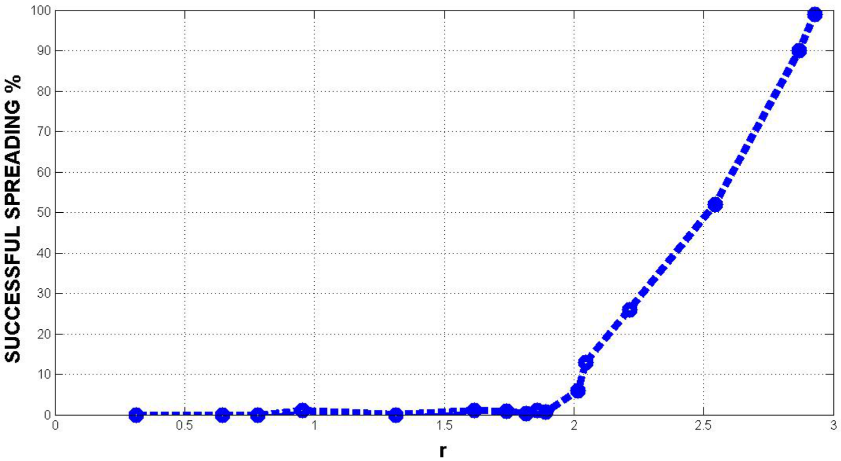


**Fig. 3.** The Italian internet AS network visualized by a Fruchterman-Rheingold force-directed algorithm

suggesting very roughly that the network is well connected (for a comparison, a poorly connected graph has $\lambda_{\mathbf{A}} < 10$), thus is able to transfer easily both information and infections [8].

## 4 Simulation of the Spreading

In our simulation we have considered for the infection rate matrix $(\beta_{ij})$ a range of values $\left[10^{-6}, 0.98\right]$ and for the cure rate vector $(\delta_i)$ a range $\left[10^{-5}, 0.012\right]$ from a uniform distribution (see Figure 4). Instead of integrating the difference equations (4) we simulate the node interactions in the most general case of spreading (no assumptions of SIS, SIR, SI) on a directed graph, by means of a Markov chain. The only assumption is that the model parameters are time-invariant, but this actually is not a limitation (a discussion on the time-variant issue will be reported elsewhere).

We consider successful a spreading if at least the 85% of the nodes have been infected. The rates $\beta_{ij}$ and $\delta_i$ are drawn from an uniform distribution and change at every run; they are represented by the ratio $r = \overline{\delta_i}/\overline{\beta_{ij}}$. Initially the spreading starts from 3 nodes selected randomly[1]; any infected node try to infect the neighbors through its links with probability $\beta_{ij}$ and at the same time, a node may be cured with probability $\delta_i$. The output of the calculation is the percentage of successful spreading over 100 runs for each value of $r$. $r_{thr} \approx 2$ may be considered an empirical threshold; in fact, the control parameter $r$ is based on two means, thus, as a consequence, the success percentage may vary as much as the 20%, according to our tests. One could speculate that in the thermodynamic limit this variability disappears but no evidence is available. Nevertheless, the

[1] The number of initially infected nodes not affect the equilibrium of the propagation.[1]

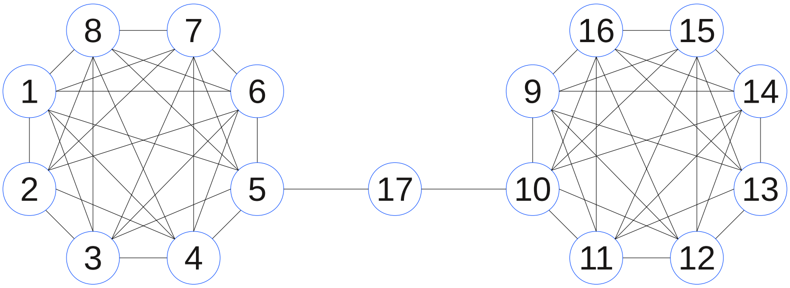


**Fig. 4.** The percentage of successful spreading and the control parameter $r = \overline{\beta_{ij}}/\overline{\delta_i}$. In $r = 2$ there is a significant number of successful spreading (every percentage has been averaged over 100 runs). At the start, the cure rate is low ($< 0.002$) and the infection rate high ($> 0.3$) in order to simulate the first steps of the attack.

analytical result (5) and our tests suggest that the control parameter $r$ is an indicator to predict the spreading power for any given $(\beta_{ij})$ and $(\delta_i)$.

## 5 Discussion

Beyond the empirical threshold $r_{thr}$ the diffusion progresses almost linearly with $r$, thus is mandatory to stay to the left of $r_{thr}$. The definition of $r$ offers two alternatives to stop the spread: re-modulate $\delta_i$ here and there or concentrate the efforts only on a limited set of nodes, according to some cost function. On the other hand, if rising the cure rate is too much expensive, the third option is to work on the network's topology to reduce $\lambda_{\mathbf{M}}$. Moreover, the proper design of the maximum eigenvalue of matrix $\mathbf{M}$ (or directly of matrix $\mathbf{A}$) shapes the network' resilience and is a powerful tool to identify the most vulnerable nodes or links [8].

Our AV11 algorithm selects a subset of $k$ nodes in order to obtain a larger reduction of the maximum eigenvalue.

## 6 The AV11 Algorithm

For a given graph $G$, we want to find simultaneously the $k$ best nodes (the "budget") to immunize or remove, to make the remaining nodes more robust to the attack. Of course, following the spectral paradigm one could remove a set of $k$ nodes and find the decrease of the eigenvalue, but this brute-force strategy is impossible to use, even for small graphs, because of the huge number of combinations. The problem is NP-complete [14], thus we resort to our suboptimal algorithm AV11 to reduce complexity and calculation time as its estimated complexity is $O\left(kn^3 \log(n)\right)$. On the other hand, the simple strategy (calculating the eigenvalue for a single node at a time, rank the results and take the first

**Algorithm 1.** AV11 pseudocode

```
Input: the adjacency matrix $\mathbf{A}$ and an integer $0 < k < n$
Output: a set $S$ with $k$ nodes
Calculate eigenvalues of the adjacency matrix $\mathbf{A}$;
Print $\lambda_{k+1}$, the absolute minimum largest eigenvalue obtainable;
Initialize: $S$ to empty; $\mathbf{Z} = \mathbf{I}_n$; $\mathbf{D} = (1 - \lambda_n)\,\mathbf{I}_n$; $node = 0$;
for $i = 0$ to $k$ do
    $\mathbf{P} = (\mathbf{Z} \cdot \mathbf{A} \cdot \mathbf{Z} + \mathbf{D})^p$;
    let $node$ be the index of an optimal diagonal element of the matrix $\mathbf{P}$.
    add $node$ to $S$;
    set $\mathbf{Z}\,[node, node] = 0$;
end for;
return $S$.
```

$k$ nodes) does not guarantee good performances, and above all, is easily understood by the attacker. In fact the attacker running the same simple algorithm would get exactly the same information. AV11 avoids this problem because can choose different subsets giving the same eigenvalue reduction as explained in the next section. The AV11 pseudocode is given in Alg. 1. (see Appendix A for a justification):

## 7 Testing the AV11 Algorithm on Artificial Graphs

Here we show how the AV11 outperforms the standard centrality measurements as degree (DC), closeness (CC), betweenness (BC), and the dynamical importance (DI), i.e. the variation produced from a single removed node on the maximum eigenvalue [8]. Remember that DC,CC, BC, DI use the simple strategy. Let $\lambda_1$ the largest eigenvalue of the adjacency matrix and $\lambda_1'$ is the new largest eigenvalue after $k$ nodes have been removed/immunized producing a variation in $\lambda_1$.

We now analyze $G1$ (Figure 5) allowing the removal/immunization of just one node ($k = 1$) and of 3 nodes ($k = 3$).

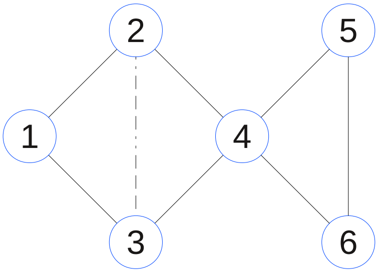


**Fig. 5.** The graph $G1$: node 17 is a bridge between two similar sub-graphs in which every node (except nodes 2 and 13) has the same degree, seven. Note that node 5 and 10 also have degree seven.

The largest eigenvalue of the graph $G1$ is $\lambda_1 = 6.303$.

$k = 1$ Removing/immunizing node 2 produces a lower eigenvalue: $\lambda_1' = 6.2877$; node 4: $\lambda_1' = 6.2875$ and node 17 : $\lambda_1' = 6.2715$. As pointed out in [2] the degree centrality in graph such as $G1$ is useless. Intuitively, node 17 looks like the most influential node and in fact offers a relevant reduction of the initial $\lambda_1$, see [2]. CC and DI select correctly node 17 while AV11 select a sub-optimal solution, thus CC and DI perform better, though the decrease of $\lambda_1$ is not dramatic. But things change when considering as soon as $k > 1$.

$k = 3$ Now we have $\binom{17}{3} = 680$ possible combinations of 3 nodes classified in 12 sub-optimal sets according to $\lambda_1'$ (the smaller, the better) by a brute-force method. The optimal subsets from the brute-force strategy are the following 36 different combinations of three nodes with the same eigenvalue:

**Table 1**

| Optimal subsets of nodes ($\lambda_1' = 5.7417$) | | | | | |
|---|---|---|---|---|---|
| $1, 9, 17$ | $3, 9, 17$ | $4, 9, 17$ | $6, 9, 17$ | $7, 9, 17$ | $8, 9, 17$ |
| $1, 11, 17$ | $3, 11, 17$ | $4, 11, 17$ | $6, 11, 17$ | $7, 11, 17$ | $8, 11, 17$ |
| $1, 12, 17$ | $3, 12, 17$ | $4, 12, 17$ | $6, 12, 17$ | $7, 12, 17$ | $8, 12, 17$ |
| $1, 13, 17$ | $3, 13, 17$ | $4, 13, 17$ | $6, 13, 17$ | $7, 13, 17$ | $8, 13, 17$ |
| $1, 15, 17$ | $3, 15, 17$ | $4, 15, 17$ | $6, 15, 17$ | $7, 15, 17$ | $8, 15, 17$ |
| $1, 16, 17$ | $3, 16, 17$ | $4, 16, 17$ | $6, 16, 17$ | $7, 16, 17$ | $8, 16, 17$ |

The results of the various algorithms are the followings:
AV11: subset $\{6, 15, 7\}$, producing $\lambda_1' = 5.7634$, contained in the 3rd suboptimal;
DI: subset $\{17, 5, 10\}$, $\lambda_1' = 6$, contained in the 5th suboptimal;
CC: subset $\{17, 5, 10\}$, $\lambda_1' = 6$, contained in the 5th suboptimal;
BC: subset $\{17, 5, 10\}$, $\lambda_1' = 6$, contained in the 5th suboptimal;

While in the optimal subsets node 17 is always present, nor node 5 neither node 10 are included. Intuitively, nodes $5, 17, 10$ would be the most probable human choice. DI and CC in this case have the same good performance but far from AV11. Remember that CC and DI select their nodes choosing the first $k$ single best results one by one, according to the new max eigenvalue and that AV11 selects the nodes as a unique subset. In this particular $G1$ case, AV11 does not choose the most important node 17, but nevertheless, its score outperforms CC and DI, indicating a counter-intuitive situation. Moreover, AV11 is faster than BC, CC and DI.

In the next example (Figure 6) we show that DC, BC, CC are not able to resolve the presence of the link 2-3.

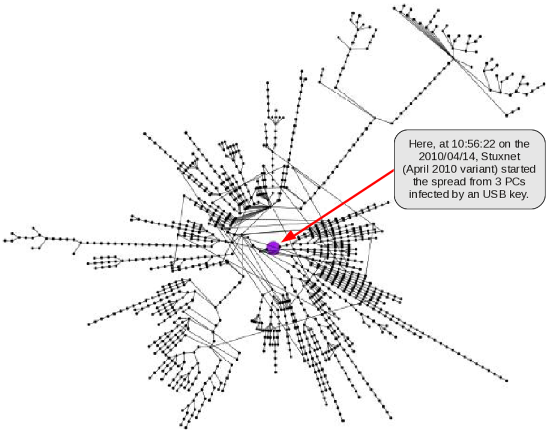

Here, at 10:56:22 on the 2010/04/14, Stuxnet (April 2010 variant) started the spread from 3 PCs infected by an USB key.

**Fig. 6.** The graph $G2$

With $k = 3$, we have compared the outcomes of the algorithms in two cases: when the link $2 \rightleftharpoons 3$ is present and when the link $2 \rightleftharpoons 3$ is absent.

The results are displayed in the following table:

**Table 2**

| | **Subsets** | |
|---|---|---|
| **Algorithms** | link $2 \rightleftharpoons 3$ present | link $2 \rightleftharpoons 3$ absent |
| Optimal | $\{1, 2, 4\}$, $\{2, 3, 4\}$ | $\{1, 4, 5\}$, $\{1, 4, 6\}$ |
| AV11 | $\{4, 2, 1\}$ | $\{4, 1, 5\}$ |
| DI | $\{4, 3, 2\}$ | $\{4, 6, 5\}$ |
| CC | $\{4, 2, 3\}$ | $\{4, 2, 3\}$ |
| BC | - | - |
| DC | $\{4, 3, 2\}$ | - |

Therefore, only AV11 and DI follow the change in the graph (note that AV11 captures in both cases the optimal subset).

## 8 Conclusions

Today the infrastructure protection and resilience is a major issue in some of the most important research programs in the world. In particular, the ICT infrastructure undergoes devastating attacks generated by malware and propagating to other Critical Infrastructures following a domino effect pattern. Software attack codes are becoming extremely sophisticated, difficult to detect or foresee and can adopt a network aware strategy or even an advanced cooperative target search. The epidemic spreading model seems suited to investigate key features as the existence of threshold phenomena by means of the spectral analysis of adjacency matrix of ICT networks. Although the simulations should be repeated on other real ICT networks, the tests suggest a transition from the absence of spreading to a sustained spreading, governed by a control parameter depending on the graph topology. Moreover, has been proposed a fast algorithm to identify some of the most important set of nodes to be immunized simultaneously as a suboptimal strategy. For these reasons, modelling the malware spread on graphs may result in a general purpose passive defence scheme, effective against a broad range of threats.

**Acknowledgements.** Authors gratefully thanks G. D'Agostino for useful suggestions.

## References

1. Wang, Y., Chakrabarti, D., Wang, C., Faloutsos, C.: Epidemic Spreading in Real Networks. In: SRDS Conference (2003)
2. Chakrabarti, D., Wang, Y., Wang, C., Leskovec, J., Faloutsos, C.: Epidemic Thresholds in Real Networks. ACM Trans. Inform. Syst. Secur. 10 (2008)
3. Pastor-Satorras, R., Vespignani, A.: Epidemic Spreading in Scale-free Networks. Phy. Rev. Lett. 86(14), 3200–3203 (2001)
4. Buldyrev, S., Parshani, R., Paul, G., Stanley, G., Havlin, S.: Catastrophic Cascade of Failures in Interdependent Networks. Nature 464 (2010)
5. Rinaldi, J., et al.: Identifying Critical Infrastructure Interdependencies. IEEE Control System Magazine 21, 337–351 (2001)
6. Osorio, L.: Seismic Response of Critical Interdependent Networks. Earthquake Eng. Struct. Dyn. 36, 285–293 (2007)
7. Peng, C., Xiaogang, J., Meixia, S.: Epidemic Threshold and Immunization on Generalized Networks. Physica A 389, 549–560 (2010)
8. Fioriti, V., D'Agostino, G., Bologna, S.: On Modeling and Measuring Interdependencies among Critical Infrastructures. In: COMPENG 2010 IEEE Conference, Roma (2010)
9. Harris, C., Miles, J.: Stability of Linear Systems. Science and Engineering, vol. 153. Academic Press, London (1980)
10. Courtesy of E. Gregori and coworkers, CNR Pisa
11. Zesheng, C., Chuanyi, J.: Measuring Network Aware Worm Spreading Strategy. In: INFOCOM 26th IEEE International Conference on Computer Communications. IEEE (2007)
12. Fruchterman, T., Reingold, E.: Graph Drawing by Force-Directed Placement. Software Practice & Experience 21, 1129 (1999)
13. Chen, Z., Ji, C.: Measuring Network-Aware Worm Spreading Ability. In: IEEE INFOCOM Conference 2007 (2007)
14. Arulselvan, A., Commander, C.W., Elefteriadou, L., Pardalos, P.M.: Detecting Critical Nodes in Sparse Graphs. Comput. Oper. Res. 36(7), 2193–2200 (2009)

## Appendix A

Because an adjacency matrix is symmetric, its eigenvalues are real. Let $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_j \geq \lambda_{j+1} \geq ... \geq \lambda_{n-1} \geq \lambda_n$ be the $n$ eigenvalues of $\mathbf{A}$. We can apply the Separation Theorem for the Hermitian operators on an $n$-dimensional space. From this classical theorem, follows that if $\mathbf{A}_r$ is a principal submatrix of $\mathbf{A}$ of order $r$ with eigenvalues $\alpha_1 \geq \alpha_2 \geq ... \geq \alpha_{r-1} \geq \alpha_r$ then $\lambda_j \geq \alpha_j \geq \lambda_{n-r+j}$. Therefore, removing $k$ nodes from the network we get a $n-k$ principal submatrix whose eigenvalues are localized in $\lambda_j \geq \alpha_j \geq \lambda_{n-(n-k)+j} = \lambda_{k+j}$. For the largest eigenvalue of $\mathbf{A}_r$, we obtain $\lambda_1 \geq \alpha_1 \geq \lambda_{k+1}$. Thus $\lambda_{k+1}$ is the absolute minimum largest eigenvalue we can obtain deleting $k$ nodes from the network. It is well known that the largest eigenvalue may be overestimated with the power method. The basics of the method are the following: at iteration $h$, the left and right products by $Z = f\left(h\right)$ reset $h$ rows and $h$ colomns of $\mathbf{A}$ ($h$ nodes removed). The result $\mathbf{A}\left(h\right)$ is the $n \times n$ adjacency matrix of the graph with $h$ nodes isolated (or immunized). The positive diagonal shift by $\mathbf{D} = \left(1 - \lambda_n\right)\mathbf{I}_n$ guarantees that every eigenvalue $\mu_{j(h)} = d + \lambda_{j(h)}$ is positive, where $\lambda_{j(h)}$ is the $j$th eigenvalue of $A_{(h)}$. Now, the trace of the power

$$\left(\mathbf{Z} \cdot \mathbf{A} \cdot \mathbf{Z} + \mathbf{D}\right)^p = \left(\mathbf{A}_{(h)} + d \cdot \mathbf{I}_n\right)^p = \left(b_{ij(h)}\right)$$

is known to satisfy

$$\text{Trace}\left(\mathbf{A}_{(h)} + d \cdot \mathbf{I}_n\right)^p = \sum_{i=1}^{n} b_{ii(h)} = \sum_{j=1}^{n} \left(d + \lambda_{j(h)}\right)^p$$

Therefore, from

$$\left(d + \lambda_{1(h)}\right)^p \left[1 + \sum_{j=2}^{n} \left(\left(d + \lambda_{j(h)}\right) / \left(d + \lambda_{1(h)}\right)\right)^p\right] = \text{Trace}\left(\mathbf{A}_{(h)} + d \cdot \mathbf{I}_n\right)^p$$

follows that

$$\left(d + \lambda_{1(h)}\right) \left[1 + \sum_{j=2}^{n} \left(\left(d + \lambda_{j(h)}\right) / \left(d + \lambda_{1(h)}\right)\right)^p\right]^{\frac{1}{p}} = \left[\text{Trace}\left(\mathbf{A}_{(h)} + d \cdot \mathbf{I}_n\right)^p\right]^{1/p}$$

Being

$$0 < \left(d + \lambda_{j(h)}\right) / \left(d + \lambda_{1(h)}\right) < 1 \qquad j = 2, 3, ..., n\,,$$

when $p \rightarrow \infty$ we have

$$d + \lambda_{1(h)} \approx \left[\text{Trace}\left(\mathbf{A}_{(h)} + d \cdot \mathbf{I}_n\right)^p\right]^{1/p}$$

Here, we have interest for the following inequality:

$$d + \lambda_{1(h)} \leq \left[\text{Trace}\left(\mathbf{A}_{(h)} + d \cdot \mathbf{I}_n\right)^p\right]^{1/p} = \left(\sum_{i=1}^{n} b_{ii(h)}\right)^{1/p} \tag{6}$$

In words, 6 states that reducing diagonal elements $b_{ii(h)}$ of $\left(\mathbf{A}_{(h)} + d \cdot \mathbf{I}_n\right)^p$ forces the reduction of the largest eigenvalue $\lambda_{1(h)}$. Resetting the $i$th row and the $i$th colomn in $\mathbf{A}$, means to reset an eigenvalue $\lambda_{k(h)}$ and make $b_{ii(h)} = d^p$. Hence, at $h$ iteration there are $h$ diagonal elements such that $b_{ii(h)} = d^p$ and

$$d + \lambda_{1(h)} \leq \left( hd^p + \sum_{j=1}^{n-h} b_{i_j i_j (h)} \right)^{1/p}$$

The positive diagonal shift $\mathbf{D} = (1 - \lambda_n)\,\mathbf{I}_n$ allows to use any positive integer $p$ for the power. In fact, if $p$ is an odd number and some $d + \lambda_{j(h)}$ are negative, then the ratios $\left(d + \lambda_{j(h)}\right) / \left(d + \lambda_{1(h)}\right)$ are not all positive; this invalid the inequality. Generally it is enough that $p$ is some unities and smaller than $n$. In addition, for the Separation Theorem we have $d + \alpha_j > 0$ for every principal submatrix of $\mathbf{A}$. Thus, we can use the same value during all the iterations: every shifted adjacency matrix will be a positive definite matrix. The computational complexity of AV11 is $O\left(kn^3 \log(p)\right)$ as the power of a matrix is computed by cumulative multiplications. If the complexity for each product of two $n \times n$ matrices is $O\left(n^3\right)$, then we have to do $O\left(n^3 \log(p)\right)$ operations for $k$ iterations. Using more efficient algorithms to square matrix multiplication, such as Strassen's algorithm which is significantly efficient for matrices with dimensions $n > 100$, make AV11 even more efficient.

# The Insider Threat in Cloud Computing

Miltiadis Kandias, Nikos Virvilis, and Dimitris Gritzalis

Information Security & Critical Infrastructure Protection Research Laboratory
Dept. of Informatics, Athens University of Economics and Business, Greece
{dgrit,kandiasm,nvir}@aueb.gr

**Abstract.** Cloud computing is an emerging technology paradigm, enabling and facilitating the dynamic and versatile provision of computational resources and services. Even though the advantages offered by cloud computing are several, there still exist second thoughts on the security and privacy of the cloud services. Use of cloud services affects the security posture of organizations and critical infrastructures, therefore it is necessary that new threats and risks introduced by this new paradigm are clearly understood and mitigated. In this paper we focus on the insider threat in cloud computing, a topic which has not received research focus, as of now. We address the problem in a holistic way, differentiating between the two possible scenarios: a) defending against a malicious insider working for the cloud provider, and b) defending against an insider working for an organization which chooses to outsource parts or the whole IT infrastructure into the cloud. We identify the potential problems for each scenario and propose the appropriate countermeasures, in an effort to mitigate the problem.

**Keywords:** Cloud Computing, Insider Threat, Security Measures.

## 1 Introduction

Outsourcing is not a new idea in the business world. In ICT, only in the last years we have experienced a real major increase in the number of companies that decide to outsource their IT infrastructure. Cloud computing has been a major influence for this move. High scalability and flexibility, service on demand, and - more importantly - cost reduction, are some of the factors that make Cloud Computing so popular. In general, there are three basic service models of cloud computing, namely: (a) Software as a service (SaaS), where software is offered by a third party provider (i.e. online word processing tools, web content delivery services), (b) Platform as a service (PaaS), which facilitates the development of new applications using APIs deployed and configured remotely (i.e. Google App Engine), and (c) Infrastructure as service (IaaS) [4], which provides abstracted hardware and operating systems capabilities, mainly through virtualization (i.e. Amazon Elastic Cloud). Outsourcing inevitably affects the organization's risk profile and thus it is crucial that the new threats introduced by the use of cloud services can be identified and mitigated. Hence, the area of IT outsourcing becomes even more interesting when security is added to the equation,

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 93–103, 2013.

especially regarding critical infrastructures [5]. Furthermore, researchers have pointed towards the security issues that cloud computing may introduce to critical infrastructures [25] [26] [31]. Nonetheless, even if a critical infrastructure chooses not to utilize cloud computing services, it will still be indirectly affected due to the fact that it's employees will use of such technologies (web based email, online data storage, social networks, etc).

A severe threat, that modern information systems and critical infrastructures need to address, is the insider threat. In general, the insider threat is defined as a person who has the appropriate access rights to an information system and misuses his privileges [1] [2]. Characterization of an attacker as an insider is not straightforward. For example, an employee who has been fired decides to attack his former employer for revenge. Although her access rights (should) have been revoked, and she is not considered a legitimate user any more, if she still has access to the infrastructure (using a backdoor she has previously installed), she is still considered an insider threat.

Mitigation of this problem is often complicated, as an insider can focus on a variety of target systems and orchestrate his attack motivated by a number of reasons [3], from personal profit to narcissism. To make things worse, the insider usually has the privilege of time, so as to study the information system and deploy a serious attack, which is very difficult to predict and detect in due time.

In this paper we focus on the insider threat in cloud computing, the ways it manifests, and the challenges in addressing the problem. Then, we propose appropriate countermeasures in an effort to mitigate the problem. We present the insider threat holistically, analyzing two scenarios: a) the insider is from the side of the cloud provider (a malicious employee of the cloud provider), or b) the insider works for an organization, which has outsourced part of its infrastructure to the cloud.

The rest of the paper is organized in as follows: Section 2 describes related work on the insider threat, in general. In Section 3, we define the problem and analyze possible attack scenarios. We conclude and present ideas for future work in Section 4.

## 2 Related Works

The research community has focused on many aspects of cloud computing security, such as authentication and authorization, digital forensics, secure data storage, as well as legal challenges. However, the problem of the insider threat in the cloud has not yet received visible research focus. The traditional insider threat is being systematically studied for more than a decade [6]. It is considered a complex issue, and there are various approaches in order to mitigate it.

Psychology and Sociology are useful tools in the battle against the insider threat. They offer precious information about the motives and the process of a potential inside attack [1] [3]. An insider must be able to conduct the attack, then she must be motivated, and, finally, she must find an opportunity to deploy the attack (CMO Model) [7]. There are some fundamental factors that should be taken into account, such as introversion, social and personal frustrations, computer dependency and ethical flexibility. [3]. Evaluation of them could be based on custom made psychometric tests.

Detection of malevolent insiders is hard to accomplice. Some systems have been proposed to detect insider threat [8], some of them utilize proactive forensics [9], graph-based analysis [10], honeypots [11] and other methods. A useful tool in the process of insider detection is intrusion detection systems (IDS) [12] [21], as they can detect abnormal actions, packets with illegal content and deviations from normal user behavior. Another useful technique, used to mitigate the insider threat, is system call analysis [13], command sequences and windows usage events [14]. The techniques based on the usage habits of the users, namely the system calls analysis, belong to a larger family of techniques called "host-based user profiling", while intrusion detection systems and honeypots belong to the "network-based sensors" family [14] [30].

Malevolent insiders have legitimate access to the information system. Thus, they might be aware of the defensive mechanisms in place. As a result, they are expected to be careful and often manage to adequately avoid detection. Furthermore, insiders are able to achieve their goals without deploying an attack, as they could exploit their given access rights and the features of the information system. In order to address this issue, a robust and flexible insider threat prediction model seems promising.

Insider threat prediction attempts have used both user and usage profiling, in order to result in a possible differentiation in user's behavior. File system, memory, I/O and hardware monitoring has been proposed [15], in order to detect differentiation from the normal usage norms, along with similar metrics about user sophistication [16], which can be used to detect usage anomalies or even confirm the stated level of knowledge of the user according to [22]. Magklaras, et al. have presented the concept of an insider threat specification language that enables analytical description of the actions of a malevolent insider [17], so as to detect such actions and tie them to suspicious insiders. Other attempts are based on knowledge graphs [18], customized minimal attack trees [19], or are concentrating on predicting insiders in a subset of the information system, like the database system [20]. There are also approaches that take psychological, sociological and educational parameters under consideration, along with technological ones [22] [27] [32].

## 3 Insider Threat in Cloud Environments

Regardless of the technical and operational countermeasures deployed in an infrastructure, defending against accidental or malicious human actions is hard to do. The insider threat affects virtually every infrastructure and remains an open research issue for decades. As mentioned in section 2, there has been some research focusing on this problem, with respect to traditional IT infrastructure, though the manifestation of insider threat in cloud computing has not been adequately researched upon. Given the functional context of cloud computing, a malicious insider with access to cloud resources can cause significantly more damage to the organization. Furthermore, as the attack can affect a large number of cloud users, the impact of such attack will be significant.

In order to study the problem in a holistic manner, we suggest that it should be studied in two distinct contexts: (a). Insider threat in the cloud provider: Where the insider is a malicious employee working for the cloud provider, and (b). Insider threat in the cloud outsourcer: The insider is an employee of an organization which has outsourced part or whole of its infrastructure on the cloud.

## 3.1 Insider Threat in the Cloud Provider

This is the worst-case scenario for both cloud providers and cloud clients, i.e. a malicious system administrator working for the cloud provider. Because of her business role in the cloud provider, the insider can use her authorized user rights to access sensitive data. For example, an administrator responsible for performing regular backups of the systems where client resources are hosted (virtual machines, data stores), could exploit the fact that she has access to backups and thus exfiltrate sensitive user data. Detecting such indirect access to data, can be a challenging task.

Depending on the insider's motives, the result of such an attack in a cloud infrastructure will vary from data leakage to severe corruption of the affected systems and data. Either way, the business impact for the provider will be significant. All common cloud types (IaaS, PaaS, SaaS) are equally affected by insider attacks as long as the insider has (or can gain) access to the datacenters or cloud management systems.

Someone could argue that the aforementioned impact of an insider threat in the cloud is similar to the impact of an insider in the classic outsourcing paradigm. This is partially true, since the decision to outsource is coupled with an innate risk of exposing sensitive data to an outsider, though cloud computing differentiates due to the fact that it offers a holistic solution to outsourcing via IaaS and PaaS. Hence, cloud computing paradigm could be utilized in order to outsource vast parts of the infrastructure instead of specific services, such as web hosting or application hosting.

### 3.1.1 Countermeasures

Effective mitigation of the insider threat requires defense in depth and a large number of countermeasures, implemented by both cloud providers and clients.

Client side

- Confidentiality/Integrity

Even in IaaS, where clients have the most access to the cloud infrastructure (administrative access to the virtual operating system), cloud clients are unlikely to detect that someone has gained unauthorized access to their data using OS level security mechanisms like IDS/IPS. The reason is that an insider working for the cloud provider (e.g. a malicious administrator) has access to the physical infrastructure which is not controlled by the client.

Clients can make use of cryptographic techniques [28], in an effort to safeguard the confidentiality and integrity of their outsourced data. However, encryption is a practical solution mainly for bulk data storage, and specifically for static data. Storing data in encrypted form, and decrypting them every time they need to be accessed (a common technique), is not an adequately effective defense against an insider, as the decryption key has to be stored somewhere in the cloud too. Considering that insiders can have access to the physical servers and thus can gain access to physical memory used by the virtual systems of the clients, all encryption keys stored in memory could be obtained [25]. A robust solution to this problem is not storing the encryption keys in the cloud but perform data manipulation directly on encrypted data. A number of techniques have been proposed in an effort to address this problem [24] [26] [29].

However, the performance overhead of such techniques is usually so high, that makes them currently impractical for real world applications.

- Availability

When it comes to availability, the use of multiple datacenters, ideally in different regions, is the only efficient solution, assuming that the cloud provider will not face a global outage. Multiple providers offer such an option to their clients, including automatic switching to the backup datacenter, in case an instance in the primary data center fails. Such geo-redundancy protects the client as long as the malicious insider cannot interfere with multiple datacenters at the same time.

Provider Side

From the provider side, a wider range of techniques can be used for detection and mitigation of insider attacks:

- Separation of duties

Strict separation of duties for the provider employees and especially system administrators is one of the most effective mechanisms for limiting the potential damage of such attacks. The insider will only have specific access rights to the infrastructure, thus she will only be able to attack the systems she can access. It appears to be safe to assume that a determined attacker will attempt to either obtain access to restricted resources or elevate her current legitimate privileges. Nevertheless, such actions will increase the possibility of detection of the attacker.

- Logging

All user actions, and especially actions of power users, such as administrators, have to be extensively logged and audited. Apart from acting as a deterrent measure for potential attackers, it will also enable early detection of potentially malicious actions and shall help the organization to trace back the incident to the actual individual, who performed the attack.

- Legal binding

Legal binding can act as a deterrent measure against a potential attacker, as it can result to civil penalties. However, there are several open legal issues, due to the fact that a cloud infrastructure is usually supported by multiple data centers in different countries. As the cloud provider's infrastructure under attack might be in a different country than the physical location of the attacker, each legal or physical entity is subject to different frameworks of law and, thus, administration of justice becomes a complex issue [23].

- Insider detection models

Insider detection models are implemented in the provider's infrastructure in an effort to detect malicious employees can be very helpful tools for prediction and in-time detection of insider attacks [22]. The models are based on predicting malevolent actions, in order to intensify monitoring of suspicious users. Furthermore, there are advanced techniques of real time insider detection that could be used [8] [10-12] [13-15] [21].

**Table 1.** Countermeasures

| Countermeasures | Implemented by: |
|---|---|
| Cryptographic techniques | Client |
| Geo-redundant data centers | Client and Provider |
| Separation of duties | Provider |
| Logging and Auditing | Provider |
| Legal contracts | Provider |
| Insider detection models | Provider |

Client: Client side countermeasures, Provider: Provider site countermeasures.

### 3.2 Insider Threat in the Cloud Outsourcer

In the second scenario, the insider is an employee of an organization, which has moved part (or the whole) IT infrastructure into the cloud. Initially, this could be considered as a traditional insider problem. However, we argue that there are a number of worth-noticing differences:

*Detection models*: Providers can use insider detection models for detecting malicious employees. However, the use of such models by the cloud client(s), who have outsourced their IT infrastructure, is problematic. As a potentially malicious user is accessing the cloud infrastructure, a detection model will have to correlate data from both the cloud infrastructure and the user workstation. Furthermore, user profiling becomes harder, as the user behavior in the cloud has to be included as a model parameter, thus significant changes to existing models are required. On the other hand, cloud use - along with extensive logging of users' actions - could lead to useful data. This data could be used to further study and depict users' behavior, which might lead to better user profiling. As long as these models have not been applied or studied within the context of cloud computing, we can only speculate about the results. The worst case scenario is that the prediction system will conclude in so many false positives/negatives, that the results cannot be trusted. Therefore, for the time being the existing detection and prediction models and techniques are unable to operate in cloud infrastructures.

In Fig. 1 we visualize a number countermeasures of an early implementation model, in an effort to defend against malevolent insiders. Risk analysis should be obviously the first step before implementing such countermeasures, as depending on the risk profile of the organization, implementation of all countermeasures may be inappropriate and result in performance degradation and high administration costs.

*IDS/IPS*: Use of Intrusion Detection/ Prevention Systems (IDS/IPS), as a means of attack identification, is also problematic. Host based IDS/IPS can be used transparently in IaaS, as they usually require the installation of a software agent on the Operating System, which is under client's control. However, this is not an option in PaaS and SaaS, unless the cloud provider supports such mechanisms. Use of a traditional network IDS/IPS's is impossible, as no such systems can be installed physically to the cloud data center.

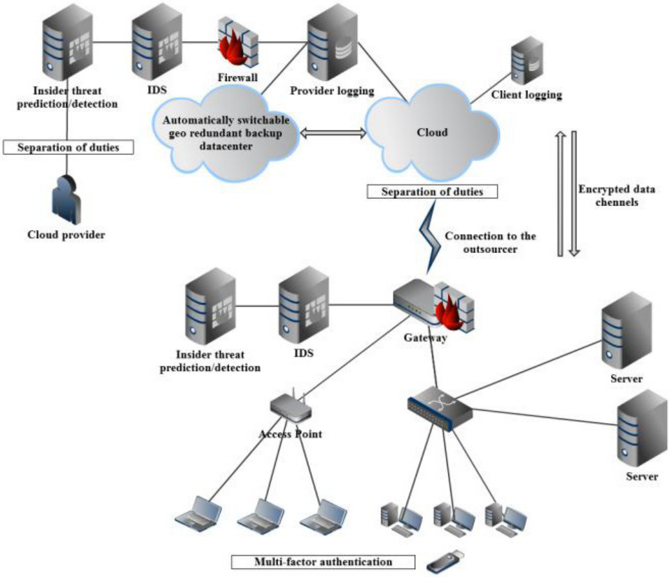

Insider threat prediction/detection
IDS
Firewall
Provider logging
Client logging
Automatically switchable geo redundant backup datacenter
Cloud
Separation of duties
Cloud provider
Separation of duties
Encrypted data chennels
Connection to the outsourcer
Insider threat prediction/detection
IDS
Gateway
Access Point
Server
Server
Multi-factor authentication

**Fig. 1.** Visualization of all security countermeasures versus insider threat

*Separation of duties*: In a traditional infrastructure there are well-defined user roles (system/network/database administrators, etc.) In the cloud, it is likely that the person who manages the cloud infrastructure (e.g. the virtual instances on an IaaS), is the same with the one that configures the firewall rules. This is evident to the users of A-mazon Elastic Cloud (EC2), where configuration about every aspect of the virtual infrastructure is done using a simple web-based dashboard.

*Attack origin identification*: Traditionally, an authorized user wishing to access the data center of her organization needs to go physically on-site, sign in, and use specific access credentials (i.e. RFID/smart card, PIN, etc.). Furthermore, she will need to have valid credentials for each system she wishes to access. Both her physical and digital presence will be logged and potentially monitored. This could be used to trace the origin of attack and act as evidence. VPN's may allow remote to the infrastructure remotely, but it is safe to assume that only a limited number of users will be granted remote access and strong authentication and monitoring will be in place. In comparison, gaining access to a virtual infrastructure on the cloud equals to getting access to the cloud console's access credentials used for managing the virtual infrastructure of the client. No physical evidence is available. The digital evidence will likely be the IP address, where the attacker logged in from. In the common case of shared credentials, identifying the individual who performed the attack can be challenging.

*Single point of failure*: One important point is the criticality of the cloud management console. Access to such console gives the user complete control over the virtual infrastructure, enabling her to create new virtual systems, modify existing ones, clone systems and destroy ("terminate", according to Amazon) virtual systems instantly. Termination equals to destruction of both the virtual machine instance (operating system, configuration options) and any data stored on it. This action is catastrophic and could lead to vast money loss, as well as damages to the infrastructure [3].

*Data leakage*: Data leakage attacks are easier to perform on a virtualized infrastructure. An attacker with access to the administrative console can exploit specific features of the virtual systems to her benefit, such as saving a snapshot of a particular system, or cloning it. Having acquired an image of the target system, she can modify it offline, circumvent the host's security mechanisms, and thus gain access to the data, while the original system will show no signs of intrusion.

### 3.2.1 Countermeasures

Provider Side

- *Anomaly detection*

From the provider's side, anomaly detection mechanisms could be used to identify abnormal behavior in client instances. The provider is then able to contact the client and inform her about the anomaly, so the client can investigate the issue. The more data input the provider has, the better the chances are for detecting potential issues. For example, if a SaaS provider identifies that a user account of a client is used for querying a large number of records in the database, while the same account was regularly making only few queries per day, then she should escalate the issue to the client for investigation. This requires the implementation of anomaly detection systems by the providers for monitoring client instances, which is not currently offered.

- *Separation of duties*

Strict separation of duties is an effective mechanism for limiting the impact of an insider attack. Cloud providers should implement robust identity and access management mechanisms and enable the cloud clients to create multiple accounts and multiple access rights for their users. By supporting multiple accounts, the client can enforce separation of duties, by giving only the required access rights to each employee according to her business role.

- *Multi factor authentication:*

Providers should support multi-factor authentication schemes in an effort to thwart phishing and password hijacking attacks against the cloud console management interface. Amazon EC2 is already supporting such mechanism, allowing clients to log in using certificates and OTP tokens.

Client Side

Clients outsourcing part(s) of their infrastructure to the cloud need to follow best practices and implement at least the same security measures implemented in their traditional infrastructure, such as System Hardening and in-time patch management. With concern to insider threat detection, the following measures are required:

- *Log Auditing*

Clients need to collect and audit all log files from their cloud systems, including any SaaS (assuming that providers offer such feature). Access logs are invaluable in helping detect in time, an attack.

- *Host based IDS/IPS*

Host based IDS/IPS's should be installed on all sensitive systems hosted in the cloud (IaaS), as they enable clients to detect in time ongoing attacks and at the same time maintain a low false-positive rate. Until cloud aware insider detection models are developed, IDS/IPS systems are some of the most effective measure for mitigating the insider threat.

**Table 2.** Countermeasures

| Countermeasure | Implemented by: |
|---|---|
| Identity and Access management | Client and Provider |
| Multi factor authentication | Client and Provider |
| Log analysis and auditing | Client |
| IDS/IPS | Client |
| Insider prediction/detection models | Client |

Client: Client side countermeasures, Provider: Provider site countermeasures.

## 4 Conclusion and Further Work

In this paper we dealt with the insider threat in the cloud environment. The insider threat is a well-known open research problem for decades, and - whilst in traditional IT infrastructures a set of adequate countermeasures has been proposed - this is not the case with cloud environments. An insider attack in the cloud is easier to perform and has far greater impact than an attack in a traditional infrastructure. At the same time, detection and identification of the physical entity that performed the attack remains challenging.

We identified two types of insider threat in cloud computing. The first is the one who works for the cloud provider. She could cause great deal of damage in both the provider and its customers. The second is the one who works for the organization that decides to outsource. We described and documented the differences between the traditional insider and the insider in cloud.

The paper has demonstrated the need for new insider prediction and detection models, to be used in the Cloud and be able to produce correct user profiling and, thus, avoid false estimations. In an effort to adequately address the problem, we proposed a number of countermeasures, for both the cloud clients and providers, for each insider scenario. These should be implemented in-line with the needs of each organization.

Furthermore, it appears that there is an emerging need for developing cloud transparent network and application intrusion detection systems or installation of traditional network IDS/IPS systems by the cloud provider on their infrastructure perimeter. Such systems can be offered as a service to their clients wishing to protect their outsourced infrastructure. Currently, no major cloud provider is offering such an option.

Our future work will focus on the implementation and analysis of insider threat prediction and detection models for the cloud, analysis of users' habits in the cloud and behavioral analysis of cloud usage along with ways for the providers to offer security as a service within the cloud. Furthermore, the implementation will enable us to measure the effectiveness of the countermeasures proposed in section 3 in practice, and enhance/modify them accordingly.

**Acknowledgments.** This work was supported in part by the S-Port (09SYN-72-650) project, funded by the Hellenic General Secretariat for Research & Technology, under the "Synergasia" Programme.

## References

[1] Theoharidou, M., Kokolakis, S., Karyda, M., Kiountouzis, E.: The insider threat to Information Systems and the effectiveness of ISO 17799. Computers & Security 24(6), 472–484 (2005)

[2] Bishop, M., Gates, C.: Defining the Insider Threat. In: Proc. of the 4th Annual Workshop on Cyber Security and Information Intelligence Research, Tennessee, vol. 288 (2008)

[3] Shaw, E., Ruby, K., Post, J.: The insider threat to information systems: The psychology of the dangerous insider. Security Awareness Bulletin 2, 1–10 (1988)

[4] Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: Above the Clouds: A Berkeley View of Cloud Computing. UCB/EECS-2009-28. Univ. of California at Berkley, USA (2009)

[5] Kandias, M., Mylonas, A., Theoharidou, M., Gritzalis, D.: Exploitation of auctions for outsourcing security-critical projects. In: Proc. of the 16th IEEE Symposium on Computers and Communications (ISCC 2011), Greece (2011)

[6] Anderson, J.: Computer security threat monitoring and surveillance. Technical Report, J. Anderson Company, Pennsylvania (1980)

[7] Schultz, E.: A framework for understanding and predicting insider attacks. Computers & Security 21(6), 526–531 (2002)

[8] Thompson, P.: Weak models for insider threat detection. In: Proc. of the Defense and Security Symposium, Florida (2004)

[9] Bradford, P., Hu, N.: A layered approach to insider threat detection and proactive forensics. In: Proc. of the 21st Annual Computer Security Applications Conference (2005)

[10] Eberle, W., Holder, L.: Insider threat detection using graph-based approaches. In: Proc. of the Cybersecurity Applications and Technology Conference for Homeland Security, pp. 237–241. IEEE Computer Society (2009)

[11] Spitzner, L.: Honeypots: Catching the insider threat. In: Proc. of the 19th Annual Computer Security Applications Conference, USA, (2003)

[12] Debar, H., Dacier, M., Wespi, A.: A Revised Taxonomy for Intrusion Detection Systems. Annales des Teecommunications 55(7-8), 361–378 (2000)

[13] Nguyen, N.T., Reiher, P.L., Kuenning, G.: Detecting Insider Threats by Monitoring System Call Activity. In: Proc. of the IEEE Workshop on Information Assurance, pp. 45–52 (2003)

[14] Salem, M., Hershkop, S., Stolfo, S.J.: A Survey of Insider Attack Detection Research. In: Insider Attack and Cyber Security, vol. 39, pp. 69–90 (2008)

[15] Magklaras, G., Furnell, S.: Insider Threat Prediction Tool: Evaluating the probability of IT misuse. Computers & Security 21(1), 62–73 (2002)

[16] Magklaras, G., Furnell, S.: A preliminary model of end user sophistication for insider threat prediction in it systems. Computers and Security 24, 371–380 (2005)

[17] Magklaras, G., Furnell, S.: Towards an insider threat prediction specification language. Information Management & Computer Security 14(4), 361–381 (2006)

[18] Althebyan, Q., Panda, B.: A knowledge-base model for insider threat prediction. In: Proc. of the IEEE Workshop on Information Assurance and Security, USA, pp. 239–246 (2007)

[19] Wang, H., Liu, S., Zhang, X.: A prediction model of insider threat based on multi-agent. In: Proc. of the 1st International Symposium on Pervasive Computing and Applications (2006)

[20] Yaseen, Q., Panda, B.: Knowledge Acquisition and Insider Threat Prediction in Relational Database Systems. In: Proc. of the International Workshop on Software Security Processes, Canada, pp. 450–455 (2009)

[21] Mun, H., Han, K., Yeun, C., Kim, K.: Yet another intrusion detection system against insider attacks. In: Proc. of the SCIS 2008 (2008)

[22] Kandias, M., Mylonas, A., Virvilis, N., Theoharidou, M., Gritzalis, D.: An Insider Threat Prediction Model. In: Katsikas, S., Lopez, J., Soriano, M. (eds.) TrustBus 2010. LNCS, vol. 6264, pp. 26–37. Springer, Heidelberg (2010)

[23] Parrilli, D.: Legal Issues in Grid and Cloud Computing. In: Stanoevska-Slabeva, K., Wozniak, T., Ristol, R. (eds.) Grid and Cloud Computing: A Business Perspective on Technology and Applications, pp. 97–118. Springer, Berlin (2010)

[24] Claessens, J., Preneel, B., Vandewalle, J. (How) can mobile agents do secure electronic transactions on untrusted hosts? A survey of the security issues and the current solutions. ACM Transactions on Internet Technology 3(1), 28–48 (2003)

[25] Johnson, C.: CyberSafety: On the Interactions between Cyber Security and the Software Engineering of Safety-Critical Systems

[26] Mather, T., Kumaraswamy, S., Latif, S.: Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance (Theory in Practice). O'Reilly Media, USA (2009)

[27] Gritzalis, D., Theoharidou, M., Kalimeri, E.: Towards an interdisciplinary information security education model. In: Proc. of the 4th World Conference on Information Security Education (WISE-4), Moscow (May 2005)

[28] Iliadis, J., Gritzalis, D., Spinellis, D., Preneel, B., Katsikas, S.: Evaluating certificate status information mechanisms. In: Proc. of the 7th ACM Computer and Communications Security Conference (CCS 2000), pp. 1–9. ACM Press (October 2000)

[29] Mylonas, A., Dritsas, S., Tsoumas, V., Gritzalis, D.: Smartphone Security Evaluation - The Malware Attack Case. In: Proc. of the 8th International Conference on Security and Cryptography (SECRYPT 2011), Spain, pp. 25–36 (July 2011)

[30] Lekkas, D., Gritzalis, D.: Long-term verifiability of healthcare records authenticity. International Journal of Medical Informatics 76(5-6), 442–448 (2006)

[31] Theoharidou, M., Kotzanikolaou, P., Gritzalis, D.: Risk-Based Criticality Analysis. In: Palmer, C., Shenoi, S. (eds.) Critical Infrastructure Protection III. IFIP AICT, vol. 311, pp. 35–49. Springer, Heidelberg (2009)

[32] Lambrinoudakis, C., Gritzalis, D., Tsoumas, V., Karyda, M., Ikonomopoulos, S.: Secure electronic voting: The current landscape. In: Gritzalis, D. (ed.) Secure Electronic Voting, pp. 110–122. Kluwer, USA (2003)

# Interdependencies between Critical Infrastructures: Analyzing the Risk of Cascading Effects

Panayiotis Kotzanikolaou[1], Marianthi Theoharidou[2], and Dimitris Gritzalis[2]

[1] Dept. of Informatics, University of Piraeus,
85 Karaoli & Dimitriou, GR-18534, Piraeus, Greece
pkotzani@unipi.gr

[2] Dept. of Informatics, Athens University of Economics & Business,
76 Patission Ave., GR-10434, Athens, Greece
{mtheohar,dgrit}@aueb.gr

**Abstract.** One of the most challenging problems, when protecting critical infrastructures, is the identification and assessment of interdependencies. In this paper we examine the possible cumulative effects of a single security incident on multiple infrastructures. Our method provides a way to identify threats that may appear insignificant when examining only first-order dependencies, but may have potentially significant impact if one adopts a more macroscopic view and assesses multi-order dependencies. Based on previous work, we utilize existing first-order dependency graphs, in order to assess the effect of a disruption to consequent infrastructures.

**Keywords:** Critical Infrastructure, Interdependencies, Risk, Cascading Effect.

## 1 Introduction

Protecting Critical Infrastructures (CI) poses challenges not only due to the significant social impact caused by disruption of their services, but also due to the high number of dependencies between them. The most important parameter that interdependencies may introduce is that they allow security incidents to escalate or cascade to different infrastructures, thus causing potentially significant impact to multiple types of sectors, individuals or countries. Motivating examples for this paper include the electric power disruptions in California (2001) [1], as well as the major blackouts in the US, Canada and Europe (2003) [2].

The electric power disruptions in California caused cross-sectoral cascading effects [1]. Electric power disruptions affected the natural gas production, the operation of petroleum product pipelines transporting gasoline and jet fuel within California and to Nevada and Arizona, and the operation of massive pumps used to move water for crop irrigation (first-order dependencies). Gas production curtailed by power losses directly impacted gas supplies for generating units, further exacerbating power problems (feedback loop). Tight natural gas supplies

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 104–115, 2013.

also had the potential to shut down gas-fired industrial co-generation units producing steam for injection into California's heavy oil fields (second-order dependencies), thus potentially reducing heavy oil recovery (third-order dependencies). Similarly, the disruption of product pipelines caused inventories to build up at refineries and draw down at the product terminals (second-order dependencies), including several major California airports. Declining jet fuel stocks at airports caused several major airline operators to consider contingency plans in the event of fuel shortages (third-order dependencies).

Similarly, the blackouts in the US-Canada (August 2003), Southern Sweden and Eastern Denmark (September 2003), and Italy (September 2003) highlight the possibility of international cascading effects. The common element in these cases is that a single event, which may have been assessed initially to pose relatively limited and isolated effect, is indeed causing problems to other infrastructures. In all three blackouts, we observe a chain of failures causing cross-border effects and significant impact to people, even without estimating the impact of their cross-sector effect like the California example.

The impact of a disruption, or failure, may spread both geographically and across multiple sectors. Identifying interdependencies may appear to be a useful task; however, there are specific dependencies, which are not easy to identify, e.g. social dependencies. Social dependencies may refer, for example, to the changes in individual behavior during a crisis, which may consequently affect various infrastructures or networks. For example, a disruption in the transportation sector may cascade in wireless communication networks [3]. Although the identification of first-order interdependencies may be sufficient, in order to assess the risks of a particular CI, they may fail to capture cascading risks in a macroscopic level. For example, one or more relatively minor, security incidents on one CI may cause cascading and escalating impacts to an interdependent CI of a second or third level. Identifying multi-order CI interdependencies leads to a more accurate assessment on the criticality level of an CI or a sector. It also enables the identification of chains between interdependent CIs. This way, it becomes possible to identify the "most" critical among the infrastructures and adopt more cost-efficient security controls, so as to reduce cumulative risks and avoid catastrophic cascading failures.

In this paper we will analyze the cascading effects of security incidents in CIs, so as to assess the possible cumulative effects of a single security incident on multiple CIs. Such effects are the result of interdependencies, which are hard to identify and - most of the times - are out of the scope of mainstream risk assessment methodologies. Our ultimate goal is to reduce the cumulative risks of security incidents and to avoid catastrophic cascading failures, by reducing threat, vulnerability, and/or impact levels, in the most appropriate and cost-efficient steps of a chain of interdependent CIs.

The paper is organized as follows. Section 2 provides definitions of interdependencies and disruptions on CIs. Section 3 summarizes the method on which the proposed approach is based on. Then, it describes the new steps required, in order to assess second-order dependencies. This is followed by a comprehensive example.

Section 4 describes other existing approaches in CI dependency assessment. The paper concludes with Section 5, where future research steps are referred to.

## 2 Interdependencies and Disruptions

Following [1, 4], dependencies may be:

- *Physical* (the state of a CI depends upon the material output(s) of the other CI),
- *Cyber/Informational*(the state of a CI depends on information transmitted through the other CI),
- *Geographic* (the state of a CI depends on an environmental event on another CI),
- *Logical* (the state of a CI depends upon the state of another CI via a non-physical, cyber, or geographic connection) or
- *Social* (the state of a CI is affected by the spreading of disorder to another CI related to human activities).

The interdependence-related disruptions or outages can be classified as cascading, escalating, or common-cause [1]. A *cascading* failure is defined as a failure in which a disruption in an infrastructure A affects one or more components in another infrastructure, say B, which in turn leads to the partial or total unavailability of B. An *escalating* failure is defined as a failure in which an existing disruption in one infrastructure exacerbates an independent disruption of another infrastructure, usually in the form of increasing the severity or the time for recovery or restoration of the second failure. A *common-cause* failure occurs when two or more infrastructure networks are disrupted at the same time: components within each network fail because of some common cause. This occurs when two infrastructures are co-located (geographic interdependency) or because the root cause of the failure is widespread (e.g., a natural or a man-made disaster).

## 3 Assessing Hidden Interdependencies for Critical Infrastructures

Critical Infrastructure Protection (CIP) is usually based on risk assessment reviews [5]. With traditional risk assessment methodologies, a Critical Infrastructure Operator (CIO for short) will assess the information risks of all the assets within the organization, in order to identify the most critical assets. The criticality of the assets is related to the potential impact for the organization, which may result because of the unavailability, disclosure, or modification of an asset. In recent CIP research [1, 5–8], the criticality of an asset depends not only on the potential impact of a security incident on the organization itself, but also on the outgoing societal risk caused to other dependent organizations. For example, if a major energy provider is experiencing a disruption for a certain period (i.e. unavailability of the core service of this CIO), this will result in potential impact not only for the operator itself, but also for any other interconnected operators belonging to various sectors, and also for all the potential users of all the dependent operators. In order to identify and mitigate the security risks caused due to the interdependencies between CIOs, our approach will be based on a recently proposed, multi-layer risk assessment methodology for interdependent critical infrastructures [7, 8].

### 3.1 Pre-requisites: A Risk-Based Criticality Assessment Methodology

In [7, 8], a risk-based criticality assessment methodology is presented. The goal of the methodology is to identify which infrastructures are the most critical, and to assess the security risks related with these CIs. The rationale of the methodology relies on the fact that traditional risk assessment methodologies are organization-oriented (i.e. they assess the security risks of an infrastructure mainly by measuring the possible consequences for the operator organization, in case of a security event). For this reason, they cannot always capture the criticality of an infrastructure in a macroscopic level (i.e. what are the *societal impacts*, in case of a security event realized on an infrastructure). This is closely related with the interdependencies between CIs.

Based on the interdependencies between different infrastructures, in [7, 8] the *criticality level* of an infrastructure (or a complete sector) is assessed based on three risk factors: (a) the *societal risk* that may be caused to the society (or to a significant number of persons), due to a security incident realized to the particular infrastructure; (b) the outgoing risk on an infrastructure, which mainly consists of the potential risk caused to other infrastructures due to a security incident to this infrastructure; (c) the incoming risk on an infrastructure, which mainly consists of the potential risk suffered by the infrastructure in question, due to a security incident caused to another dependent infrastructure.

The methodology is organized in three phases or levels of analysis. In the Operator level, it is assumed that all the participating CIOs have already conducted an organization-wide risk assessment and have, thus, identified their first-order dependencies. Since these interdependencies are known, each CI is expected to assess its incoming risks, i.e. the potential risks caused to the CI due to a security event in another connected infrastructure. In the second phase (Sector level), the results of the previous phase (incoming risks[1] of CIs) are analyzed by experts of each infrastructure sector, in order to estimate the outgoing societal impact of an incident or threat on other infrastructures (dependent CIs) and the society. In the third phase, the sector coordinators will reexamine all the results of the previous layers, in order to identify and confirm the dependencies between CIs, and form a more macroscopic view for the criticality of each sector at a national level, e.g. ICT, transport or power sector.

[1] For a particular dependency $A \rightarrow B$, the incoming risk estimated by $CI_B$ is essentially equivalent to the outgoing risk estimated by $CI_A$. By considering both views, sector-level experts may fine-tune the risks identified at the operator-level analysis.

This methodology [7, 8] identifies and assesses interdependencies between infrastructures, despite which sector each infrastructure is located or depends on. However, it only considers first-order dependencies, i.e. direct physical, logical, procedural, geographical or social dependencies between two CIs. Thus, the identification of second- or third-order dependencies is not captured and as described in Section 1 through real examples, such complex, chain dependencies are often the cause of major consequences.

### 3.2 The Proposed Method

Following the approach suggested in [8], by defining the first-order outgoing risks of various infrastructures in an Operator level and analyzing their societal risk in a Sector level, it is possible for the risk assessor to construct the *Dependency Risk Table*, as shown in Table 1 (based on an example of 8 infrastructures and 4 sectors).

The Dependency Risk Table summarizes the dependencies of each infrastructure to others. It also indicates for each dependency the source impact $SImp$ (i.e. the effect on the source of the dependency), the incoming impact $IImp$ (i.e. the potential effect on the dependent infrastructure), as well as the incoming impact scale and the likelihood of the source impact being realized. The product of the last two values is used for assessing the dependency risk. Method [8] assesses the societal risk of a disruption due to an (inter)dependency, and does not take into account the impact on the infrastructure operator at this stage (Sector level).

For example, as shown in Table 1, $CI_A$ has two dependent CIs, mainly $CI_G$ and $CI_F$. The infrastructure $CI_F$ (the second raw of the table) has a Cyber (or Infromational) dependency from $CI_A$, since $CI_F$ has outsourced its payment services to $CI_A$. A possible service unavailability of $CI_A$ will produce an incoming dependency impact to $CI_F$ (unavailability of its payment services), denoted as $I_{A,F}$. This would cause loss of public confidence to $CI_F$, of a relatively low impact ($I_{A,F} = (L)ow$). The likelihood of an event causing unavailability to $CI_A$ (and consequently a cascading unavailability to $CI_F$) is considered low, i.e. $L_{A,F} = (L)ow$. Thus the outgoing risk of this dependency, denoted as $R_{A,F} = I_{A,F} \times L_{A,F}$ has a risk value equal to 4, based on a risk matrix as described in [8]. Although the example considers total loss of availability as source and incoming impact, modified risk matrices can also be formed in order to assess various levels of service loss.

Dependencies can be visualized through graphs, as shown in Figure 1. An infrastructure is denoted as a circle. An arrow from $X \rightarrow Y$ denotes a risk dependency, i.e. an outgoing risk from the infrastructure $CI_X$ to the infrastructure $CI_Y$. A bi-directional arrow $X \leftrightarrow Y$ denotes an outgoing risk from $CI_X$ to $CI_Y$ and another outgoing risk from $CI_Y$ to $CI_X$. The number in each arrow refers to the level of the incoming risk for the receiver due to the dependency, based on a risk scale $[0-9]$. For example, $CI_G$ has an incoming dependency risk of 6 from the infrastructure $CI_A$. This risk value refers to the likelihood of a disruption from $CI_A$ to cascade to $CI_G$, as well as the societal impact in the case of such an event.

**Table 1.** Dependency Risks

| **Dependent CIs** | **Dep. Type** | **Description** | **SImp** | **IImp** | **IImp Type** | **Scale** $I_{j,i}$ | **LH** $L_{j,i}$ | **Risk** $R_{j,i}$ |
|---|---|---|---|---|---|---|---|---|
| | $CI_A$ (Finance Sector) | | | | | | | |
| $CI_F$ | C | Provides payment services | UA | UA | Public Confidence | L | L | 4 |
| $CI_G$ | C | Provides payment Services | UA | UA | Public Confidence | H | L | 6 |
| | $CI_B$ (Energy Sector) | | | | | | | |
| $CI_A$ | P | Depends for power | UA | UA | Economic Impact | VL | L | 3 |
| $CI_C$ | P | Depends for power | UA | UA | Public Confidence | H | VL | 5 |
| $CI_D$ | P | Depends for power | UA | UA | Economic Impact | VH | VL | 6 |
| $CI_E$ | P | Depends for power | UA | UA | Economic Impact | H | VL | 5 |
| $CI_F$ | P | Depends for power | UA | UA | Public Confidence | L | L | 4 |
| $CI_G$ | P | Depends for power | UA | UA | Public Confidence | H | L | 6 |
| | $CI_C$ (ICT Sector) | | | | | | | |
| $CI_F$ | C | Network Services | UA | UA | Public Confidence | L | VL | 3 |
| $CI_G$ | C | Network Services | UA | UA | Public Confidence | H | VL | 5 |
| | $CI_D$ (ICT Sector) | | | | | | | |
| $CI_C$ | P | Depends for network connectivity | UA | UA | Public Confidence | H | VL | 5 |
| $CI_E$ | P | Depends for network connectivity | UA | UA | Economic Impact | H | VL | 5 |
| | $CI_E$ (ICT Sector) | | | | | | | |
| $CI_F$ | G | Hosts backup systems | UA | UA | Public Confidence | L | VL | 3 |
| $CI_G$ | G | Hosted in its facilities | UA | UA | Public Confidence | VH | VL | 6 |
| | $CI_F$ (Government Sector) | | | | | | | |
| $CI_G$ | C | Receives insurance information | UA | UA | Public Confidence | L | L | 4 |
| $CI_G$ | S | Industrial action | UA | UA | Economic Impact | L | M | 5 |
| | $CI_G$ (Government Sector) | | | | | | | |
| $CI_F$ | S | Industrial action | UA | UA | Economic Impact | M | M | 6 |

**Dependency.** P: Physical, C: Cyber, G: Geographic, Log: Logical, S: Social
**Source/Incoming Impact (SImp/IImp).** UA: Unavailability, DS: Disclosure, MD: Modification
**Scale/Likelihood.** VH: Very High, H: High, M: Medium, L: Low, VL: Very Low

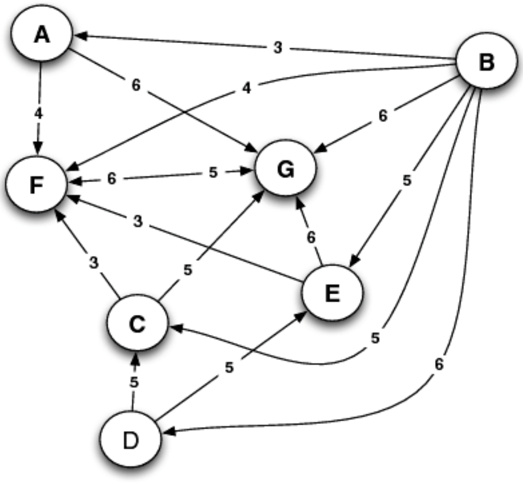


**Fig. 1.** Dependency Risk Graph of interdependent CIs

In order to estimate second-order dependency risks, the following steps are performed for each examined critical infrastructure $CI_i$:

1. **Identification of the $1^{st}$-order dependencies of $CI_i$.** Identify all the incoming dependency risks of $CI_i$. For simplicity, and without loss of generality, we assume that the incoming risk $CI_j \rightarrow CI_i$ has risk value $R_{j,i} = L_{j,i} \times I_{j,i}$, where $I_{j,i}$ is the incoming impact and $L_{j,i}$ is the likelihood of this incoming impact, as computed in Table 1. For example, as shown in Figure 1, the infrastructure $C$ has an incoming dependency from $B$ and another one from $D$. We will examine the $D \rightarrow C$, $1^{st}$-order dependency.
2. **Identification of the $n$-order dependencies of $CI_i$.** During this step, we identify the correlated $2^{nd}$ and more generally, $n$-order dependencies of $CI_i$. For each $1^{st}$-order incoming $CI_j \rightarrow CI_i$ dependency of the examined infrastructure $CI_i$, examine the source infrastructure $CI_j$ in order to identify its possible incoming dependencies $CI_k \rightarrow CI_j$. If the incoming impact of the dependency $CI_k \rightarrow CI_j$ is of the same type as the source impact of the $CI_j \rightarrow CI_i$ dependency, then mark this dependency and continue until all the possible threads of the $n$-order dependencies of $CI_i$ have been examined. In the example of Figure 1, for the $C \rightarrow D$ dependency identified in step 1, we examine the $2^{nd}$-order dependency $B \rightarrow D$ (the complete $n$-order dependency of this thread is $B \rightarrow D \rightarrow C$). By examining Table 1 the incoming impact of the $B \rightarrow D$ dependency is of type Unavailability ($IImp(B \rightarrow D) = UA$), which is of the same type as the source impact of the $D \rightarrow C$ dependency ($SImp(D \rightarrow C) = UA$). Thus this second order dependency is marked and we continue by examining possible $3^{rd}$-order dependencies. Since $B$ has no other incoming dependencies, all the possible $n$-order dependencies of this thread have been examined and marked according to the rule

of this step and we can continue with another thread of $C$'s dependencies. By examining Figure 1 we see that the infrastructure $C$ has another incoming dependency $B \rightarrow C$. Thus $C$ has a $1^{st}$-order and a $2^{nd}$-order dependency from $B$.

3. **Evaluation of the $n$-order dependency risks.** Check if the $CI_k \rightarrow CI_j$ dependency has been marked in the previous step. In this case, the $2^{nd}$-order dependency risk $Risk(CI_k \rightarrow CI_j \rightarrow CI_i) \equiv R_{k,j,i}$ for short, can be computed as:

$$R_{k,j,i} = R_{j,i} \times L_{k,j,i} = (I_{j,i} \times L_{j,i}) \times L_{k,j,i} = I_{j,i} \times (L_{j,i} \times L_{k,j,i}) \quad (1)$$

where $L_{k,j,i}$ is the conditional probability of the likelihood $L_{j,i}$ being realized, given the fact that the likelihood $L_{k,j}$ has been realized, i.e.

$$L_{k,j,i} = P(L_{j,i}/L_{k,j}) = \frac{(L_{j,i} \cap L_{k,j})}{(L_{k,j})} \quad (2)$$

If we consider a worst-case scenario, then $L_{k,j}$ and $L_{j,i}$ can be considered as likelihoods of independent events and thus the conditional probability of Equation 2 can become:

$$L_{k,j,i} = P(L_{j,i}/L_{k,j}) = \frac{(L_{j,i} \cdot L_{k,j})}{(L_{k,j})} = L_{j,i} \quad (3)$$

Thus from Equations 1,3 we have:

$$R_{k,j,i} = R_{j,i} \times L_{j,i} = I_{j,i} \times {L_{j,i}}^2 \quad (4)$$

Equation 4 can be trivially extended in order to compute the $n$-order dependency risk $Risk(CI_1 \rightarrow CI_2 \rightarrow ... \rightarrow CI_n) \equiv R_{CI_1,CI_2,...,CI_n}$ as:

$$R_{CI_1,CI_2,...,CI_n} = R_{CI_{n-1},CI_n} \times L_{CI_{n-1},CI_n} = I_{CI_{n-1},CI_n} \times (L_{CI_{n-1},CI_n})^n \quad (5)$$

4. **Examine next infrastructure.** Repeat from step one until all the examined infrastructures are exhausted.
5. **Rank cascading risks.** Rank all the examined cascading risks and choose the most critical paths (according to a risk threshold set by the security experts).
6. **Mitigate cascading risks.** Consider risk mitigation controls throughout the path under a cost-benefit analysis, in order to reduce the dependency risks below the threshold, both on a sector and an infrastructure level. The examination of n-order dependencies allows the identification of the most critical infrastructures and their respective sectors in terms of chain effects. The examination of the risk path provides additional options for risk mitigation, in a 'cost-efficient' way. For example, the alternative risk mitigation approaches include:
   - Controls to reduce the likelihood of the possible events that may cause the source impact in the source of the examined dependency chain.

- Controls that reduce the likelihood of the possible events that cause the source impact in any intermediate node within the chain.
- Controls that reduce the impact of dependencies by creating alternative paths.
- Controls that increase the resilience of critical nodes in a dependency chain, thus reducing the impact on individual nodes.

When planning investments for critical infrastructures or sectors, the information provided by the dependency graphs and n-order dependencies can be significant. This is due to the fact that adopting such a macroscopic view permits a more efficient distribution of budget within or across sectors. It also reduces the cost of applying excessive countermeasures on all infrastructures, while it increases their effectiveness, not only in respect of the particular infrastructure, but of the dependent ones as well.

### 3.3 Example

If we consider an example of a second-order dependency, we would have three infrastructures: $CI_A$: Power Generator, $CI_B$: Train, $CI_C$: Mobile Network. These infrastructures face the following interdependencies:

$CI_A \rightarrow CI_B$: Physical Dependency (power supply)
$CI_A \rightarrow CI_C$: Physical Dependency (power supply)
$CI_B \rightarrow CI_C$: Social Dependency

Following the method described above, we perform the following steps:

1. We examine possible threats that will result in the Source Impact "disruption of $CI_A$", which causes blackout in a region (Societal Risk of $CI_A$).
2. Disruption in power supply causes several trains to be immobilized for several hours in this region (Incoming Impact in $CI_B$). This is a cascading disruption from A to B.
3. Disruption to communication network due to the blackout (Incoming Impact in $CI_C$). This is a cascading disruption from A to C, but it is also a common cause disruption between B and C.
4. Disruption to communication network follows due to heavy load (Incoming Impact in $CI_C$). This is a cascading disruption from B to C.

In order to calculate the cascading risk of the initial event from $CI_A$ to $CI_C$, we will have to assess the conditional probability (likelihood) $L_{C,A}$ and take into account all the potential societal impacts due to the following paths:

(a) $CI_A \rightarrow CI_B \rightarrow CI_C$: $R_{C,B,A} = f(R_{B,A}, R_{C,B})$ (second-order dependency risk) and

(b) $CI_A \rightarrow CI_C$: $R_{C,A}$ (first-order dependency risk)

The next step will be to evaluate these risks and examine which is the most cost-efficient way to mitigate them. Both paths of dependency need to be examined. Countermeasures options would be (a). the use of alternative power supply for $CI_C$ (reduce the probability $L_{C,A}$), (b). countermeasures for load management during crisis (reduce the probability $L_{C,B}$ or increase the resilience of infrastructure $CI_C$).

## 4 Related Work

Interdependency models and approaches found in the literature vary according to the level of analysis selected. Some adopt a microscopic and some a macroscopic view of dependencies. One approach [10] focuses on CI components (microscopic view), and demonstrates several types of multi-dependency structures for both linear and particularly cyclical dependencies among multiple infrastructure types. It also considers un-buffered and buffered types of resources. Another approach [11] focuses on the component level, as well, and models/simulates two types of vulnerability: (a). structural and (b). functional. It calculates the interdependent effect and the effect of interdependence strength. It includes examples on power grid and gas pipeline models. Other models examine dependencies between different CIs [12] or within the same or different sectors of a country [13]. A method to map interdependencies, with a workflow enabling the characterization of coupled networks and the emerging effects related to their level of interdependency, is presented by [14]. This work aims at mapping the interdependency between electrical and related communication nodes.

Several methods that are proposed for evaluating risk in interdependent CIs apply Leontief's Inoperability Input-Output model (IIM), which calculates economic loss due to unavailability on different CI sectors based on their interdependencies [6, 13, 15–17].

Theoharidou et al. assess risk in three layers: (a). infrastructure level, (b). sector level, and (c). national/intra-sector level [5, 7, 8]. The authors identify first-order dependencies and provide a method for evaluating societal risk between CIs and sectors. A similar approach is adopted on [18]. It follows six steps: (1). Identify the initiating event, (2). Identify interdependencies and Perform qualitative analysis, (3). Perform semi-quantitative assessment of the scenario, (4). Perform detailed quantitative analysis of interdependencies (optional), (5). Evaluate risk and measures to reduce interdependencies, and (6). Perform Cost/benefit analysis (optional).

## 5 Conclusions

In this paper we examine the possible cumulative effects of a single security incident on multiple CIs. Such paths of dependent CIs add complexity and are usually out of the scope of typical risk assessment methodologies. Our method provides a way to identify threats that may appear insignificant when examining only first-order dependencies, but may have potentially significant impact if one adopts a more macroscopic view and assesses multi-order dependencies.

Based on previous work, we utilize existing first-order dependency graphs, in order to assess the effect of a disruption to consequent infrastructures. This approach utilizes existing risk assessments that refer to the societal risk of first-order interdependencies (performed at a sector level). Also, the assessment of impact is not expressed only on economic terms, like most IIM approaches. Finally, it is scalable to n-order dependency assessments.

The current approach does not analyze the graphs fully, so it does not evaluate possible cycles or reverse interdependencies. Also, it does not consider parallel paths in an automated way, as well as their potential effect to minimize risk. Future steps will include the adoption of graph analysis algorithms, in order to identify the most critical paths of dependencies, and to provide ways to reduce risks by adopting alternative paths in a graph. In order to validate our method, we also plan to apply the model in a real scenario which will analyze interdependencies between transport, ICT and energy infrastructures.

**Acknowledgments.** This work was supported in part by the S-Port (09SYN-72-650) project, which is funded by the Hellenic General Secretariat for Research & Technology, under the *Synergasia* Programme.

## References

1. Rinaldi, S., Peerenboom, J., Kelly, T.: Identifying, Understanding and Analyzing Critical Infrastructure Interdependencies. IEEE Control Systems Magazine 21(6), 11–25 (2001)
2. Andersson, G., Donalek, P., Farmer, R., Hatziargyriou, N., Kamwa, I., Kundur, P., Martins, N., Paserba, J., Pourbeik, P., Sanchez-Gasca, J., Schulz, R., Stankovic, A., Taylor, C., Vittal, V.: Causes of the 2003 major grid blackouts in North America and Europe and recommended means to improve system dynamic performance. IEEE Trans. on Power Systems 20(4), 1922–1928 (2005)
3. Barrett, C., Beckman, R., Channakeshava, K., Huang, F., Kumar, V., Marathe, A., Marathe, M., Pei, G.: Cascading failures in multiple infrastructures: From transportation to communication network. In: 5th Int. Conf. on Critical Infrastructure (CRIS), pp. 1–8 (2010)
4. De Porcellinis, S., Oliva, G., Panzieri, S., Setola, R.: A Holistic-Reductionistic Approach for Modeling Interdependencies. In: Palmer, C., Shenoi, S. (eds.) Critical Infrastructure Protection III. IFIP AICT, vol. 311, pp. 215–227. Springer, Heidelberg (2009)
5. Theoharidou, M., Kotzanikolaou, P., Gritzalis, D.: Risk-based criticality analysis. In: Palmer, C., Shenoi, S. (eds.) Critical Infrastructure Protection III. IFIP AICT, vol. 311, pp. 35–49. Springer, Heidelberg (2009)
6. Setola, R., De Porcellinis, S., Sforna, M.: Critical infrastructure dependency assessment using the input-output inoperability model. Int. J. Critical Infrastructure Protection 2(4), 170–178 (2009)
7. Theoharidou, M., Kotzanikolaou, P., Gritzalis, D.: A multi-layer criticality assessment methodology based on interdependencies. Computers & Security 29(6), 643–658 (2010)
8. Theoharidou, M., Kotzanikolaou, P., Gritzalis, D.: Risk Assessment Methodology for Interdependent Critical Infrastructures. Int. J. Risk Assessment & Management (2011) (to appear)
9. Rinaldi, S.: Modeling and simulating critical infrastructures and their interdependencies. In: 37th Hawaii Int. Conf. on System Sciences, USA, vol. 2. IEEE (2004)
10. Svedsen, N., Wolthunsen, S.: Connectivity models of interdependency in mixed-type critical infrastructure networks. Information Security Technical Report, vol. 1, pp. 44–55 (2007)

11. Min, O., Liu, H., Zi-Jun, M., Ming-Hui, Y., Fei, Q.: A methodological approach to analyze vulnerability of interdependent infrastructures. Simulation Modeling Practice and Theory 17, 817–828 (2009)
12. Nieuwenhuijs, A., Luiijf, E., Klaver, M.: Modeling dependencies in critical infrastructures. In: Goetz, E., Shenoi, S. (eds.) Critical Infrastructure Protection II. IFIP, vol. 290, pp. 205–214. Springer, Boston (2008)
13. Aung, Z.Z., Watanabe, K.: A framework for modeling Interdependencies in Japan's Critical Infrastructures. In: Palmer, C., Shenoi, S. (eds.) Critical Infrastructure Protection III. IFIP AICT, vol. 311, pp. 243–257. Springer, Heidelberg (2009)
14. Rosato, V., Issacharoff, L., Tiriticco, F., Meloni, S., De Porcellinis, S., Setola, S.: Modeling interdependent infrastructures using interacting dynamical models. Int. J. Critical Infrastructures 4(1/2), 63–79 (2008)
15. Santos, J., Haimes, Y.: Modeling the demand reduction input-output inoperability due to terrorism of interconnected infrastructures. Risk Analysis 24(6), 1437–1451 (2004)
16. Haimes, Y., Santos, J., Crowther, K., Henry, M., Lian, C., Yan, Z.: Risk Analysis in Interdependent Infrastructures. Critical Infrastructure Protection 253, 297–310 (2007)
17. Crowther, K.: Decentralized risk management for strategic preparedness of critical infrastructure through decomposition of the inoperability input-output model. Int. J. Critical Infrastructure Protection 1, 53–67 (2008)
18. Utne, I.B., Hokstad, P., Vatn, J.: A method for risk modeling of interdependencies in critical infrastructures. Reliability Engineering & System Safety 96(6), 671–678 (2011); ESREL 2009 Special Issue

# How to Perform Verification and Validation of Critical Infrastructure Modeling Tools

Alfonso Farina[1], Antonio Graziano[1], Stefano Panzieri[2],
Federica Pascucci[2], and Roberto Setola[3]

[1] Selex ES - A Finmeccanica Company, via Tiburtina Km 12.400, Roma, Italy
{alfonso.farina,antonio.graziano}@selex-es.com
[2] University Roma Tre, Via della Vasca Navale 79, 00146 Roma, Italy
pascucci@dia.uniroma3.it, panzieri@uniroma3.it
[3] University Campus Bio-Medico, Via A. del Portillo 21, 00128 Roma Italy
r.setola@unicampus.it

**Abstract.** Simulation tools appear the only solution suitable for with the complexity of the actual critical infrastructure scenarios. Indeed, due to interdependency phenomena and the fast innovation of the technologies, our capability to predict the global behavior of such system of systems on the basis of past history and experiences is dramatically reduced, especially in the presence of anomalous or crisis situations. This drives many groups to develop simulation platforms also able to support decisions during crisis. However, a crucial and not adequately investigated aspect in this framework is the qualification of the predictable capabilities of these tools. This paper would start a discussion on how to validate different approaches, and how to asses their predictable capabilities, providing some insights on this strategic and very challenging task.

**Keywords:** Simulation, Critical-Infrastructures, Credibility, Accuracy.

## 1 Introduction

Modern societies largely depend on the existence of a set of technological infrastructures such as electricity, telecommunication, transportation, etc., that for their relevance are generally indicated as Critical Infrastructures [1]. Due to their increased complexity (largely induced by the presence of dependencies and interdependencies) they appear prone to catastrophic failures as shown by the 2003 black-out or by the Katrina hurricane (for a more extensive survey see [2]).

As a consequence, it imposed to develop innovative methodologies and tools to improve the resilience and the robustness of these infrastructures. Unfortunately, the complexity of critical infrastructure scenarios overcome the actual analytical methodologies and impose to adopt simulators as the exclusive tools able to support the experts in the understanding of the global behavior of such a system of systems especially in the presence of anomalous situations.

Actually, there are several initiatives all over the world to develop modeling and simulation platforms for critical infrastructures. Among the others we can

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 116–127, 2013.

cite the CISIA project [3], the SimCIP tool developed within the EU project IRIISS [4], the federated approach proposed by Idaho National Lab. [5], and the US ambitious project developed at National Infrastructure Simulation and Analysis Center [6], as well as the attempt to create an European Infrastructure Simulation and Analysis Center within the EU project DIESIS (an overview of the most promising initiatives can be found in [7–9] and in the references therein).

One of the main challenges in this framework is to qualify the capability of the different tools to correctly reproduce the reality or, in other terms, to provide an answer to the question: "Are the results provided by the simulator believable, and, if yes, with which degree of credibility?"

In this paper, after a short review of the classical *Verification & Validation* approaches, we provide some considerations about their applicability in the critical infrastructures framework. The results of our analysis emphasize that the validation phase, i.e. the test of the adequateness of the modeling to reproduce the real world, is a very challenging task due to the impossibility to perform any significant *experiment* and to the unfeasibility to operate on historical data. To overcome such limitations, the paper delineates a set of procedures that appear, even if not able to rigorously validate the tools, useful to increase our confidence on them.

## 2 Verification and Validation

Simulation tools are largely adopted for a broad range of purposes, including what-if analysis, design, problem solving and training. Any simulation analysis is composed by a set of complex and interrelated tasks that starts with the conceptual model formulation (i.e. the representation of the phenomena under investigation with a formal language as mathematical equations) encompass the implementation phase (i.e. the translation of the conceptual model into an executable software) to ended with the result presentations. A successful simulation study is defined to be the one which produces a credible and acceptable solution for the prescribed task. How to support a successful simulation project has been largely investigated in the past, especially in the military, spatial and recently also in surgery frameworks, due to the dramatic consequences that could be generated by erroneous previsions. To this aim, several principles and techniques have been proposed to assess the accuracy of Modeling and Simulation (M&S), known under the label of Verification and Validation (V&V), well reviewed in [10] and [11] and the references therein. To better understand V&V and evaluate their impact over critical infrastructure framework, it is useful to start detailing the meaning of these two terms.

In literature there are several definition of *Verification*; however, the most accepted is that initially expressed by the ANSI/IEEE standard and subsequently refined by the DoDI, and specifically:

- The process of determining whether or not the products of a given phase of the software development cycle fullfil the requirements established during the previous phase [12].
- The process of determining whether a model implementation and its associated data accurately represent the developer's conceptual description and specifications [13].

*Verification* activities are focused to *verify* that the conceptual model has been correctly codified into an executable software. Therefore is substantially aimed to guarantee that the model is transformed from its conceptual/mathematical form into software able to operate as intended and with sufficient accuracy (see Figure 1). Model verification deals with *building the model right.* Verification questions whether the simulator works in accordance with its specifications/requirements. As noted in [14] verification has two aspects: design (all specifications and nothing else are included in the model or simulation design) and implementation (all specifications and nothing else are included in the model or simulation as built). Verification activities are (or better should be) generally performed concurrently with software development.

Conversely, the ANSI/IEEE, and the subsequently refinement of the DoDI, definition for *Validation* is:

- The process of evaluating the software at the end of the software development process to ensure compliance with software requirements [12].
- The process of determining to which extend a model and its associated data provide an accurate representation of the real world from the perspective of the intended uses of the model [13].

The goal of *Validation*, therefore, is to *validate* the conceptual model: i.e. to verify that the conceptual model, within its domain of applicability, behaves with satisfactory accuracy consistent with the real world. Hence model validation deals with building the *right* model, i.e. those able to correctly capture and reproduce all the relevant aspects and features of the phenomena under study. Validation addresses two questions: "Is the model adequate to reproduce the reality?" (this question is generally referred as *Conceptual Model Validation*) and "Are the results of the simulation coherent with the reality?" (that represents the *Data Validation*). Generally, data obtained from the real world (or a credible source) is used to compare the behavior and results of the simulation. Validation should be performed at the best before the coding phase; however for most of non-trivial projects it is unfeasible. Thus, it should be performed as early as possible, eventually on partial results, in order to prevent and correct in early stages inconsistent/erroneous assumptions.

Roughly speaking, as schematically illustrated in Figure 1, the verification phase is intended to perform those activities to guarantee that model is correctly implemented, while the goal of the validation task is to compare the model with the phenomena. On the one hand, verification is devoted to check that the implemented software correctly solve the mathematical conceptual model used to describe the phenomena under study. On the other hand validation is devoted

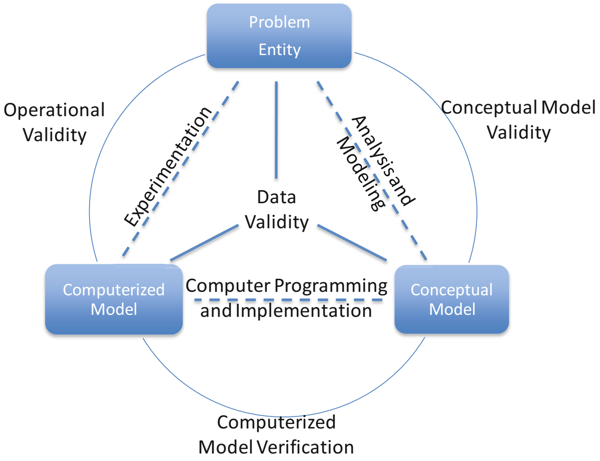


**Fig. 1.** M&S life cycle as proposed by Sargent [26]

to assess the effectiveness of the conceptual model to reproduce the real world with an accuracy adequate with the intent of the study.

Historically, the V&V has been dominated by post-construction testing. However, software engineering wisdom suggests that substantial benefits will result performing V&V during all the M&S life cycle [27]. Obviously, V&V approach must be tailored to match the nature of the problem, which includes also the types of decisions that are driving the employment of the simulation [13].

There are several V&V techniques presented in the literature [10, 11], most of these derived from software engineering for this reason the term testing is used frequently when referring to the implementation of these techniques because they involves the testing of the model or simulation to assess its credibility. On the basis of the use of mathematical and logic (as well as the complexity) a possible taxonomy of V&V techniques classifies them into four, partially overlapping, categories [22]: *informal techniques*, *static techniques*, *dynamic techniques*, and *formal methods*

*Informal techniques* are the most commonly used. They are called informal because the tools and approaches used rely heavily on human reasoning and subjectivity, without stringent mathematical formalism. The *informal* label does not imply any lack of structure or formal guidelines for the use of the techniques. In fact, these techniques are applied using well-structured approaches under formal guidelines and they can be very effective if employed properly. Among others, of specific interest are those generally referred as *Subject Matter Expert (SME)* where, as for the *Face Validation* technique [17], potential users of the model and experts of the system under study, based on their intuition, subjectively judge if the models and its results are reasonable. A similar approach is based on the famous *Turing test*, i.e. on the capability of an operator to discriminate between answer provided by a human or a machine. The experts are presented with two sets of outputs obtained, one from the model and one from the system, under the same input conditions. Without identifying which one is which, the experts are asked to differentiate between the two. If they cannot differentiate, the confidence in model validity is increased [18].

*Static techniques* assess the accuracy of the static model design and source code. They do not require machine execution of the model, but mental execution can be used.

On the other side, *dynamic techniques* evaluate the model based on its execution behaviour. This requires the insertion of additional code (probes or stubs) into the executable code to collect information about model behaviour during execution. For our purpose, an interesting approach is the so called *Predictive Validation* technique [16], used to test the predictive ability of a model. It requires past input and output data from the system being modeled. The model is driven by past system input data and its outputs are compared with the corresponding past system output data.

Finally, *formal methods* are based on formal mathematical proofs of correctness. The most commonly known techniques are: induction, inference, logical deduction [29]. Although their potentiality, the applicability to concrete problems seem hard.

Furthermore, specific techniques have been developed to validate distributed and/or federated simulators in order to manage the specific problems imposed by such architectural approaches [23].

## 3 V&V in Critical Infrastructure Framework

As mentioned in the Introduction, a crucial question posed to on-going M&S projects in the field of critical infrastructures is related to the qualification of their credibility: “how much we can base emergency plans and impact analysis on their outputs?” Unfortunately, there is no answer to this question due to both the intrinsic complexity of the problem and the peculiarities of the critical infrastructures domain.

However in this framework we can only partially re-use classical V&V techniques illustrated in the previous section. Even if the complexity of the critical infrastructures scenario is comparable with those of the battlefield and/or space explorations, it has some peculiarities that make more challenging its validation. In the critical infrastructure scenario, indeed, we have, for some aspects, characteristics of both these domains together with other further difficulties.

A battlefield scenario is composed by a plethora of entities, having each one with some degree of autonomy and interacting in a complex way. In this framework, the elementary behavior of any element can be inferred with high accuracy by means of experimental tests, but the ultimate goal of the M&S is deducing the overall behavior generated by the entities interaction. On the other side, space exploration scenarios are composed by a limited set of entities that exhibit a large number of complex interrelations inside a only partially known environment, where the possibility to perform experiments is very limited.

Like battlefield, critical infrastructure scenarios are typified by the presence of a large number of (quite) autonomous entities that largely interacts each other. However, in this cases, the behavior of any element is largely influenced by the interactions with the other elements. In other terms, the so called *emergent behaviors* are of the most relevant. Moreover, they cannot be reasonably predicted on the sole base of the knowledge of the atomistic functioning of the single entity, as, for example, it is impossible to predict the behavior of an ant colony on the base of the knowledge of the behavior of any single ant. In this framework a further element of complexity is given by the problem to include into the model also the behavior of the human operators and users. Indeed in any significant scenario there are hundreds or thousands of operators that have the potentiality to modify the behaviour of each infrastructure. Such operators follow strategies and goals specific for their own organization which sometime is aligned with those of the other infrastructure operators, but in other cases are indifferent to the others, or even have clashing interests. Moreover, operators during high stressing situations might assume erroneous/anomalous decisions [15], while users, during crisis, tend to perform irrational behaviors (creating the so called sociological interdependencies phenomena [3]). As well know the inclusion into the modeling of human behavior, and especially how to validate is, by itself, an hard challenge [14].

Like in space exploration, the interactions among the entities (in this case referred as dependencies and/or interdependencies) are only partially know because a large number of them have been never planned but they are *created*, or better induced, by modifications of the environments. Moreover, because critical infrastructures are largely owned by private operators for which data about the actual architecture is considered sensible information, there is a strong reluctance to provide detailed data so as information about incidents (or near-missing incidents).

Just to add a further complication, due to the fast innovations that characterize the actual infrastructure architectures and technologies, historical data can be used only marginally to infer predictive behavior for such scenarios. Furthermore, it is evidently unfeasible any type of large scale (i.e., systemic dimension) experiments. Thus, we are in the unpleasant situation where we need to validate M&Ss tools but, at the same time, M&Ss are the only instruments that we have to characterize the behavior of the system. More formally, to test the quality of the different conceptual models, i.e., the conceptual model validity of Figure 1, we are forced to largely use the results obtained by means of simulation tools, with very limited capability to perform any experimentation on the field.

Before concluding, it is worth to be stressed again that, in spite of other domains, actually there are quite no metrics to be used for validation [11]. Questions like: in which way comparing the affinity of two scenarios (or just only how much two infrastructure are dependent/interdependent), or how comparing the outputs of a simulator with the actual data have been still only marginally investigated [9]. Many authors suggest to report all data to a monetary base, but this represents a very crude simplification not able to capture all the relevant aspects. Consequently we have no chance to quantitatively validate M&S tools (i.e., no accuracy information can be realistically provided), but we have to limit our goals to qualitative statements, i.e. no more than the Level 1 as validation model maturity [28].

To start to start a discussion about this topic, in the following we will critically review some techniques that appear useful to perform V&V techniques in the framework of critical infrastructures.

### 3.1 Tune the Simulator on a Reference Scenario

The need to improve the protection of critical infrastructures emerged from several catastrophic episodes happened in the last years, e.g. the attack to the Twin Towers of 9/11, the 2003 black-out in US and in Italy, the Katrina hurricane, the earthquake in Japan in 2011, just to cite someones. These episodes have been deeply analyzed, hence a chance to validate an M&S tool is to prove its capability to correctly reproduce the time sequence of the different events as they were registered during the crisis. This can be considered as a special case of the *Predictive Validation* technique [16] described in Section 2.

The main difference with this technique is that we do not have a series of causes/effects histories to compare with the input/output sequences of the simulator, but for any specific scenario we have just a single episode. In fact, any episode happened in a specific geographical location and involved some specific infrastructures. Even if our simulator were able to reproduce with high accuracy what happened for the specific event, we have no information about its prediction capability in different scenarios (and also to study the effects of different threats on the same geographical site). In other terms, we have no elements to estimate the *inference* capability of our M&S tool, i.e. the credibility of the solutions for an application domain *different* by those used for the validation process [11].

Consequently, such a technique appears feasible in the case one needs a M&S tool to perform a deeply ex-post analysis of a specific crisis, e.g. to improve the capability of lesson learning and for training, but it helps only very marginally to improve our level of confidence in the simulator outputs. A benefit of this technique is its capability to provide quantitative measurements of the accuracy of the simulation allowing estimating its bounds of confidence.

### 3.2 Subject Matter Expert (SME)

A well-known technique to validate a simulation tool is showing to a set of experts the outputs of the simulator and asking them to give an opinion about their plausibility [14]. Unfortunately, in our case, the limitation for this approach is mainly related to the difficulties of any expert to foreseen the *right* behavior of a complex scenario where several infrastructures interact. Since the innovative socio-technological framework deeply changed the operational capability of the different infrastructures especially in the presence of anomalous events, past data and past experiences can only minimally be re-used to predict the future, as already outlined.

The main risk occurring with this kind of validation technique is that the experts should use the results to *strengthen* their own opinions rather to validate the M&S platform. In other terms, there is the concrete risk that experts use the simulator to have a confirmation about their beliefs rather than use it to explore *unexpected* behaviors.

### 3.3 Multi Models Check

To study only some aspects of Critical Infrastructures, several approaches have been proposed in the literature further to the force brute simulators [24, 25]. These studies are based on simplified conceptual model where only some peculiarities of the system are considered ranging from the underlying characteristics of the networks (e.g., complex networks [19] or using holistic approaches, e.g. IIM [20]). Even if such models are generally less detailed and less accurate than the M&S tools their outputs can be used in the validation process, since as noted in [21], more abstract models are generally based on data which are less affected by uncertainties and more unbiased because they limit the subjective hypothesis necessarily introduced when facing more complex models.

Given a scenario, comparing the outputs of a simulator with those obtained with others, conceptual model allows us to increase our confidence in the adopted conceptual model. Obviously this type of comparisons can be performed only within the domain of applications of the other models, this means generally only in situations close to the nominal behavior of the system.

### 3.4 Rule Games

In the military and civil protection field it is normal to perform *table-top exercises* to improve the capability to cooperate among all the actors involved in the management of a given event where a hypothetical scenario is designed and the different actors operate as they would have done in a real situation.

A similar approach could be used to validate Critical Infrastructures M&S tools. The idea is to design a hypothetical scenario, find out its evolution with by the simulator, and analyse its behaviour with the help of several experts. Specifically, the simulator generates a first set of outputs, i.e., at a given time shortly after the start of the crisis, and these outputs are shared with the experts participating to the exercise. They initially validate the coherence of the outputs, as done within the *SME* technique, but this activity is not performed autonomously by the different experts, it is achieved via an *active* discussions performed by all the experts around the table. This allows to dramatically reduce the risk of self-strengthening by sharing different experiences and visions, while analyzing short time effects at each step.

Once the experts validated the first set of outputs, they design their reaction strategies which represent inputs for the simulator. This last will evolve for another step and its outputs will be shown to the experts and so on. In this way, the technique should be considered as the transposition of software debugging into the validation process: the simulator evolves step by step reporting the actual values of the different variables to the experts and allowing them to check the coherence of the different data during the running.

The main drawback of this technique becomes evident in the case where, at a given step, the outputs of the simulator are considered erroneous by the experts. In this case it is necessary to analyze the conceptual model and the implemented solution to discover the wrong element, fix it and re-run the last step of the simulator (assuming that the correction does not affect also the previous results, in this case one has to re-start all the exercise). It is evident that this activity cannot be generally performed during the exercise, since one needs to suspend it and re-call the experts for a new session of the exercise. This mechanism considerably increases the time-consumption and the costs of the procedure. Moreover, if the experts discover several errors, i.e. if they are called several times to discuss about the same scenario, they rapidly lost interest in the exercise. For these reasons, rule game technique should be used for a last validation, i.e. when already there is some confidence on the quality of the model.

However, this technique for validation show several intrinsic positive elements: at the same time the M&S tool is validated, a training session for operators is carried out and an occasion for information sharing among the experts is created. This last element is by itself a valuable result.

### 3.5 Cross Validation

As illustrated in the paper, the only instrument today available to analyze and predict the behavior of a scenario composed by heterogeneous and interdependent infrastructures is the simulation. In this framework, the best way to validate a new simulator is to compare its outputs with a those produced by a *qualified* simulator. By itself this is not a new idea, quite all the simulator developed for complex phenomena such as, for example, pollution diffusion, are qualified using as reference some well tested simulators. However, as mentioned, currently there are no *tested* simulators for critical infrastructures. Moreover, most of the developed simulators are tailored on specific scenarios which detailed cannot be shared.

A possible solution is therefore, as stressed in the Conclusions, to set-up some benchmarking scenarios that should be implemented by several simulators in order to compare their outputs and, consequently, discover the coherent and the incoherent elements among the different conceptual models allowing improving our confidence in them.

### 3.6 Some Remarks

It is evident that no one of the previous illustrated procedures are adequate to effectively validate a M&S tool, however a strategy that combine different techniques appear able to provide more valuable insights. To this end, a multi steps validation process should be considered, allowing the validation of the S&M system all along the life cycle.

According with this vision, the first validation approach should be the comparison of the outputs of the simulators obtained for the first simulation steps with the results provided by other (more abstract and simplified) approaches as foreseen for the *multi models check*. The goal of this step should be a first *data validation*, which is abel to check whether all the relevant relations/elements have been considered or not.

Afterward, with the help of one or more *SME* sessions, the evolution trends of each single infrastructure should be checked.

Tuning the system on one or more *reference scenarios* allows fixing the parameters and providing an estimation of the result accuracy.

Finally, the validation of the M&S is achieved with one or more sessions of *rule games*, each one designed on one of the previous reference scenarios after having introduced some modifications in order to evaluate the *inference capability* of the system. Unfortunately, this process is very time and cost consuming, especially because it requires a strong cooperation with end-users and domain experts. In the presence, as solicited in the Conclusions of this paper, of benchmarking reference scenarios some of these activities should be performed more efficiently and cost-effective.

## 4 Conclusions

Can we base emergency plans and disaster recovery strategies on the outputs of one of the many Critical Infrastructures simulators on-developing all over the World? Or, better, how much we can be confident about their results? Are they *validated*? These are crucial questions and answers for them are mandatory in order to make really useful and effective the large efforts committed in this field.

These questions are common to other applications contexts where simulators are used to predict the behavior of complex scenarios as the battlefields, the space explorations, the minimal invasive surgeries, etc. Unfortunately, the classical V&V techniques developed with references to these fields can be applied only partially to the problem at hand, due to its intrinsic complexity, the presence of several autonomous actors, and the unfeasibility of performing any large scale experiment, as well as the limited value of past experiences and historical data.

Specifically, while the techniques for the verification of the *right* implementation of the code can be assumed to be more or less the same of other fields of the same complexity, the validation phase, i.e. the proof to have implemented the *right* model, is more challenging.

In order to provide some hints on how to manage validation activities in critical infrastructure scenarios, we proposed in this paper some approaches.

An interesting aspect that emerges from this analysis is the urgency to improve our capability to share data and experiences inside the community. Indeed, as illustrated for the *cross validation* technique, comparing the outputs of different conceptual models is a concrete and effective method to increase their confidence. Unfortunately all the on-going research activities refer to specific scenarios, whose details are often classified due to the sensible nature of the information treated reducing significantly the chance to perform cross validation activities. To overcome such a drawback, as done in other fields of research, it should be very useful the design some *reference* scenario, to be used as a benchmark, i.e. one or more hypothetical (no-sensitive) detailed scenarios to be used as a challenge over which test the predictable capability of the different tools so as the effectiveness of counter-measurements.

To conclude, this would be an invitation to all the people currently involved in M&S of critical infrastructures to disseminate to the community information (and details) about their scenarios, so as to design together a *reference* scenario (sufficiently complex and detailed) to be enough representative of the real-world. Its availability will provide large benefit to all the actors involved in Critical Infrastructure Protection field.

## References

1. Brunner, E., Suter, M.: International CIIP Handbook 2008/2009, CRN Handbooks series, Center for Security Studies (CSS), ETH Zurich (2008)
2. Bologna, S., Setola, R.: The Need to Improve Local Self-Awareness in CIP/CIIP. In: Proc. of First IEEE International Workshop on Critical Infrastructure Protection (IWCIP 2005), Darmstadt, Germany, pp. 84–89 (2005)
3. De Porcellinis, S., Panzieri, S., Setola, R., Ulivi, G.: Simulation of Heterogeneous and Interdependent Critical Infrastructures. Int. J. Critical Infrastructures (IJCIS) 4(1/2), 110–128 (2008)
4. EU Project IRRIIS: Integrated Risk Reduction of Information-based Infrastructure Systems, `http://www.irriis.org/`
5. Critical Infrastructure Resiliency Simulation (CIPR/Sim), `http://www.inl.gov/research/critical-infrastructure-resiliency-simulation/`
6. National Infrastructure Simulation and Analysis Center (NISAC), `http://www.sandia.gov/nisac/`
7. Pederson, P., Dudenhoeffer, D., Hartley, S., Permann, M.: Critical Infrastructure Interdependency Modelling: A Survey of U.S. and International Research, Idaho National Lab (2006)
8. EU project DIESIS (Design of an Interoperable European federated Simulation network for critical InfraStructures), Deliverable, D2.3 Report on available infrastructure simulators, `http://www.diesis-project.eu/`
9. Setola, R.: How to Measure the Degree of Interdependencies among Critical Infrastructures. Int. J. of System of Systems Engineering (IJSSE) 2(1), 38–59 (2010)
10. Topcu, O.: Review of Verification and Validation Methods in Simulation, Defence R&D Canada (2003)
11. Oberkampf, W.L., Trucano, T.G., Hirsch, C.: Verification, Validation and Predictive Capability in Computational Engineering and Physics. In: Proc. of Workshop Foundations for Verification and Validation in the 21st Century, October 22-23. Johns Hopkins University, Laurel (2002)
12. ANSI/IEEE, Standard Glossary for Software Engineering Terminology (1983)
13. DoD Instruction (DoDI), 5000.61 DoD Modeling and Simulation (M&S) Verification, Validation, Accreditation (VV&A) (1996)
14. Pace, D.K.: Modeling and Simulation Verification and Validation Challenges. Johns Hopkins APL Technical Digest 25(2), 163–172 (2004)
15. EU Project, Vital Infrastructure Threats and Assurance (VITA), `http://vita.iabg.eu/`

16. Emshoff, J.R., Sisson, R.L.: Design and Use of Computer Simulation Models. MacMillan, New York (1970)
17. Hermann, C.F.: Validation Problems in Games and Simulations with Special Reference to Models of International Politics. Behavioral Science 12(3), 216–231 (1967)
18. Turing, A.M.: Computing Machinery and Intelligence. In: Feigenbaum, E.A., Feldman, J. (eds.) Computers and Thought, pp. 11–15. McGraw-Hill, New York (1963)
19. Caldarelli, G., Vespignani, A.: Large Scale Structure and Dynamics of Complex Networks. World Scientific Publisher (2007)
20. Haimes, Y.Y.: Risk Modeling, Assessment, and Management. Wiley (2009)
21. Bigelow, J.H., Davis, P.K.: Implications for Model Validation of Multiresolution, Multiperspective Modeling. RAND (2003)
22. Balci, O.: Verification, validation and testing. In: Banks, J. (ed.) The Handbook of Simulation. John Wiley & Sons, New York (1997)
23. IEEE, 1516.4 Recommended Practice for Verification, Validation, and Accreditation of a Federationan Overlay to the High Level Architecture Federation Development and Execution Process (2007)
24. Panzieri, S., Oliva, G., Setola, R.: Modelling and Simulation of Critical Infrastructures. In: Flammini, F. (ed.) Critical Infrastructure Security: Assessment, Prevention, Detection, Response. WIT Press (2011)
25. Satumitara, G., Duenas-Osorio, L.: Synthesis of Modelling and Simulation Methods on Critical Infrastructure Interdependencies Research. In: Gopalakrishnan, K., Peeta, S. (eds.) Sustainable and Resilient Critical Infrastructure Systems. Springer, Berlin (2010)
26. Sargent, R.G.: Verification and Validation of Simulation Models. In: Proc. of Winter Simulation Conf., pp. 121–130 (1998)
27. Hone, G.N.: Application Domain Modeling to support the Verification and Validation of Synthetic Environments. In: Simulation Interoperability Workshop (1998)
28. Harmon, S.Y., Youngblood, S.M.: A Proposed Model for Simulation Validation Process Maturity. Journal of Defense Modeling and Simulation (JDMS) 2(4), 179–190 (2005)
29. Birta, L.G., Zmizrak, F.N.: A Knowledge-Based Approach for the Validation of Simulation Models: The Foundation. ACM Transactions on Modeling and Computer Simulation 6(1), 76–98 (1996)

# Petri Net Modelling of Physical Vulnerability

Francesco Flammini[1], Stefano Marrone[2], Nicola Mazzocca[3],
and Valeria Vittorini[3]

[1] AnsaldoSTS, Innovation and Competitiveness Unit (Italy)
francesco.flammini@ansaldo-sts.com

[2] Seconda Università di Napoli, Dip. di Matematica e Fisica (Italy)
stefano.marrone@unina2.it

[3] Università di Napoli "Federico II", Dip. di Ingegneria Elettrica e Tecnologie dell'Informazione (Italy)
{nicola.mazzocca,valeria.vittorini}@unina.it

**Abstract.** Several multi-disciplinary aspects need to be addressed in security risk evaluation, including the estimation of risk attributes. One of the most widespread definitions of security risk relates it to the attributes of: probability of occurrence (or rather "frequency") of threats, system vulnerability with respect to the threat (or rather "probability of success of the threat"), and expected consequences (or rather "damage"). In this paper we provide a straightforward generic model based on Stochastic Petri Nets which can be adopted for the quantitative evaluation of physical vulnerability. The model allows to evaluate besides effectiveness parameters (e.g. probability of sensing, assessment, neutralization, etc.) also efficiency related ones (e.g. time to sense, assess, neutralize, etc.). Some examples will be provided in order to show how the model can be used in real-world protection systems applications.

**Keywords:** Risk Analysis, Model-Based Vulnerability Assessment, Stochastic Petri Nets, Physical Security.

## 1 Introduction

Nowadays security risk analysis of critical systems and infrastructures is a primary issue. One of the most widespread and simple mathematical model for the quantitative evaluation of the risk associated with a certain threat accounts for: threat occurrence rate, system vulnerability with respect to the threat and expected damage caused by the threat. In particular, the vulnerability parameter represents the (conditional) probability that the attack is successful, that is to say the threat finally damages the target asset. If the asset is not hardened nor protected, than the vulnerability is 1, otherwise it is less than 1. Therefore, in this paper vulnerability is not defined as "a flaw of the system which can be exploited", a widespread qualitative definition especially in computer security.

A foremost problem with the aforementioned risk model is that the vulnerability is very difficult to evaluate. Several ad hoc models have been proposed for risk evaluation, however most of them fail to answer the simple question "Given

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 128–139, 2013.

a certain threat and a certain protection system, which is the probability that the threat succeeds?" (or in other words, "which is system vulnerability with respect to the threat?"). In this paper we provide a stochastic model based on a certain class of Petri Nets, which allows to give an approximate answer to that question in a way which is as simple as possible. The objective is that the model can be easily customized by only slight modifications to its structure and/or parameters. While (complex) models have been proposed in the scientific literature for risk evaluation in specific applications, to the best of our knowledge no simple generic model exists allowing quantitative vulnerability evaluation based on threat characterization and on the effectiveness and efficiency parameters of the protection system. Furthermore, often risk models are not described in detail for confidentiality reasons [16]. We believe that the generic customizable model described in this paper can help in supporting quantitative vulnerability evaluation in many real-world applications, as demonstrated by the example case-studies we provide.

The rest of this paper is organized as follows. Section 2 provides a brief risk taxonomy and pointers to the related literature. Section 3 describes and discusses the vulnerability model. Section 4 provides some evaluation examples using parameters of real-world applications. Finally, Section 5 draws conclusions and provides some hints about future developments.

## 2 Basic Definitions and Related Works

The Department of Homeland Security Risk Steering Committee has provided a publicly accessible document which represents a comprehensive reference of risk taxonomy [9]. In the remainder of this section, we will concentrate on definitions which are most related to the topic of this paper.

With reference to a specific threat, the quantitative risk $R$ can be formally defined as follows [14]:

$$R = P \cdot V \cdot D$$

- $P$ (sometimes found as "T", from the initial of *Threat*) is the expected frequency of occurrence of the threat, which can be measured in [events/year];
- $V$ is the vulnerability of the asset with respect to the threat, that is to say the likelihood that an attack is successful, given that it is attempted;
- $D$ (sometimes found as "C", from the initial of *Consequences*) is an estimate of the measure of the expected damage occurring after a successful attack, which can be quantified and expressed in any currencies, e.g. Euros [€].

The vulnerability $V$ is a non-dimensional parameter, since it represents the conditional probability:

$$P(success|threat)$$

Therefore, a quantitative way to express the risk associated to a specific threat is to measure it in lost Euros per year: [€/year]. Though subject to criticism in some applications [13], the risk model defined above has been widely accepted by risk analysts, including the ones belonging to US national laboratories (see e.g. [2]). Nevertheless, the model is so simple to be nearly useless without a supporting methodology for the evaluation of the parameters involved in the analysis. Further details about practical applications and security surveys for vulnerability assessment can be found in references [6,4,1] and in [7,8] in the context of information security. In addition, reference [10] provides the description of a tool to automatically compute the expected annual benefit of a security system starting from the quantitative attributes of threats and protection mechanisms (and their interrelationships) using an extension of the basic risk model described above.

While many different definitions of Vulnerability Assessment may be found in the scientific literature (see e.g. [11]), in this paper we will only refer to the quantitative model-based evaluation of the V parameter of the risk formula. Generally speaking, evaluating the vulnerability corresponds to assessing the effectiveness of protection systems, which poses many challenges. For instance, in [18] a framework is described which addresses (but does not solve) several issues related to the evaluation of deployed security systems, considering both game theory and reliability theory. A simpler model which can be used to assess the vulnerability of a facility with respect to a threat has been adopted by Hennessey et al. [12]. In that model:

$$V = 1 - P_E \qquad P_E = P_D \cdot P_I \cdot P_N \qquad P_D = P_S \cdot P_A$$

Where:

- $P_E$ (probability of effectiveness) is the probability that the physical protection system is effective against the threat;
- $P_D$ (probability of detection) is the probability that the intruder has been detected;
- $P_S$ (probability of sensing) is the probability that a sensor detects the intrusion;
- $P_A$ (probability of an assessment) is the probability that the control room operator correctly assesses the situation and reacts accordingly;
- $P_I$ (probability of interruption) is the probability that the response force gets to the scene in time to neutralise the threat;
- $P_N$ (probability of neutralization) is the probability that the response force successfully neutralises the threat.

In simple words, in order to defeat an attacker, a series of activities must be successfully completed, including sensing, assessment and neutralization. How to quantify such probabilities is out of the scope of this work (some hints on stochastic modeling approaches are provided in [15]); however, estimations of sensing, assessment and neutralization probabilities may be sometimes derived by historical data, simulations and/or or expert judgment. Once such estimations are available, a simple multiplication would be enough to evaluate $P_E$, with the exception of the $P_I$ parameter, which being time-dependent is more complicated to evaluate. In fact, even if it were 100% effective in terms of detection and neutralization success rate, the security system would be completely useless in case the response force would be unable to stop the perpetrators before they have the possibility to strike. Many real-world systems suffer from such a limitation, which has been raised as a major criticism against security technologies, which often only serve as a means to improve the sense of security instead of actually reducing the vulnerability. In this paper we will use the vulnerability definition reported above to focus on the evaluation of the $P_I$ parameter.

Even though, as mentioned above, usually the term vulnerability has a different meaning when used in the context of computer security, that does not mean that the method described in this paper cannot be employed in order to evaluate computer security risks. In fact, according to the computer dependability taxonomy, physical attacks belong to the class of human-made deliberate malicious threats; as such, they are relevant in the evaluation of overall system resiliency against physical attacks and/or hacker penetration/access to networked terminals [19]. In this regard, some surveys of stochastic modeling techniques which can be employed also for security evaluation are provided in [15,17].

We have chosen to use the Stochastic Petri Net (SPN) formalism in the TimeNET tool [20] since it has a virtually unlimited expressive power, so that the basic models (which are easy to understand even to non skilled modelers) may be customized and/or extended in order to account for behaviors of any type and complexity.

## 3 The Petri Net Vulnerability Model

Before starting the description of the vulnerability model, we would like to remark that the following assumptions hold: (1) the model does not account for possible deterrent effects (influencing the threat occurrence rate) nor for consequence mitigation effects (influencing the expected damage), which should be considered only in higher level risk models; (2) due to its stochastic nature and simple structure, the model only provides a rough approximation of the result (which is what is needed in practice); as a consequence, its parameters do not require a very high precision (which would be nearly impossible to achieve in practice); (3) since failure parameters (both for threat and protection mechanisms) can be accounted for by a simple multiplication (as explained in the previous section), it is not necessary to complicate the model in order to consider them; (4) the model describes a single threat scenario: in case multiple scenarios need to be modeled, more models should be evaluated and their results summed or combined somehow; in case concurrent scenarios need to be modeled, also possible limitations in the number of active responders should be modeled; (5) the basic model does not account for (a) multiple levels of threat progression and/or detection and (b) any intelligent/adaptive behaviors of attackers/defenders however, it may be extended to account for them if required; (6) as a constraint of the SPN formalism, the completion time of the basic activities is distributed as a negative exponential stochastic variable, whose mean should be chosen as the expected delay in "nominal" or "standard" operational conditions, or rather as an appropriate mean among the most common scenarios.

All the assumptions listed above are essential to simplify the model in order to make it easy to use in practical applications and still meaningful. Regarding the last assumption (no. 6), please note that it introduces the necessary non-determinism which allows to account for variations in the activity completion times, which is especially important since the system is a "human-in-the-loop" type (but also sensing times of technological devices are not always deterministic).

In such assumptions, the resulting vulnerability model is the one depicted in Figure 1 with its elements described in Table 1, where:

- $L_T$ is the threat latency, that is the mean time for the threat to reach the target asset starting from the sensing point;
- $L_S$ is the sensing latency, that is the mean time for the sensors to generate[1] and transmit to the control center a warning event or an alarm;
- $L_A$ is the (remote) assessment latency, that is the mean time for the control room operator(s) to assess the situation and react accordingly;
- $L_R$ is the response latency, that is the mean time for the response force to get to the scene in order to neutralize the threat.

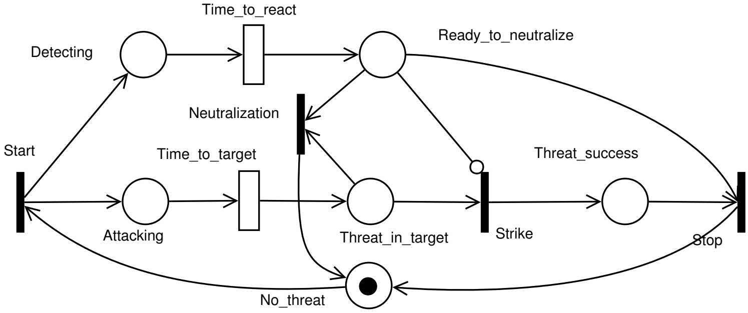


**Fig. 1.** The basic SPN model for vulnerability evaluation

To the best of our knowledge, all those latencies have not been explicitly taken into account in any generic models (like the one addressed in [12] and already discussed in the previous section); however, it is easy to understand that - together with the fail probabilities - they are essential parameters for the stochastic evaluation of physical vulnerability. The model works as follows. First of all, since vulnerability evaluation is conditional to the presence of a threat, the initial state in which the place *No_threat* has 1 token (enabling the *Start* transition) is evanescent (that is, the mean number of tokens in that place is

[1] Please note that not all sensors provide instantaneous outputs. For instance, smart-sensors like intelligent cameras or trace detectors include classifiers which require several seconds for the elaboration of input data.

**Table 1.** Description of the SPN Nodes

| Node Name | Type | Description | Parameter Value |
|---|---|---|---|
| No_threat | Place | Initial status | Initial Marking = 1 |
| Start | Immediate transition | Threat start trigger | Priority = 1, Weight = 1 |
| Attacking | Place | Threat started the attack scenario | Initial Marking = 0 |
| Detecting | Place | Sensor(s) started to detect | Initial Marking = 0 |
| Time_to_target | Stochastic Transition | It models the threat delay to get to the target asset | Delay = $L_T$ |
| Time_to_react | Stochastic Transition | It models the overall reaction delay including sensing, assessment and response latencies | Delay = $L_S + L_A + L_R$ |
| Threat_in_target | Place | Threat has reached the target asset | Initial Marking = 0 |
| Ready_to_neutralise | Place | Countermeasure(s) ready to neutralize the threat | Initial Marking = 0 |
| Strike | Immediate transition | Threat strike trigger | Priority = 1, Weight = 1 |
| Neutralization | Immediate transition | Threat neutralization | Priority = 1, Weight = 1 |
| Threat_success | Place | The attack has been successful | Initial Marking = 0 |
| Stop | Immediate transition | Attack scenario ends | Priority = 1, Weight = 1 |

0). The scenario always starts from the left, with 2 tokens generated by the *Start* transition, one in the *Attacking* place and one in the *Detecting* place: that models the situation in which the threat starts moving from the sensing point to the target asset. Moving to the right of the model, the two parallel stochastic transitions *Time_to_react* and *Time_to_target* are meant to model the concurrent actions of the attacker(s) and the defender(s):

- If the attacker arrives first to the target (1 token in *Threat_in_target* and no token in *Ready_to_neutralise*), then it has the possibility to strike (*Strike* transition is enabled, firing one token in *Threat_success*). Here the threat success probability is assumed to be 1, which is a sort of worst case which could be adjusted to get a more precise result in case threat failures are not negligible. Finally, the *Stop* transition resets the network to its initial state.
- In case the defender arrives first (1 token in *Ready_to_neutralise* and no token in *Threat_in_target*), then the *Strike* transition is disabled due to the inhibitor arc connecting it to the *Ready_to_neutralise* place, while the *Neutralization* transition is enabled, firing a token in *No_threat* and thus completing the

scenario. Here the detection probability is assumed to be 1 since, as mentioned above, detection failures can be simply evaluated by multiplying the result by $P_S$ and $P_A$ (see Section 2).

Figure 2 shows how to separately model the detection, assessment and response latencies. A similar approach could be used for threat latency modeling as well (including e.g. time to deploy, time to activate, etc.). That influences the kind of probability distribution of the overall latencies, since a sum of exponentially distributed stochastic variables features another type of distribution. Though in some cases it could make sense to go into these details, for the sake of simplicity we will not specify into our reference model any sub-activities (we will come back to discussing such an aspect later in this section).

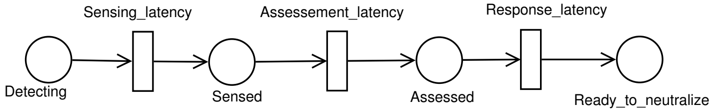


**Fig. 2.** PN modeling for distinct sensing, assessment and response latencies

Instead, explicit failure modeling complicates the network, increases the size of the reachability graph, and hence it can significantly slow-down model evaluation due to the state-space growth. As an example, we report in Figure 3 how to model the detection failure: two additional immediate transitions *Detect_success* and *Detect_failure* are enabled by tokens in place *Detecting*, with their weights representing the complementary probabilities, e.g. if $P_D = 0.9$ then:

$$weight(Detect\_success) = 0.9 \quad and \quad weight(Detect\_failure) = 0.1$$

It is easy to prove that the required reward expression to evaluate vulnerability (in the assumption $P_S = P_A = P_N = 1$) is as follows:

$$V = 1{-}P_E = 1{-}P\{\#Ready\_to\_neutralise = 1 \quad IF \quad \#Threat\_in\_target = 0\}$$

In fact, in order to neutralize the threat, the response must be ready before the threat has the possibility to strike. In such a condition, the inhibitor arc from the place *Ready_to_neutralise* prevents the transition *Strike* to fire, giving priority to the other enabled transition named *Neutralization*. Given the above assumptions, it is straightforward to understand that in case we need to know $P_I$, that can be simply obtained after model evaluation as (1–$V$).

A basic validation of the reward expression may be performed by applying a boundary analysis to its parameters. As expected:

$$\begin{array}{l}
Delay(Time\_to\_react) = 0 \quad AND \quad Delay(Time\_to\_target) > 0 \Rightarrow V = 0 \\
Delay(Time\_to\_react) > 0 \quad AND \quad Delay(Time\_to\_target) = 0 \Rightarrow V = 1 \\
Delay(Time\_to\_react) \gg Delay(Time\_to\_target) \Rightarrow V \backsimeq 1 \\
Delay(Time\_to\_react) \ll Delay(Time\_to\_target) \Rightarrow V \backsimeq 0 \\
Delay(Time\_to\_react) = Delay(Time\_to\_target) \Rightarrow V = 0.5
\end{array}$$

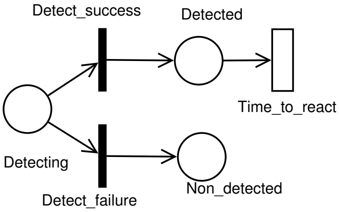


**Fig. 3.** Example SPN failure modeling

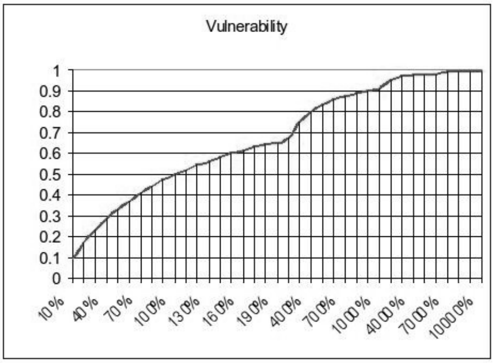


**Fig. 4.** Vulnerability as a function of the percentage ratio: delay(Time_to_react) / delay(Time_to_target)

In Figure 4 we report the results of a generic model evaluation. A single model execution (i.e. stationary analysis) in the TimeNET tool (Windows XP version) running on a typical office laptop (Intel Core 2 CPU 1.83 GHz with 2GB RAM) lasts only a few seconds. Since absolute timings are not relevant, we evaluated Vulnerability with respect to the following percentage ratio: delay(*Time_to_react*)/delay(*Time_to_target*). Neglecting approximations which alter curve regularity, the shape is parabolic tending asymptotically to 1 as the ratio tends to infinite, as expectable.

Regarding the probability distributions for the activities, though the exponential model is the most convenient in practice, it is important to highlight that (citation from [3], p. 165, 7.2): "The possibility of including timed transitions with general firing time distributions in GSPN (Generalized Stochastic Petri Nets) models is provided by the phase expansion that allows the behaviour of a random variable with general distribution to be approximated by means of a combination of negative exponential random variables with appropriate parameters. These distributions are called Phase-Type (PH) distributions. This means that an activity that lasts for a generally distributed random amount of time can be modelled by breaking it down into a set of stages (phases), all lasting for exponentially distributed periods of time."

As an example, consider the 2-phase attack vulnerability model depicted in Figure 5. In that model, the *Time_to_target* is split into two contributions: *Phase1* and *Phase2*. With such a model, the following result holds:

$$Delay(Phase1) = Delay(Phase2) \quad AND$$
$$Delay(Phase1) + Delay(Phase2) = Delay(Time_t o_r eact) \Rightarrow V = 0.56$$

In other words, the PH distribution assumption on the attacker side has increased (i.e. worsened) the vulnerability of about the 12%. In case of non homogenous bipartitions, the vulnerability increases a little bit less. More in detail:

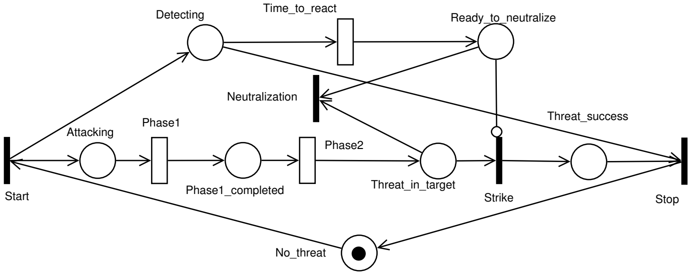


**Fig. 5.** SPN model of a 2-phase attack

$$Delay(Phase1) \neq Delay(Phase2) \quad AND$$
$$Delay(Phase1) + Delay(Phase2) = Delay(Time_to_react) \Rightarrow 0.5 < V < 0.56$$

Generally speaking, it could be shown that the result worsens as more attack phases are added (e.g. 4-phases with homogenous delays imply V = 0.66). However, excluding extreme cases, the impact on the results of considering more phases is generally limited and, nevertheless, it can be evaluated on a case by case basis by means of proper sensitivity analyses on the overall risk model.

# 4 Vulnerability Evaluation Examples

## 4.1 Case 1: Anti-theft and Intrusion Detection Systems

Valuable assets which are not continuously guarded are usually protected by means of active intrusion detection sensors which are part of surveillance systems featuring operators in remote control rooms or directly connected to the police stations. Let us assume we need to evaluate the vulnerability to thefts of a mission-critical server in a technical room which has an access door from the outside. Let us further assume that there is no active local siren, which is realistic in many industrial surveillance cases, to avoid disturbing people in case of false/nuisance alarms. Access control devices and magnetic contacts are used to detect unauthorized door openings. The magnetic contact has a very high reliability, lets suppose 98% (which usually gives the wrong perception that the overall protection system is very effective). The alarm is propagated to the control room in a few seconds, lets suppose 5s, and usually (in 95% of the cases) assessed in another bounce of seconds, say 15s, plus the time to call the responders and communicate the situation, say 30s. The responders are available and effective in 95% of the cases, needing about 3 minutes (180s) to get to the location. Once the door has been opened, the thief needs about 2 minutes (120 s) to disconnect the server and take it out.

Therefore: $P_S = 98\%, P_A = 95\%, P_N = 95\%, L_T = 120s, L_S = 5s, L_A = 15 + 30 = 45s, L_R = 180s$. $P_I$ can be evaluated using the model in Figure 1 with: Delay(Time_to_react) = 230s, Delay(Time_to_target) = 120s. With the above parameters we obtain: $P_I = 34\%$. Therefore: $V = 1 - P_S \cdot P_A \cdot P_I \cdot P_N = 0.7$.

Therefore, despite of the reliability of the detection device, in more than 2 out of 3 cases, the theft will be successful in its intent. That suggests to install additional stand-off detection devices (e.g. motion detection cameras in the external area), quick response countermeasures (e.g. fog generators to blind thieves without damaging electronic devices), or even to guard the asset locally, depending on the other risk parameters (frequency of theft attempts, criticality of the asset, etc.).

### 4.2 Case 2: CBRNe Detection

The protection against Chemical Biological Radiological Nuclear and explosive (CBRNe) threats is often required in infrastructure security applications. In that case more than in others, the presence of detectors is not enough to decrease system vulnerability. In fact, the response strategy is essential, as we will formally demonstrate in the following. Consider a metro railway application in which detectors are installed before the turnstile barriers and no people/baggage screening is performed by dedicated security staff. Let us assume that (see also previous example): $P_S = P_A = P_N = 95\%$.

The average times to get to the target asset (e.g. a crowded area, like platform or train) and drop the substance/device is around 30s (a little bit more if the perpetrator needs to completely leave the station before the explosive device activates), and about the same time holds for the response latency (assuming local guards in the station). Using ad-hoc radio communications, the time needed to operators to assess the alarm and contact guards can be as low as 30s, but sensing times of CBRNe are usually higher (about 15s). Therefore: $L_T = 30s, L_S = 15s, L_A = 30s, L_R = 30s$.

Hence: Delay(Time_to_react) = 75s, Delay(Time_to_target) = 30s. Model evaluation provides the following result: $P_I = 29\%$. Therefore: $V = 0.75$, that is to say on average that in 3 out of 4 attacks the perpetrators will be successful. In those conditions, despite of the "perceived security", the CBRNe detection system is almost useless. However, a simple countermeasure can make it much more effective: the automatic blocking of the entrance turnstile doors in case of detected alarms. If such a countermeasure is adopted, with the only drawback of slowing down the passenger flow in case of false-alarms, the response latency becomes a few seconds, say 3s, hence some latencies change as follows: $L_A = 2s$ (computer elaboration), $L_R = 3s$ (actuator command).

Also, since there is no human-in-the-loop, reliability parameters change as follows: $P_A = P_N = 99\%$. Thus: Delay(Time_to_react) $= L_S + L_A + L_R = 15 + 2 + 3 = 20s$, and the result becomes: $P_I = 60\%$. Therefore: $V = 0.44$. That is to say, in more than one half of the cases the CBRNe protection mechanism is able to neutralize the perpetrators.

## 5 Conclusions and Future Developments

In this paper we have presented a simple, generic and customizable model for the quantitative evaluation of physical vulnerability starting from parameters characterizing threat and countermeasure dynamics. The model is based on a certain class of SPNs allowing a very high expressive power; despite of that, it has an easy to understand basic structure which can be enriched in order to model more complex scenarios whenever required. In practical applications, however, just a rough approximation of the vulnerability is needed, since input parameters are not known with a very high precision. Therefore, the basic model can be more than enough to evaluate the effect of response latencies versus the time dynamics of the threat. That is required to populate risk models like the one presented in [10], which has mainly motivated the work presented in this paper. The effectiveness of any risk modeling approach is questionable under several points of view, including type (qualitative vs quantitative) and complexity (simple vs extensive). The approach is based on the three pillars which are well summarized by the following quotes:

1. "You can't control what you can not measure", Tom DeMarco
2. "Make things as simple as possible, but no simpler", Albert Einstein
3. "All models are wrong but some are useful", George E. P. Box

The first one suggests that quantitative models need to be adopted in order to govern the security risk. The second and third suggest to build models which are easy to manage and quick to evaluate as far as they provide us with usable results. Besides that, the work presented in this paper can be a starting point to build libraries of models for the modular/compositional development (e.g. by superposition of different nets) of more complex risk models, in which more threats and more protections are concurrently considered, together with the interrelationships of vulnerability with threat frequency and expected consequences. A further work is related to sthe definition of a model-driven automatic generatation of formal models from high level descriptions as successfully done in the reliability field [5].

**Acknowledgements.** The work of Stefano Marrone is funded by Campania region within the *AGRIGENET* project (PSR Campania 07/13).

## References

1. Journal of physical security, `http://jps.anl.gov/`
2. A risk assessment methodology for physical security. White paper. Technical report, SANDIA National Laboratories (2008)
3. Ajmone Marsan, M., Balbo, G., Conte, G., Donatelli, S., Franceschinis, G.: Modelling with generalized stochastic petri nets. SIGMETRICS Perform. Eval. Rev. 26, 2 (1998)
4. Baker, G.H.: A vulnerability assessment methodology for critical infrastructure sites. In: DHS Symposium: Rand D Partnerships in Homeland Security (2005)
5. Bernardi, S., Flammini, F., Marrone, S., Merseguer, J., Papa, C., Vittorini, V.: Model-driven availability evaluation of railway control systems. In: Flammini, F., Bologna, S., Vittorini, V. (eds.) SAFECOMP 2011. LNCS, vol. 6894, pp. 15–28. Springer, Heidelberg (2011)

6. Broder, J.F.: Risk Analysis and the Security Survey. Butterworth-Heinemann (2006)
7. Casola, V., Mazzeo, A., Mazzocca, N., Vittorini, V.: A policy-based methodology for security evaluation: A security metric for public key infrastructures. Journal of Computer Security 15(2), 197–229 (2007)
8. Casola, V., Preziosi, R., Rak, M., Troiano, L.: A reference model for security level evaluation: Policy and fuzzy techniques. Journal of Universal Computer Science 11(1), 150–174 (2005)
9. Risk Steering Committee. DHS risk lexicon, `http://www.dhs.gov/xlibrary/assets/dhs_risk_lexicon.pdf`
10. Flammini, F., Gaglione, A., Mazzocca, N., Pragliola, C.: Quantitative security risk assessment and management for railway transportation infrastructures. In: Setola, R., Geretshuber, S. (eds.) CRITIS 2008. LNCS, vol. 5508, pp. 180–189. Springer, Heidelberg (2009)
11. Garcia, M.L.: Vulnerability Assessment of Physical Protection Systems. Butterworth-Heinemann (2005)
12. Hennessey, B., Wesson, R.B., Norman, B.: Security simulation for vulnerability assessment. IEEE Aerospace and Electronic Systems Magazine 22(9), 11–16 (2007)
13. Cox Jr., L.A.: Some limitations of risk = threat x vulnerability x consequence for risk analysis of terrorist attacks. Risk Analysis 28(6) (2008)
14. Lewis, T.G., Darken, R.P., Mackin, T., Dudenhoeffer, D.: Model-Based Risk Analysis for Critical Infrastructures. Critical Infrastructure Security - WIT Press (2011)
15. Nicol, D.M., Sanders, W.H., Trivedi, K.S.: Model-based evaluation: From dependability to security. IEEE Trans. Dependable Secur. Comput. 1, 48–65 (2004)
16. Rinaldi, S.M.: Modeling and simulating critical infrastructures and their interdependencies. In: Proceedings of the 37th HICSS 2004 - Track 2, vol. 2. IEEE Computer Society, Washington, DC (2004)
17. Sallhammar, K.: Stochastic Models for Combined Security and Dependability Evaluation. PhD thesis, Norwegian University of Science and Technology (2007)
18. Taylor, M.E., Kiekintveld, C., Western, C., Tambe, M.: A framework for evaluating deployed security systems: Is there a chink in your armor? Informatica 34 (2010), Special Issue on Quantitative Risk Analysis Techniques for Security Applications
19. Weingart, S.H.: Physical security devices for computer subsystems: A survey of attacks and defenses. In: Paar, C., Koç, Ç.K. (eds.) CHES 2000. LNCS, vol. 1965, pp. 302–317. Springer, Heidelberg (2000)
20. Zimmermann, A., Freiheit, J., German, R., Hommel, G.: Petri net modelling and performability evaluation with timeNET 3.0. In: Haverkort, B.R., Bohnenkamp, H.C., Smith, C.U. (eds.) TOOLS 2000. LNCS, vol. 1786, pp. 188–202. Springer, Heidelberg (2000)

# Investigating the Effect of Network Parameters on Coordinated Cyber Attacks against a Simulated Power Plant

Béla Genge and Christos Siaterlis

Joint Research Centre, European Commission
Institute for the Protection and Security of the Citizen
Via E. Fermi, 2749, Ispra (VA), 21027, Italy
{bela.genge,christos.siaterlis}@jrc.ec.europa.eu

**Abstract.** The fact that modern Networked Industrial Control Systems (NICS) depend on Information and Communication Technologies (ICT), is well known. Although many studies have focused on the security of these systems, today we still lack the proper understanding of the effects that cyber attacks have on NICS. In this paper we use our previously developed framework to study the effects of network parameters, i.e. delay, packet losses and background traffic, on coordinated cyber attacks against NICS. Coordinated attacks rely on several infected hosts to disrupt the normal functionality of the system. Within the context of NICS we consider multiple infected control hardware, a highly similar setting to the recently reported Stuxnet worm, the first malware specifically designed to attack NICS. Furthermore, we assume that the coordinator is located outside the system, in the Internet, from where it launches attacks by sending packets to each infected control hardware. The main goal of the attacker is to bring the physical process into a critical state, i.e. dangerous, or more generally unwanted state of the system. For the physical process we used the Boiling Water Power Plant (BWPP) model developed by Bell and Åström.

**Keywords:** Coordinated attack, Networked Industrial Control Systems, network parameters, Boiling Water Power Plant.

## 1 Introduction

Modern Critical Infrastructures (CIs), e.g. power plants, water plants and smart grids, rely on Information and Communication Technologies (ICT) for their operation since ICT can lead to cost reduction as well as greater efficiency, flexibility and interoperability between components. In the past CIs were isolated environments and used proprietary hardware and protocols, limiting thus the threats that could affect them. Nowadays CIs or more accurately Networked Industrial Control Systems (NICS) are exposed to significant cyber-threats; a fact that has been highlighted by many studies on the security of Supervisory Control And Data Acquisition (SCADA) systems [1], [2]. For example, the recently reported

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 140–151, 2013.

Stuxnet worm [3] is the first malware that is specifically designed to attack NICS. Its ability to reprogram the logic of control hardware in order to alter physical processes demonstrated how powerful such threats can be; it has served as a wakeup call for the international security community.

As already highlighted by previous research [4], coordinated attacks have a much greater impact on the target system than non-coordinated ones. In a coordinated setting the attacker relies on several infected hosts to disrupt the normal functionality of the system. The recently reported attack on Twitter [5], where a hacker used thousands of infected hosts to launch a DoS attack, has demonstrated just how powerful these attacks can be. Consequently, in this paper we use our previously developed framework [6] to study the effects of network parameters, i.e. delays, packet losses, background traffic, on coordinated cyber attacks against NICS. The coordinator, located outside the power plant, in the Internet, uses multiple infected control hardware to bring the system into a *critical state*, i.e. dangerous, or more generally unwanted state of the system [7]. The control hardware is infected with malicious code and is able to receive commands from the coordinator, which is a reasonable assumption if we consider that the Stuxnet malware showed a similar behavior. The attack scenario was implemented with our previously developed framework [6] that uses simulation for the physical components and an emulation testbed based on Emulab [8] in order to recreate the cyber part of NICS, e.g., SCADA servers, corporate network, etc.

The paper is structured as follows. Our study is presented in the context of other related approaches in Section 2, followed by a short overview of our previous work in Section 3. The experimental scenario and setup are presented in Section 4, followed by the analysis of coordinated attacks involving a Boiling Water Power Plant in Section 5. Conclusions are presented in Section 6.

## 2 Related Work

An approach where real sensors and actuators, combined with simulated PLCs and communication protocols were used to study cyber-physical systems has been proposed by Queiroz, *et al.* [10]. Their study showed that while PLCs are under a DoS attack, operators might take delayed or wrong decisions that could disrupt the operation of the plant. A similar experiment has also been documented by Davis, *et al.* [11] that used the PowerWorld server to study the effects of communication delays between the physical process and human operators. In the same direction, the work of Chabukswar, *et al.* [12] proved that a DDoS attack against communication nodes between controllers and sensors causes the PLCs to take wrong decisions based on old sensor values. Finally, we mention the work of Cárdenas, *et al.* [14] that didn't only document the effect of DoS attacks on sensors, but also proposed a new detection mechanism together with possible countermeasures.

The previously mentioned approaches demonstrated the effectiveness of DoS attacks, but without reaching a sophistication level that would have allowed the attacker to reprogram the low level control logic of the PLCs. This fact sets an important barrier in terms of knowledge, skills and efforts required by the attacker, as was the case of Stuxnet, where developers had also knowledge of the PLC code, OS and hardware details. In this category we find the work of Nai Fovino, *et al.* [13] that proposed an experimental platform for studying the effects of cyber attacks against NICS. In their paper the authors described several attack scenarios, including DoS attacks and worm infections that send Modbus packets to control hardware. Although the authors provided a wide range of countermeasures, they did not identify communication parameters that affect the outcome of the attacks.

## 3 Experimentation Framework Overview

The experimentation framework developed in our previous work [6] follows a hybrid approach, where the Emulab-based testbed recreates the control and process network of NICS, including Programmable Logical Controllers (PLCs) and SCADA servers, and a software simulation reproduces the physical processes. The architecture, as shown in Fig. 1, clearly distinguishes 3 layers: the cyber layer, the physical layer and a link layer in between. The cyber layer includes regular ICT components used in SCADA systems, while the physical layer provides the simulation of physical devices. The link layer provides the glue between the two layers through the use of a shared memory region. The physical layer is recreated through a soft real-time simulator that runs within the *SC* (Simulation Core) unit and executes a model of the physical system. The simulator's execution time is strongly coupled to the timing service of the underlying operating system (OS). As the OS uses multitasking, achieving hard real-time is difficult without the use of kernel drivers. However, soft real-time is achieved

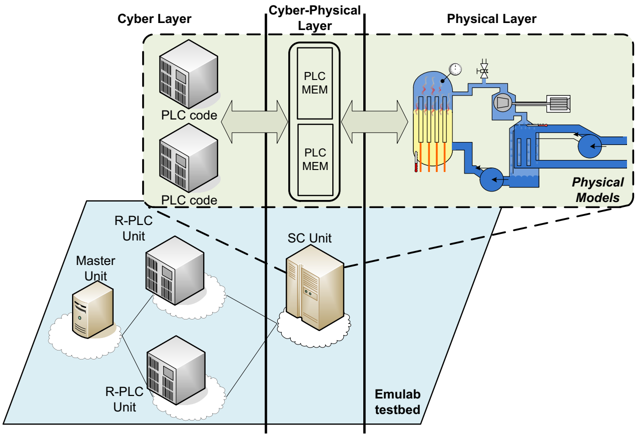


**Fig. 1.** Experimentation framework architectural overview

by allowing a certain deviation from the OS clock. The cyber layer is recreated by an emulation testbed that uses the Emulab architecture and software [8] to automatically and dynamically map physical components (e.g. servers, switches) to a virtual topology. Besides the process network, the cyber layer also includes the control logic code, that in the real world is implemented by PLCs. The control code can be run sequentially or in parallel to the physical model. In the sequential case, a *tightly coupled* code (TCC) is used, i.e. code that is running in the same memory space with the model, within the SC unit. In the parallel case a *loosely coupled* code (LCC) is used, i.e. code that is running in another address space, possibly on another host, within the *R-PLC* unit (Remote PLC). The cyber-physical layer incorporates the PLC memory, seen as a set of registers typical to PLCs, and the communication interfaces that glue together the other two layers. Memory registers provide the link to the inputs (e.g. valve position) and outputs (e.g. sensor values) of the physical model.

Prototypes of SC, R-PLC and Master Units have been developed in C# (Windows) and have been ported and tested on Unix-based systems (FreeBSD, Fedora and Ubuntu) with the use of the *Mono* platform. Matlab Simulink was used as the physical process simulator (physical layer). From Simulink models the corresponding 'C' code is generated using Matlab RTW. The communication between SC and R-PLC units is handled by .NET's binary implementation of RPC (called *remoting*) over TCP. For the communication between the R-PLC and Master units, we used the Modbus over TCP protocol.

## 4 Description of the Experimental Setup

### 4.1 Scenario

As pointed out by Cárdenas, *et al.* [14] attacks targeting the minimum/maximum value of parameters/control variables are the ones that can damage the process in relatively short time periods. Such attacks cause the accumulation of products (e.g. steam, water, fuel) by completely opening valves that feed products into process units and completely closing valves that free products from the process units. Our employed adversary model followed the same procedure to damage the physical process.

In the implemented scenario the attacker interacts with PLCs by sending legitimate Modbus packets to close/open specific valves. The attacker is located in the Internet and uses TCP connections to communicate with infected PLCs. Identifying the attack vector that could compromise the system to enable such a scenario is not the main focus of this study. However, we should also mention that the Stuxnet worm together with other studies such as the one performed by Nai Fovino, *et al.* [13], showed that such scenarios are possible in real settings. For instance, the attack reported in 2010 on Google's stations [15] is a clear example of how malicious software is able to exploit a Web browsers vulnerability in order to infect the entire corporate network of a large organization. Similarly for an industrial installation, once the malware is installed within the corporate network, it could spread to the process and control networks and it could compromise network protection mechanisms, e.g. firewalls, in order to give access to an adversary, i.e. *coordinator*, located outside the system.

The main target of the attacker was a power plant, integrated into our framework with the Boiling Water Power Plant (BWPP) model developed by Bell and Åström in [9]. This models a 160MW oil-fired electric power plant based on the Sydsvenska Kraft AB plant in Malmö, Sweden. Within the context of this model the attacker is able to control 3 valves: fuel valve, steam valve, and feed water valve. The desired critical state is given by the value of the pressure inside the steam drum. In other words, the goal of the attacker is to increase the pressure up to a specific value, representing the critical state, which can cause the plant to fail, shut down or even explode. The attacker achieves his goal by infecting PLCs that control the 3 valves and by coordinating the attack with packets sent remotely to each PLC.

### 4.2 Experimental Setup

The following experiments were implemented in the Joint Research Centre's (JRC) Experimental Platform for Internet Contingencies (EPIC) laboratory. The Emulab testbed included nodes with the following configuration: FreeBSD OS 8, AMD Athlon Dual Core CPU at 2.3GHz and 4GB of RAM. As shown in Fig. 2 the experimental setup consisted of 6 hosts, 1 host for running the SC unit, 3 hosts for running the R-PLC units, 1 host for running the Master unit and 1 host to run the malicious coordinator software. Within the Emulab testbed we emulated communication delays, packet losses and background traffic in order to recreate a dynamic and unpredictable environment such as the Internet. For emulating communication delays and packet losses we used *Dummynet* and for the background traffic we used UDP packets generated with *iperf*. Dummynet and iperf are running on the malicious coordinator and the infected Master unit, as shown in Fig. 2. Additionally, we used two 10Mb/s networks to emulate the limited bandwidth in the Internet (*Lan2*) and the communication limitations of PLCs (*Lan1*). The communication between R-PLCs and the SC unit was implemented with a 100Mb/s Lan (*Lan0*) to provide maximal performances for the interaction between R-PLC units and the BWPP model.

The main role of the SC unit was to run the BWPP model and to enable its interaction with the other components. Within the previously described scenario, each R-PLC unit controls a specific valve. Thus, R-PLC unit 1 controls the fuel valve, R-PLC unit 2 controls the steam valve and, finally, R-PLC unit 3 controls the feed water valve. The attack initiation commands are transmitted by the malicious coordinator using TCP connections and are forwarded by the infected Master unit as Modbus packets. This way, we emulate the functionality of other infected units in the process network that collaborate with the coordinator to succeed in the execution of the attack.

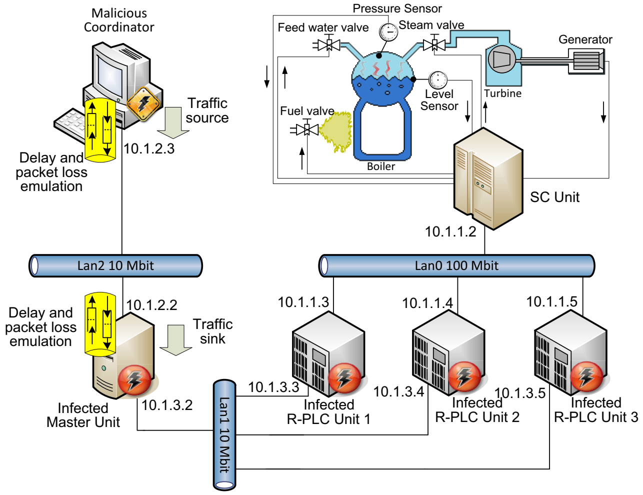


**Fig. 2.** Experimental setup

## 5 Attacks and Analysis

In this section we show that communication delays, packet losses and background traffic have a major impact on the success of coordination-based attacks. For this purpose we consider two settings. In the first setting the goal of the attacker is to bring the BWPP into a critical state where the value of the pressure is 249.7285 kg/cm$^2$. This is more than twice the value of a typical operating point (i.e. 105.006 kg/cm$^2$) and was obtained by running the model for a total of 260s with the fuel valve completely opened, the steam valve completely closed and the feed water valve set to 0.433. Consequently, in order to bring the BWPP into this state, the attacker needs to open the fuel valve exactly 79s before closing the steam valve, while running with the feed water valve set to 0.433 at all times. The attacker also calculates that after receiving the initiate commands, PLCs need to run the malicious code for 3 minutes in order to bring the BWPP into the critical state. For the second setting we consider that lower *precisions*, i.e. deviations from a fixed steam pressure value, can also bring the plant into a critical state. We show that this consideration increases the attacker's success rate, however, for extreme settings of communication delay, packet losses and background traffic, the coordinated attack still represents a challenge to the attacker.

The parameters we consider for the following experiments are communication delays, packet losses and background traffic. For communication delays we used the following values: 0s, 0.1s, 0.5s, 1s, 3s, 6s and 9s. For packet losses we used two rates: 1% and 5%. Finally, for background traffic we used: 2.5Mb/s, 5Mb/s, 7.5Mb/s and 10Mb/s. For each configuration setting, representing a combination of communication delay, packet loss rate and background traffic we ran 20 experiments, with a total of 1120 experiments executed in 112 hours.

### 5.1 Effect of Communication Delays and Packet Losses

Communication delays and packet losses between the coordinator and the compromised Master unit were emulated with the *Dummynet* software. We emulated 7 different delays up to 9s and two different packet loss rates: 1% and 5%. As previously mentioned, we assumed that the critical state includes a steam pressure of 249.7285 kg/cm$^2$.

Within this context we measured a maximal success rate of 70% and a minimal success rate of 0%. The results show that even for zero communication delays the average success rate does not exceed 51.25%. More specifically, this means that from 20 attempts, an average of 10 attempts will fail to bring the BWPP into the desired state. What is even more surprising is that for the majority of cases we measured a higher success rate for a larger loss rate. An explanation for this behavior is the reduced number of packets that are sent by the coordinator as opposed to the number of packets generated for the background traffic. As the number of packets transmitted between stations also affect the delay between packets transmitted by the coordinator, a higher packet loss rate translates to a more reduced number of packets and effectively to smaller delays between coordinator packets. However, this statement is only valid for delays smaller than 1s. For larger delays the success rate drops to 0% as the critical state can not be reached even after PLCs receive the initiate commands. These results are depicted in Fig. 3.

| | **Attack success rate** | |
|---|---|---|
| **Delay** | **1% packet loss rate** | **5% packet loss rate** |
| 0s | 45% | 51.25% |
| 0.1s | 20% | 26.25% |
| 0.5s | 40% | 40% |
| 1s | 45% | 31.25% |
| 3s | 0% | 2.5% |
| 6s | 0% | 0% |
| 9s | 0% | 0% |

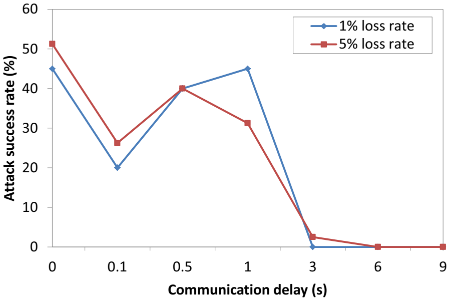
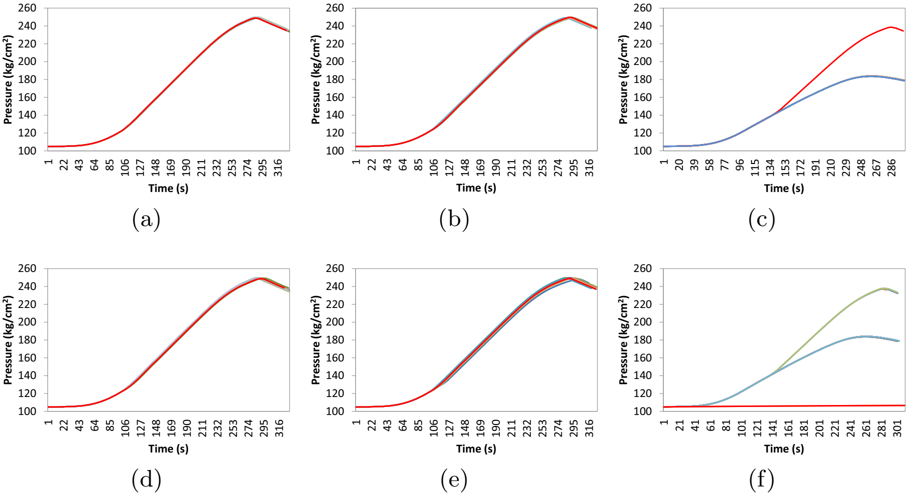


**Fig. 3.** Effect of communication delays and packet losses on the attack success rate (average background traffic)

For a better understanding of the behavior of the physical process, in the following we provide several figures illustrating the steam pressure for 6 different settings. The behavior of the process for communication delays of 0s, 1s and 9s and packet loss rates of 1% and 5% is shown in Fig. 4. As shown in Fig. 4 (b), a delay of 1s introduces only small variations that are barely visible. On the other hand, larger delays such as 9s illustrated in Fig. 4 (c), lead to connection time-outs that in turn cause a successful execution of commands in only one experiment (out of 20), that was illustrated with a red line. In order to illustrate the effect of a 5% loss rate we have also included Fig. 4 (d), (e) and (f).

For this setting variations are more visible. Nevertheless, in case of Fig. 4 (d) and (e) more than 50% of the attacks are successful. For a 9s delay (Fig. 4 (f)) we notice a setting marked with a red line in which none of the PLCs receive the commands to initiate the attack.



(a) (b) (c)

(d) (e) (f)

**Fig. 4.** Effect of delays and packet losses on the steam pressure for a constant background traffic of 2.5Mb/s: (a) 1% packet loss and 0s delay; (b) 1% packet loss and 1s delay; (c) 1% packet loss and 9s delay; (d) 5% packet loss and 0s delay; (e) 5% packet loss and 1s delay; (f) 5% packet loss and 9s delay

By analyzing the previous results we realize that achieving a 100% success rate for a fixed pressure in a limited time interval is a difficult task. As the attack scenario is highly time critical, emulated network delays and packet losses introduce additional delays to the already existing ones caused by communications and OS task switches. Based on these facts we can clearly state that a coordinated attack launched from outside a power plant has a low success rate (an average of 51.25% for 0s emulated delay) in case of time-critical scenarios.

### 5.2 Effect of Communication Delays and Background Traffic

The *iperf* software was used to generate UDP background traffic with four different configurations: 2.5Mb/s, 5Mb/s, 7.5Mb/s and 10Mb/s, that was the maximum capacity of *Lan2* (see Fig. 2). This way we were able to simulate real Internet conditions with a permanent background traffic that could also introduce additional delays and thus interfere with the outcome of the attack.

In order to analyze the effect of background traffic on the attack success rate we assumed that the critical state includes the previously discussed steam pressure of 249.7285 kg/cm$^2$. Additionally, for every configuration we considered an averaged packet loss rate with the results shown in Fig. 5. As expected, the background traffic also influences the attacker's success rate. We clearly see that the highest success rate is achieved for 2.5Mb/s followed by 5Mb/s. A background traffic of 10Mb/s introduces larger delays in *Lan2* (with a 10Mb/s capacity) and reduces the success rate to maximum 20%.

| **Delay** | **Attack success rate** | | | |
|---|---|---|---|---|
| | **2.5Mb/s traffic** | **5Mb/s traffic** | **7.5Mb/s traffic** | **10Mb/s traffic** |
| 0s | 52.5% | 67.5% | 55% | 17.5% |
| 0.1s | 32.5% | 25% | 15% | 20% |
| 0.5s | 52.5% | 45% | 47.5% | 15% |
| 1s | 52.5% | 52.5% | 27.5% | 20% |
| 3s | 0% | 0% | 0% | 5% |
| 6s | 0% | 0% | 0% | 0% |
| 9s | 0% | 0% | 0% | 0% |

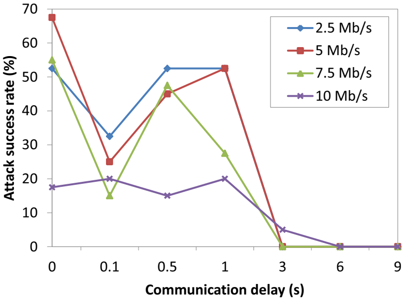
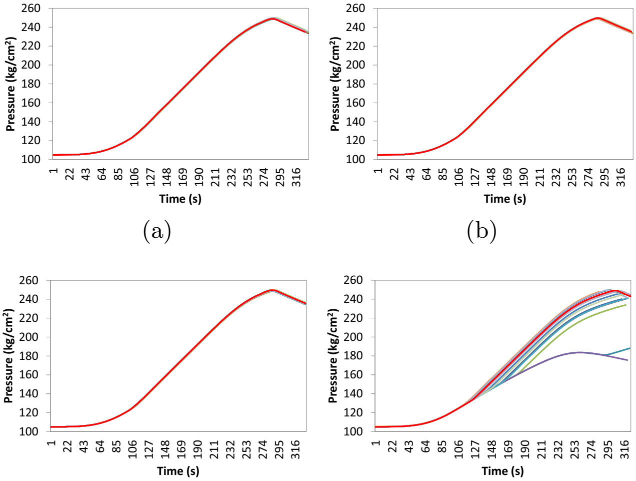


**Fig. 5.** Effect of communication delays and background traffic on the attack success rate (average packet loss rate)

The behavior of the plant in terms of steam pressure for each background traffic configuration is shown in Fig. 6 (a), (b), (c) and (d). In order to illustrate the effect of background traffic we considered a communication delay of 0s and a 1% packet loss rate in all four settings. These figures clearly show that a background traffic lower than 10Mb/s does not have a major impact on the behavior of the plant. The explanation for this is the low number of commands the attacker needs to send to the remote PLCs in order to initiate the attack. Furthermore, if we compare the effect of background traffic (Fig. 6) with the effect of packet losses (Fig. 4) we realize that packet losses have a greater impact than background traffic. Nevertheless, by increasing the background traffic to 10Mb/s the impact becomes immediately visible as the additional delays affect the timing of the commands received by each PLC. Based on these results we can clearly state that the impact of background traffic on the attack success rate is mainly minor. However, for a background traffic that is close to the network capacity the success rate drops to 20% or even 5% for delays larger than 3s.

### 5.3 Effect of Lower Attack Precisions

In the previous sub-sections we assumed that the target steam pressure of the attack is fixed to 249.7285 kg/cm$^2$. For this setting we measured an average success rate of 51.25% (for 0s emulated delay), with a minimal success rate of 0% and a maximal success rate of 70%. The previous results have also shown that reaching a fixed critical state is a rather difficult task for a coordinator located outside the power plant. However, if the target steam pressure does not require such a high *precision*, i.e. deviation from a fixed value, then the attacker's success rate could suffer major changes. Fig. 7 illustrates these changes in terms



(a) (b)

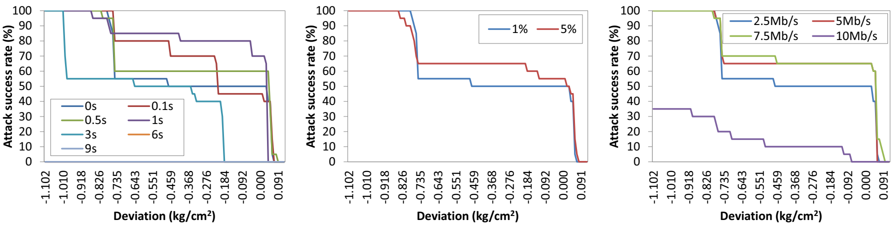
(c) (d)

**Fig. 6.** Effect of delays and background traffic on the steam pressure for a constant delay of 0s and a constant packet loss rate of 1%: (a) 2.5Mb/s background traffic; (b) 5Mb/s background traffic; (c) 7.5Mb/s background traffic; (d) 10Mb/s background traffic

of communication delays (Fig. 7 (a)), packet losses (Fig. 7 (b)) and background traffic (Fig. 7 (c)).

As shown in Fig. 7 (a), a critical state with a lower precision increases the attacker's success rate up to 100%, for a precision of 1 kg/cm$^2$. Nevertheless, larger delays (3s) still have a negative effect on attacks. More specifically, a delay of 3s decreases the success rate to 55%, while delays of 6s and 9s decrease the success rate to 0%. The effect of packet losses are also negligible if we consider lower precisions, as shown in Fig. 7 (b). In this case also the success rate increases up to 100% for a precision of 1 kg/cm$^2$. The same figure also shows that for the majority of cases a 5% loss rate leads to a higher success rate. As already mentioned in the previous sub-sections the reason behind this behavior is the low number of packets that the attacker uses to initiate the attack, as opposed to the high number of packets available for the background traffic. Finally, as shown in Fig. 7 (c), the effect of background traffic seems to be the most persistent even for lower precisions, as the success rate remains below 40% for a background traffic of 10Mb/s. Nevertheless, the attacker is able to achieve a 100% success rate for a precision of 1 kg/cm$^2$ and a lower background traffic.

The results from this sub-section have shown that if the critical state allows a slight deviation from the fixed steam pressure then the success rate of the attacker increases dramatically. However, the success rate is still affected by specific delays, packet losses and background traffic, as all of these parameters directly affect the timing between packets. Furthermore, extreme configurations still manage to decrease the success rate from 100% to below 40%.



(a) (b) (c)

**Fig. 7.** Effect of various attack precisions on the attack success rate: (a) communication delays; (b) packet losses; (c) background traffic

## 6 Concluding Remarks

In this paper we have analyzed the effects of network parameters on coordinated attacks against a Boiling Water Power Plant (BWPP). The experimental results prove that a coordinated attack where timing between commands is critical has a low success rate (an average of 51.25% for 0s emulated delay). Furthermore, such attacks are highly sensitive respect to communication delays, packet losses and background traffic. Nevertheless, the attacker's success rate increases significantly if the critical state allows a certain deviation from the target parameters. The experimental results also show that while a small deviation might increase the success rate up to 100%, there are configurations in which even these do not ensure a 100% success rate. Such configurations include communication delays larger than 3s and a high network background traffic, close to the network capacity.

## References

1. Nai Fovino, I., Carcano, A., Masera, M., Trombetta, A.: An experimental investigation of malware attacks on SCADA systems. International Journal of Critical Infrastructure Protection 2(4), 139–145 (2009)
2. East, S., Butts, J., Papa, M., Shenoi, S.: A Taxonomy of Attacks on the DNP3 Protocol. In: Palmer, C., Shenoi, S. (eds.) Critical Infrastructure Protection III. IFIP AICT, vol. 311, pp. 67–81. Springer, Heidelberg (2009)
3. The Symantec Stuxnet Dossier (2010), `http://www.wired.com/images_blogs/threatlevel/2010/11/w32_stuxnet_dossier.pdf`
4. Tan, Y., Sengupta, S., Subbalakshmi, K.P.: Analysis of Coordinated Denial-of-Service Attacks in IEEE 802.22 Networks. IEEE JSAC Special Issue on Cognitive Radio Networking and Communications 29(4), 890–902 (2011)
5. Botnet Twitter Attack (2009), `http://www.usatoday.com/tech/news/2009-08-06-twitter-attack_N.htm`
6. Genge, B., Siaterlis, C., Nai Fovino, I., Masera, M.: A Cyber-Physical Experimentation Environment for the Security Analysis of Networked Industrial Control Systems. Computers and Electrical Engineering 38(5), 1146–1161 (2012)

7. Nai Fovino, I., Masera, M., Guglielmi, M., Carcano, A., Trombetta, A.: Distributed Intrusion Detection System for SCADA Protocols. In: Moore, T., Shenoi, S. (eds.) Critical Infrastructure Protection IV. IFIP AICT, vol. 342, pp. 95–110. Springer, Heidelberg (2010)
8. White, B., Lepreau, J., Stoller, L., Ricci, R., Guruprasad, S., Newbold, M., Hibler, M., Barb, C., Joglekar, A.: An integrated experimental environment for distributed systems and networks. In: Proc. of the Fifth Symposium on Operating Systems Design and Implementation, pp. 255–270 (2002)
9. Bell, R.D., Åström, K.J.: Dynamic models for boiler-turbine alternator units: data logs and parameter estimation for a 160MW unit. Lundt Institute of Technology. Report TFRT–3192, Sweden (1987)
10. Queiroz, C., Mahmood, A., Hu, J., Tari, Z., Yu, X.: Building a SCADA Security Testbed. In: Proc. 3rd NSS, pp. 357–364 (2009)
11. Davis, C.M., Tate, J.E., Okhravi, H., Grier, C., Overbye, T.J., Nicol, D.: SCADA Cyber Security Testbed Development. In: Proc. NAPS, pp. 483–488 (2006)
12. Chabukswar, R., Sinopoli, B., Karsai, G., Giani, A., Neema, H., Davis, A.: Simulation of Network Attacks on SCADA Systems. First WSCS (April 2010)
13. Nai Fovino, I., Masera, M., Guidi, L., Carpi, G.: An Experimental Platform for Assessing SCADA Vulnerabilities and Countermeasures in Power Plants. In: Proc. HSI, pp. 679–686 (2010)
14. Cárdenas, A., Amin, S., Lin, Z.S., Huang, Y.L., Huang, C.Y., Sastry, S.: Attacks Against Process Control Systems: Risk Assessment, Detection, and Response. In: Proc. ASIACCS, pp. 355–366 (2011)
15. Google Aurora attack (2010), http://www.wired.com/threatlevel/2010/01/operation-aurora/

# The Role of the DNS in the Secure and Resilient Operation of CIs, the Energy System Example

Igor Nai Fovino, Salvatore Di Blasi, and Andrea Rigoni

Global Cyber Security Center (GCSEC),
Viale Europa 175, 00144, Rome, Italy
igor.nai@gmail.com

**Abstract.** The pervasiveness of Information and Communication Technologies in the control and governance of Critical Infrastructures (CIs) (e.g. power plants, energy grids, oil pipelines etc.) makes the Cyber Security problem a matter of citizen protection and safety. In this work, taking as example the Power System, we analyze the impact of malicious attacks agains the Domain Name System (DNS) on the operation of the modern, open and distributed critical infrastructures.

**Keywords:** DNS, Security, Power System.

## 1 Introduction

We define as Critical Infrastructure a system having a strong impact on the daily life of a citizen and that, if damaged, might put in danger the safety and the security of the citizen. Examples of critical industrial infrastructure includes Power Plants, Energy Grids, Gas Pipelines, Chemical Installations etc.

Those infrastructures are increasingly incorporating in a massive way *Information and Communication Technologies* (ICT). This trend obviously allowed to enhance and optimize the services provided, to implement distributed self-orchestration mechanisms, to manage remote installations in efficient manners. As a result, we can state that:

- The ICT infrastructure used to realized these services must be considered now as part of the critical infrastructure by itself.
- Since several of these services take advantage of the public network to operate, also the public network and its core elements have became a critical infrastructure.

In this work we concentrate our attention on the core role of the Domain Name System (i.e. the world wide system allowing to the Internet to operate correctly) in the secure and resilient operation of CIs and on its disrupting effect as a likely target in cyber attack scenarios.

On the light of the coming Energy Smart Grid, a continental critical infrastructure making massive use of networking services, we have taken into consideration as use case the Energy System and after analyzing at high level its functional structure, we magnified the effects of some classes of DNS vulnerabilities on the whole operation capabilities of this system.

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 152–163, 2013.

## 2 Related Works

The ICT security of critical industrial systems is a relatively new field of research. In this context, Adam and Byres [1] presented an interesting high level analysis of the possible threats affecting industrial critical infrastructures.A more detailed work on this topic is presented by Chandia et al. [2]. Some work has been done about the security of industrial communication protocols: for example the DNP3 User Group proposed a "Secure DNP3" implementing authentication mechanisms for certain type of commands and packets. Pothamsetty and Franz (CISCO), released a ModBUS transparent firewall [4] based on Linux Netfilter, however, at the moment it still appears to be in an embryonic stage of development. With specific reference to the ICT security of energy systems, Nai et al. presented an analysis of the cyber vulnerabilities of turbogas power plants [5][6], and a set of cyber-attack scenarios aimed at taking the control of the process network of an energy system [7]. Finally, in 2010 the case of Stuxnet, a malware conceived to directly hit the field devices of nuclear power plants [8], brought definitively under attention of the public opinion the strict interconnection between the security of ICT devices and the security of critical infrastructures. Regarding the Domain Name System, in [11] the authors draw a comprehensive picture of current threats affecting DNS, identifying on a coarse grained fashion data corruption, availability and information exposure issues. In [12] a serious global vulnerability has been discovered at protocol level, showing how DNS resource records (RRs) integrity can be seriously menaced, not having in place any authenticity check mechanisms; DNSSEC [13] has been introduced as a security extension to the DNS protocol, to provide authentication and integrity to DNS data. Despite of the commitment of the DNS community in gradually adopting DNSSEC, there are some open challenges yet to be addressed from both operational and administrative perspective, as evidenced in [14] and [15].

## 3 Energy System Overview

The Energy System comprises a huge number of subsystems, with different missions, collaborating to maintain a sort of cross-country balance and to provide energy to hundreds of million of people. In the following we provide an high level description of its most important elements and dynamics.

The physical layer of power systems is represented by the network hardware: stations, lines, transformers and circuit breakers. The control strategies maintaining the transmission system operating are transferred to the physical systems through ICT control and communication centers and devices (cyber layer of the system). From a physical point of view it is possible to categorize the elements constituting the power system in:

- *Transmission Stations*: generally operated directly by the Transmission System Operator (TSO).
- *Power Plants*: usually owned by different companies.

- *Distribution Systems feeders*: these are buses, equipped with transformers, in which a Medium voltage distribution system is originated. Each Distribution System Operator (DSO) owns and operates as a monopolist the distribution system over a certain portion of territory.
- *Large Utilizer*: energy users that demand high power (> 5 MW).
- *End Users*: they are connected to the distributions buses and contitute the leaves of the energy system. With the advent of the modern smart-grid in which each end user can also be an energy producer of course this categorization will change.

To be maintained, such complex system need to exchange a considerable amount of information (real-time data, but also commercial and administrative data) between control centers and substations, and between the different operators.

The cyber-layer of an energy grid is composed of different subsystems:

- *Control Network*: it contains all the Remote Terminal Units (in the following RTU) and Programmable Logic Controllers (in the following PLC). It is directly interfaced with the field network, i.e. the network of actuators and sensors that physically perform the process tasks on the system. Moreover, it is connected with the Process Network described in the following.
- *Process Network*: it is composed of the SCADA servers and all the other systems that gather the data coming from the Control Network and send commands to the Control Network.
- *Exchange Area*: this area usually contains aggregation databases that receive data from the process network. Such data represent the working state of the system and are used by the diagnostic systems also contained this area, to detect anomalies. The operators of the control centers remotely access such databases in order to have a high level view of the process state.
- *Control Centers*: these areas, usually composed of systems that act as remote Human Machine Interfaces (HMI), are used by the operators to obtain information about the process and eventually perform operational sequences to modify the process state.

This view of the system has to be interpreted in a multilayered fashion, where several of these infrastructures owned by different companies, interact in an interleaved manner at different levels.

## 4 Domain Name System Overview

The Internet is the world's largest computing network. It maintains two namespaces, the IP address system and the Domain Name hierarchy. The Domain Name System (DNS) is in charge of maintaining the Domain Name space and provides the services allowing to map domain names on the correspondent IP addresses. DNS can be considered, at the same time, an Internet critical system, a service, a protocol and information infrastructure.

The DNS infrastructure is composed of entities, geographically and logically organized in a hierarchical shape: the topmost level in the hierarchy is the root domain, represented as a dot ("."), while the next level is called the top-level domain (TLD). Each TLD, in turn, can have many sub-domains, called second-level or enterprise-level domains.

Each of these entities have authority over a portion of the domain name space: those associated to the root domain are called root operators; the organizations that run name servers related to a TLD are called registries. Country Code TLDs (ccTLDs) are run by registries designated in the respective countries, and gTLDs are run by global registries.

To facilitate this administration process, the DNS defines the concept of 'zone', which is an administrative building block of the DNS name space, typically used to refer to a domain managed as a single administrative entity (e.g. the root zone, the .com zone).

DNS functions can be mainly summarized in:

– **DNS query/response:** This is the most known and used transaction in DNS. A query originates from a client component, known as stub resolver or resolver, towards either an authoritative or caching name server (the process can be either iterative or recursive). Query/response data are normally sent in plain text thus letting a potential attacker the possibility to intercept and alter response information back to end-users.
– **Zone management:** A zone transfer represents an operation where a secondary slave server refreshes the entire contents of its zone file from the primary master servers. This process enables a secondary name server to keep its zone file in synchronization with its primary name server. A zone transfer process has different security implications because it can expose some more information than a normal query and because it can trigger an increased resource usage of the message for a potential attacker.
– **Dynamic services:** Through this service it is possible to dynamically add/delete a subset of the Resource Records (RRs) for an existing domain, to delete an entire domain or to create new domain.
– **DNS Administration:** This includes all the administrative tasks performed by the responsible entity in order to guarantee an appropriate level of service and assure security.

## 5 The Role of DNS in Energy System Operations

The previous sections provides an overview of the ICT architecture of the Power System. It is evident how, in a similar infrastructure, a relevant role is played by the ICT networks making the system interconnection possible. Looking at the scientific and operative literature it comes out that in this context, very little attention is paid to the role of the Domain Name System. To understand the deepness of the involvement of the DNS infrastructure in the Power System operation, we have partitioned the system in two views: the "high level infrastructure" and the "low level infrastructure". For each of these views, we have identified a set of operation classes. We have then taken into consideration some classes of vulnerabilities traditionally associated with the DNS system: *Repository Corruption, System Corruption, Protocol issues, Denial of Service and Information Exposure.*

On the basis of these classes we have made some high level speculation on the effects of the failure of the DNS on the Energy System.

### 5.1 DNS and the Power System High Level Infrastructure

We can define as "high level infrastructure" of the Power Framework, the infrastructure used for the so-called high-level operations: *(1) Management of the energy market*, *(2) Links between industrial actors and end users*, *(3) Actions at the customers' premises*, *(3) Links between the power sector and industrial actors*, *(4) Coordination among Power producers*, *(5) Coordination among transmission companies* , *(6) Management of crisis/blackout.*

Each of these functional operations involve in some way the DNS. In the following we provide an overview its role and of the effects of a possible failure.

**Management of the Energy Market**
It includes the interactions between the industrial actors, brokers, and the wholesale market and market clearinghouse. The aspects of these high level operations are, technologically speaking, very close to the traditional Web Application scenarios. It is then evident how DNS plays a relevant role and how its failure can directly impact the availability and stability of the energy market, possibly causing serious financial damages.

With reference to the four classes of vulnerabilities associated to the DNS here we describe briefly some threat scenarios:

- *Repository Corruption*: a DNS repository corruption (e.g. authoritative or cache database corruption) can be part of some more complex attack aiming at rerouting part of the energy market data flow to fake servers, in order to alter the perception of the market trend. In other scenarios, this might also impact the energy production, for example, if an energy producer buys an energy stock on the market, where this is done through dedicated servers, accessing a fake server as the result of a DNS repository corruption. The effect of these operations might have a country level or, even worst, a continental level repercussion, both economically and socially speaking (if the lack of energy forces grid shutdowns and energy cuts).
- *System Corruption and protocol issues*: the same considerations done in the previous case can be made also in these two cases.
- *Denial of Service*: DNS DoS might cause the unreachability of the Energy Market network infrastructure. The impact of this attack, being immediately evident, would be limited, since for a certain amount of time each actor of the energy market can operate without needing the access to the market services. It is however true that in particular cases (e.g. during unexpected peaks of energy requests or similar situations), the unavailability of the energy market could cause unpredictable damages.

- *Information Exposure*: attacks to the DNS aiming at violating the confidentiality of the infrastructure might be part of more complex attacks (see for example the attacks aiming at corrupting the DNS cache). The immediate damage here is mostly null, but if we consider the "big picture", learning how certain DNS nodes involved in the energy market operation are configured might give a powerful knowledge to potential attackers.

**Links between Industrial Actors and End Users**

This logical operation refers to all the communication phases between energy companies and end-users including meters, billing energy services interface, aggregators of retail energy providers and energy service providers.

An example where DNS might be used here is in the context of smart meters: there already exist several examples of metering infrastructure composed by a mixture of GPRS technologies and classic TCP/IP channels. Normally the communication is "GPRS Based" from the meter to the local aggregation center and "IP based" from the local aggregation center to the Energy enterprise servers. With regards to the IP part of the data control and acquisition architecture, any attack on the DNS can have impact on services such as: remotely turning power on or off to a certain customer, alter energy usage information, disabling service outages detection, favouring unauthorized use of electricity, alter the maximum amount of electricity that a customer can demand at any time, remotely altering the meter's billing plan. Mobile applications already exist that allow consumers to check home energy consumption remotely; also in this case it is probable that DNS is used to make the service accessible from anywhere. In the same way, billing services, just another example of a web application architecture, make use of DNS, to make the frontend servers and the payment servers accessible to users.

DNS Repository Corruption, DNS System corruption and protocol issues, in this case, as a part of a more complex attack, can be used in several scenarios:

- The DNS cache can be corrupted so that it would be possible, at some point in between meters and aggregation servers, to reroute the traffic to a fake server. At the same way these classes of vulnerabilities can be used in a scenario in which the billing process is involved or, in the case of end-user energy production, it can be used in attacks aiming at altering the end-users production records. The damage here would be mostly economical, but, luckily, it would not impact core installations.
- *Denial of Service*: DNS DOS might interfere in the metering and billing process. Again we can speak mostly of economical damages.
- *Information Exposure*: as in the previous caset, the immediate damage here is mostly null, but if we consider the "big picture", this scenario might surely consist in an intermediate step for other attacks.

**Actions at the Customers' Premises**

In this context we consider operations such as management of appliances, electric vehicles, other related services (gas/water metering), home automation etc.

These operations fall, technically speaking, in the same class of the previous operations, and for that reason the DNS, depending on the underlying communication architecture, might have a relevant role.

**Links between the Power Sector and Industrial Actors**
To maintain their core infrastructure (power plant, transmission centers etc.), energy companies are tightly linked with the power device producers. It's quite common for the device producers to provide remote maintenance support. This, normally, is implemented by the establishment of a VPN connection (through the public Internet) from the device producer home site to the power company installation sub-network, allowing remote operation of the local process control system. In this case, DNS is involved in the resolution of the names of the servers involved, and any unavailability, corruption or disruption might prevent a required maintenance operation on the physical installation. When a site-to-site VPN tunnel is established, the name resolution process of internal name servers is achieved through the two DNS systems acting at both sites: misconfigurations, internal corruption or availability issues on either the producer or the energy company site affect inevitably the whole system security, with the consequence that operations flow can be redirected to bogus servers or can be prevented at all. It is also important to understand how DNS can play the twofold role of infection dozer (e.g. by transparently redirecting users request to fake sites and consequently triggering a silent installation of malicious code which will then produce the real damages, when in action) and infection actuator (for example by directly impacting the availability of the company installation sub-network services).

**Coordination among Power Entities**
In this category we consider:

- The coordination among power companies, mainly related to the amount of energy to be produced, which is increasingly making use of the Internet infrastructure. The impact of DNS on these operations might be then considered high over these information exchange services, as for example power companies might not be able to communicate properly energy production plan details to each other.
- The Coordination among transmission companies: the same considerations of the previous point are valid here.
- Management of crisis/blackout: traditionally the coordination among energy actors during crisis (e.g. during a blackout) is well structured and defined by a set of operational policies. The use of the public network, mailing systems and other applications to coordinate the actions during an energy crisis are increasing. Also in this case, DNS might be involved. In this case the impact of DNS repository corruption, System Corruption and DoS is potentially heavy. A delay in the coordination of a blackout emergency can lead to dramatic situations where entire countries are left without energy. This can be considered, at this level, the most sensible operation in term of impact on the citizen life.

### High Level Layer Impact Conclusions

Essentially, all of these high level power infrastructure operations rely on web-services/applications making use of Internet to exchange information, perform transactions, and provide services. In all these cases, the DNS plays a relevant role. A failure or a corruption of the DNS might have a dramatic impact, for example affecting pricing or availability in the energy market. Similarly, if during the management of an energy crisis (e.g. blackout risks) the DNS fails, this might impact the high level control centers collecting field data, and indirectly slow down the definition of a proper contingency plan. The coordination among power producers is necessary to guarantee the stability of the energy grid. A failure of the DNS could impair this process.

## 5.2 DNS and the Power System Low Level Infrastructure

In the early '90s, the Power control system was considered a completely closed environment. The control of the field network was based on serial communication protocols and everything was monitored and managed locally. With the increasing use of TCP/IP, process engineers decided to port all the serial industrial protocols to TCP/IP (usually embedding these protocols as application layers within the TCP/IP suite). Today, basically every active element in the modern energy control system is associated with an IP address. Studies conducted in the field (see for example [1]) have shown how it is becoming more and more common for power systems to rely on the DNS for the resolution of the server involved in the control process.

Here some examples of common operational activities in which the DNS might be involved and some speculations on the effects of a DNS failure on these activities.

### Maintenance Operations

Power plants, transmission substations, and other power system elements require constant maintenance. These activities are typically outsourced to external companies. These companies perform several of the maintenance operations remotely. The standard procedure consists of:

1. Establishing a site-to-site VPN connection between the external company network and the network of the plant owner
2. Accessing the power company domain through a Radius authentication
3. Accessing the installation sub-network
4. Performing the required maintenance operation.

To resolve the addresses of the different servers involved in the process both internal and external DNS at both sites are normally used. A failure of the DNS during these operations, might impact the safety and stability of the power system. Repository Corruption and protocol issues (allowing for example to perform a DNS cache poisoning) can be used as part of complex attacks aiming at rerouting the maintenance flow between the device producers site and the local plant network site. DNS DoS can be used to make harder to establish a connection between the remote site and one of the server on which perform the maintenance operations.

The aims of these attacks can be twofold:

1. To cause a corrupted state of the real system while showing false data to the operator
2. To prevent a maintenance operation to be correctly performed

In both the two cases the impact of the installation might be extremely heavy. Dealing with critical devices such as gas turbines, high voltage lines, or in the worst case, nuclear power plant, a missed maintenance operation might have dramatic effects.

**Process Network Interactions**

The process network contains all the servers controlling the industrial processes (e.g. energy production, energy transmission etc.). It is quite common to rely on an internal DNS for the resolution of the server names. In this case, a failure of the DNS might impact the detection of anomalies in the process network of the power system or on the control capabilities of the SCADA servers. In the first case an undetected anomaly (for example a variation in the rotation of a gas turbine) can cause physical damage to the system and a system stop (a very expensive mistake given an average power plant costs around 2 million euros per day). In the second case, losing the control capabilities of the SCADA servers could make it impossible to react sufficiently rapidly to a change in system critical state.

**Operator Monitoring**

Human operators use the HMI to monitor the activities of the process system. To perform this activity, they often access the history servers contained in the exchange network. More rarely, they directly access the SCADA servers; finally, the trend of accessing servers and services using names instead of IP addresses is increasing. Moreover, in several situations these activities are performed remotely in the broader sense, i.e. from operators located in a completely different place, using an external network and relying on the Internet to reach the access point of the installation sub-network. Again, the DNS plays a role in making the connection possible as in the case of the maintenance operations, and again, its failure or its corruption might make it harder or impossible to control the process system remotely. In[7] Nai et al. show how a DNS poisoning attack could be used as part of a complex cyber attack against a turbo-gas power plant to re-route the operator on a false SCADA server.

**Control Center Operations**

Control centers manage simultaneously multiple installations of the Power System. The different applications hosted in the control centers generate query/response flows from the local HMIs to the remote RT-Databases of the installations and to the diagnostic servers. DNS is again used to resolve the name of the entry points of the different remote subnets, and to resolve the names of the remote servers. Another important function of control centers consists in delivering the daily production plans specifying the energy production, hour by hour for each power plant of the system. These plans are automatically delivered to each plant by using (a) a dedicated network (b) the public network in combination with the use of VPNs and MPLS features. A failure of the DNS here might have significant effects on the definition of reaction plans against energy crisis or might compromise the energy production plan.

In these last scenarios, further attention should be also paid to the role of DNS as a vehicle for establishing unfiltered covert channels with already infected hosts within the energy company sub-networks: exfiltration of data from control systems or exchange networks, be it measurement data, performance reports, operational plans or critical assets inventory can lead to severe security risks for the continuity of operations. Typically, even though company sub-networks are isolated from public networks, they need a set of basic services such as the resolution name service, and when a target host within a company sub-network is already compromised, data exfiltration can be achieved through forwarded DNS queries, which resolve to a nameserver actually under the attacker control; in this way, the malicious application, running over the infected machine, can send ad-hoc queries to a specific URL, which will issue for example the transfer of sensitive data archived in the sub-network to the attacker nameserver, without having application level firewalls or intrusion detection/prevention systems to actively log suspect HTTP traffic. Deep DNS queries and responses inspection should be ensured in order to mitigate this risk.

## 6 The DNS Health Measurement Framework

This use case demonstrates the need to develop and standardize metrics for what Security, Stability and Resiliency (SSR) of the DNS actually means. A similar set of metrics would be extremely useful in the power system context, to assess the SSR level of the DNS system involved in the power operations. The outcomes of the analysis of the SSR data would allow operators improve the understanding of the security level of their DNS infrastructure; moreover, a configurable and modular framework supporting "what-if" and impact analysis of DNS re-engineering and DNS policy making would again make easier to understand their potential effects on the power system.

In the context presented in the previous section, the efforts of Internet Corporation for Assigned Names and Numbers (ICANN) provided a highly useful foundation for further studies on the Security, Stability and Resiliency (SSR) of the DNS. The results of the ICANN DNS SSR symposium 2010[10] introduced the concept of "DNS Health" that includes the concept of DNS SSR.However, the definition of security metrics in the DNS remains at a primitive stage and metrics for DNS stability and resiliency are largely uncharted territory. The open points and unanswered questions we identify after a reasoned analysis of the report [10] are related to:

1. The need of viable indicators of DNS health and security for the different DNS actors (Root server operators, Operators of non-root authoritative name servers, recursive caches, open DNS resolver,end users);
2. The need to understand and refine proper methods and techniques for the measurement of DNS health and security indicators;
3. The need to refine and improve existing metrics (and measurement approaches) for coherency, integrity, speed, availability, resiliency, vulnerability and security;
4. The need for metric threshold levels that allow the DNS community to know, possibly in advance, when DNS health and/or security are being compromised.

Answer to these open questions is mandatory to define a global and coherent action to enforce at every level the security of the DNS providing at the same time to the critical end users the tools for evaluating their exposure to DNS threats.

## 7 Conclusion

For decades, considered a totally closed system, the powes system is now quickly evolving toward a completely open, heterogeneous, interconnected and distributed model. This Copernican revolution will deeply impact our society, introducing new economic models and new services. The backbone of this model will increasingly be based on ICT networks. In this context, it is evident how the DNS plays more and more a strategic role in maintaining reachability of all nodes of this large, distributed system. For that reason it will soon be necessary to assess and evaluate the security, stability and resiliency of those DNS elements providing services to this system. We envisions that it would be beneficial for all the actors of critical infrastructures impacted by the DNS, to have a broadly adopted, cooperatively achieved model for DNS Security, Stability and Resiliency (SSR) measurement and benchmarking based on the notion of DNS health. On the basis of the outcomes of this work, we are planning to design a layered and multi-perspective framework for the measurement and benchmarking of the DNS SSR level. This framework is intended to support risk analysis, what-if analysis and impact analysis of changes to the DNS infrastructure as well as DNS policy-making. The goal of this work will be to refine the current concept of DNS SSR and to enhance the awareness among the "critical" end-users of the DNS and among the private DNS operators.

## References

1. Creery, A.A., Byres, E.J.: Industrial Cybersecurity for power system and SCADA networks IEEE Industry Application Magazine (July-August 2007)
2. Chandia, R., Gonzalez, J., Kilpatrick, T., Papa, M., Shenoi, S.: Security Strategies for SCADA Networks. In: Goetz, E., Shenoi, S. (eds.) Critical Infrastructure Protection. IFIP, vol. 253, pp. 117–131. Springer, Boston (2008)

3. Nai Fovino, I., Carcano, A., Masera, M., Trombetta, A.: An experimental investigation of malware attacks on SCADA systems. International Journal of Critical Infrastructure Protection 2(4) (2009)
4. http://modbusfw.sourceforge.net/ (last access May 28, 2010)
5. Nai Fovino, I., Masera, M., Guidi, L., Stefanini, A.: Cyber Security Assessment of a Power Plant. International Journal of Electric Power System Research 81(2), 518–526
6. Leszczyna, R., Nai Fovino, I., Masera, M.: Security Evaluation of IT Systems Underlying Critical Networked Infrastructures. In: Proceeding of the 1st International Conference on Information Technology, Gdansk, Poland, May 18-21 (2008)
7. Nai Fovino, I., Masera, M., Guidi, L., Carpi, G.: An Experimental Platform for Assessing SCADA Vulnerabilities and Countermeasures in Power Plants. In: Proceedings of the IEEE 3rd International Conference on Human System Interaction, Rzeszow, Poland, May 13-15 (2010)
8. http://www.symantec.com/connect/blogs/stuxnet-introduces-first-known-rootkit-scada-devices (last access: May 14, 2011)
9. Nai Fovino, I., Masera, M., Leszczyna, R.: ICT Security Assessment of a Power Plant, a Case Study. In: Proceeding of the Second Int. Conference on Critical Infrastructure Protection, Arlington, USA (March 2008)
10. Measuring the health of the Domain Name System, Report of the 2nd Annual Symposium on DNS Security, Stability, & Resiliency, Kyoto, Japan (February, April 2010), https://www.icann.org/en/topics/ssr/dns-ssr-symposium-report-1-3feb10-en.pdf
11. Santcroos, M., Kolkman, O.: DNS Threat Analysis, NLnet Labs (May 2007)
12. Kaminsky, D.: It's the end of the cache as we know it. Blackhat, USA 2008 (August 2008), http://www.blackhat.com/presentations/bh-jp-08/bh-jp-08-Kaminsky/BlackHat-Japan-08-Kaminsky-DNS08-BlackOps.pdf
13. IETF RFC 2535 "Domain Name System Security Extensions", http://tools.ietf.org/html/rfc2535
14. Chandramouli, R., Rose, S.: Open issues in Secure DNS Deployment. US National Institute of Standards and Technology (NIST)
15. Osterweil, E., Zhang, L.: Interadministrative challenges in managing DNSKEYs. IEEE Security and Privacy: Securing the Domain Name System (September 2009)

# The Development of Warning, Advice and Reporting Points (WARPs) in UK National Infrastructure

Tony Proctor

School of Technology, University of Wolverhampton, Wulfruna Street, Wolverhampton, WV1 1LY, UK
t.proctor@wlv.ac.uk

**Abstract.** This purpose of this paper is to examine the development of Warning, Advice and Reporting Points (WARPs) as part of the information sharing strategy for UK National Infrastructure. It identifies and discusses the origins of the Computer Emergency Response Team (CERT) and Information Exchanges. It then reflects on the authors own experience of managing Warning, Advice and Reporting Points, defining and describing these important forums for information sharing in the UK information security community and beyond. One of the problems in protecting critical infrastructure is how to get the right information to the right people. The paper identifies key drivers for information sharing. It outlines the University of Wolverhampton involvement in the WARP programme since 2006 and the success that has been achieved creating and working with several WARPs in the public sector.

**Key words:** security information, information sharing.

## 1 Background

Delivering appropriate information to the right people is an essential aspect of critical infrastructure protection. This is equally applicable in both incident prevention and incident response. Two incidents, sixteen years apart identify the need for improved methods of information sharing.1988 experienced the first major internet incident, the Morris Worm. A report written by Purdue University [1] concludes that, the attack, "*should also point out that we need a better mechanism in place to coordinate information about security flaws and attacks. The response to this incident was largely ad hoc, and resulted in both duplication of effort and a failure to disseminate valuable information to sites that needed it" and "methods did not ensure timely, widespread dissemination of useful information"*. Sixteen years later, the report into the 9/11 attacks on the US identifies a failure in information sharing. The US Government [2] describes the biggest impediment as, *"the human or systemic resistance to information sharing"*. It describes the use of databases that might not normally be thought of as intelligence (e.g. customs or immigration) providing an *"immense storehouse of information"*.

The UK Government describes the sharing of information about the risks facing networks as, "beneficial to both government and industry".

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 164–174, 2013.

It describes mechanisms through which one company can learn from the experiences of others, "without fear of exposing company sensitivities" as being an opportunity for every participant to improve their level of assurance [3].

The increasing availability of electronic information combined with inter-organisational collaboration and sharing of services provide some of the other drivers for information sharing. But there are barriers to overcome in order to develop information security, information sharing.

A WARP is a community based service for sharing timely advice relating to information security threats, incidents and solutions. WARPs were developed by the Centre for the Protection of the National Infrastructure (CPNI) as part of their Information Sharing Strategy. They recognised the need to provide a cost-effective way to facilitate information security among a diverse range of organisations, many of which form part of the critical national infrastructure.

In 2007, the University of Wolverhampton in collaboration with West Midlands Police, created a WARP for Local Government in the region. WARP is a developmental project that has attracted both national and international attention. The challenge is for it to both develop as a concept and adapt to the changing needs of the members during a time of decreasing budgets.

## 2 The Development of Information Sharing

The report into the incidents of 9/11, supports establishing a culture where availability of information is defined not on a "need to know" but instead on a "need to share" basis. The report makes an interesting contrast between the penalties for over-classification of information (cost to the organisation) and the risk of sharing (criminal, civil and administrative sanctions). It recommends that procedures should provide incentives for sharing. This provides a better balance between "securing" and "sharing" information. It provides weight for this intention, identifying the President as the person to resolve the legal, policy and technical issues in order to create a trusted information network.

From a technological perspective, the report recognises that each organisation operates their own databases. It recommends that "horizontal searching" is available across agency lines and that the security remains protected by the design of the network and Information Rights Management (IRM).

In the UK, the demand for information sharing across the public sector continues to grow. In the Local Government Sector there is a requirement to share information with the health service, police and others. Some of the challenges that this presents are illustrated by Leicestershire County Council [4] who define an information sharing protocol for multiple partners. This helps to address one of the main issues affecting organisations who need to share information; establishing the rules for sharing.

### 2.1 The Emergence of the Computer Emergency Response Team (CERT)

The Morris Worm was created by Robert Morris, a student at Cornell University. In the Perdue University Report (described in section 1) it is stated that, "It is clear from the code that the worm was deliberately designed to do two things: infect as many machines as possible, and be difficult to track and stop. There can be no question that this was in any way an accident".

Developed for DEC hardware supporting the UNIX operating system, the replication of the Worm caused a denial of service to approximately 6 000 machines. This accounted for more than 10% of the internet at that time. The code allowed the Worm to replicate multiple instances on a single computer, resulting in a denial of service. The cost of the damage exceeded $10 million. Morris received a community service sentence and a 3 year probation order. It is interesting to consider what the penalty would be today for creating a 10% denial of service on the internet?

The US Government determined that a response was necessary in order to address future problems. In 1989 the first CERT (CERTCC – CERT Coordination Centre) was established in partnership with Carnegie Mellon University [5]. Other nations followed suit. In 1992 the UK Government created the Unified Incident Reporting and Alert Scheme (UNIRAS) [6]. The functions of this were to respond to electronic attack and other significant IT security incidents, warn about IT security incidents and vulnerabilities and to gather information relating to IT security incidents.

Today, the UK National "CERT" is formed by two organisations. GovCERTUK is operated by the Government Communications Headquarters (GCHQ). Essentially, GCHQ has overall responsibility for the .gov.uk domain and anything attached to it. The other organization which helps to provide a national CERT function is CSIRTUK, operated by CPNI (see 2.2). In addition to these national "CERTs", the Cyber Security Strategy of the United Kingdom [7] announced the creation of the Office of Cybersecurity (OCS) and Cyber Security Operations Centre (CSOC). The OCS provides strategic direction on cyber security and information assurance for the UK and works with private sector partners on exchanging information and promoting best practice. CSOC's primary role is to actively monitor and coordinate incident response. The key differentiator in role appears to be that one is "Strategic" and the other "Operational".

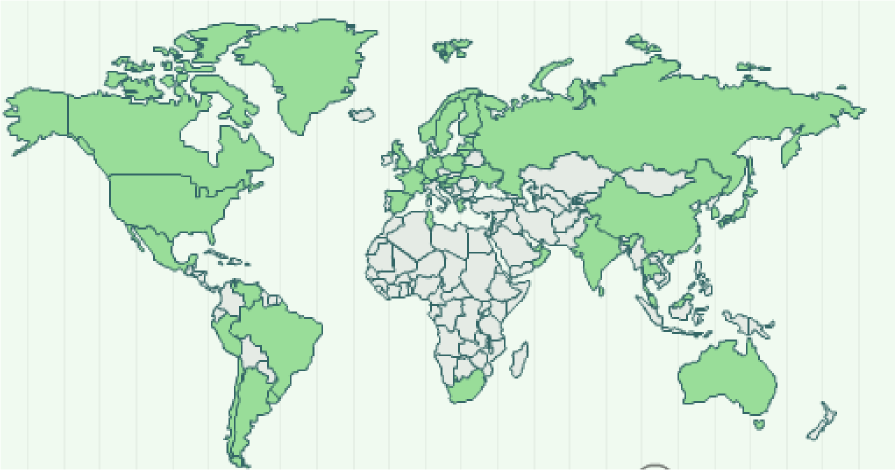


**Fig. 1.** Word Map of Computer Incident Response Teams (CERTs and CSIRT FIRST Members) FIRST [8]

The darker areas in Figure 1 identify many of the nations that operate CERTs and / or CSIRTs. CERTs and CSIRTs perform a similar role, discussed by GovCERT.NL [9]. US-CERT [5] describe themselves as, "providing response support and defense against cyber attacks for the Federal Civil Executive Branch (.gov) and information sharing and collaboration with state and local government, industry and international partners". WARPs have been compared to the "outreach component" of a CERT [10].

### 2.2 Centre for the Protection of the National Infrastructure (CPNI)

In 1999 the UK established the National Infrastructure Security Coordination Centre (NISCC). More recently renamed the Centre for the Protection of the National Infrastructure (CPNI), it is the UK Government body responsible for providing security advice to the businesses and organisations which make up the national infrastructure. They are the Communications, Emergency Services, Energy, Finance, Food, Government, Health, Transport and Water sectors [16]. CPNI's focus is primarily to defend against attacks from terrorist or other sources of an electronic, physical or personnel security nature.

### 2.3 Information Exchanges

As part of an information sharing strategy, CPNI operates Information Exchanges (IE). They are defined as, "a mechanism through which one company can learn from the experiences, mistakes, and successes of another, without fear of exposing company sensitivities" CPNI [3]. An Information Exchange is based upon the personal trust of representatives, sharing information in a confidential meeting. Representatives at Information Exchanges are expected to attend all meetings, which are held every two months. Meeting face-to-face is intended to build up a small, trusted community with a common interest. It is considered that strangers may inhibit the sharing of sensitive information. So each organisation is permitted a maximum of two representatives and substitutes cannot attend. At the time of writing, there are 12 information exchanges as shown in table 1.

**Table 1.** The 12 Sectors for which CPNI Operates Information Exchanges

| | |
|---|---|
| Aerospace and Defence Manufacturers | Financial Services |
| Communications Industry | Managed Service Providers |
| Personnel | Northern Ireland |
| Pharmaceutical Industry | Network Security |
| Supervisory Control and Data Acquisition (SCADA – critical infrastructure and Control Systems) | European SCADA |

### 2.4 WARP

A WARP or Warning, Advice and Reporting Point to cite its fullest definition, is a community based service for sharing up-to-date advice on information security threats, incidents and solutions. WARPs were developed by NISCC. They now form part of CPNI's Information Sharing Strategy. CPNI states that, "A WARP works because its membership is a community, based on geography, technology, business need or another area of common interest, CPNI [11]. On the ground, this means that a security concern of one member is probably a concern of the other members and their WARP is the most effective way of sharing information between them".

WARPs are an extension to the Information Exchange concept. They have fewer rules, can operate beyond the critical infrastructure sector and are independent (as they are not directly operated by CPNI). A WARP provides warnings, advice and is a place to which incidents may be reported. Warnings are most commonly distributed via email and are filtered, hence WARP members receive only relevant information. Advice is facilitated via a number of methods: directly from the WARP Operator, regular face to face member meetings, member to member discussions and a virtual network of experts that has been established. Table 2 summarises the processes that provide WARP functionality. WARPs have developed an appropriate structure for the dissemination of information: they can create links with their peers, share information with other WARPs and other relevant organisations (e.g. GovCERT, CPNI, and Ministry of Defence) nationally via the WARP Operators Forum.

**Table 2. WARP Function and Process**

| WARP Function | Process |
|---|---|
| Warnings | Daily issue of warnings, advisories and news via email (SMS, RSS and telephone may also be used) |
| Advice | Available via email and telephone. Self-help advice is facilitated by discussion in regular face to face meetings |
| Reporting Point | Incidents are discussed in the regular meetings. Members may also report incidents via email or telephone |

## 3 The University of Wolverhampton as a WARP Operator

Following a WARP presentation delivered by NISCC, West Midlands Police approached the University of Wolverhampton with a view to creating a WARP for the West Midlands region. This partnership was joined by the Local Government Association for the region.

Initial funding was provided by the Higher Education Innovation Fund (HEIF) and the Office of the Deputy Prime-Minister (who created a fund to develop WARP in the Local Government sector). The first WARP established by the University was specifically for the Local Government community in the West Midlands region of the UK. West Midlands Councils WARP was officially launched by the e-Government Minister at the end of 2006. Reference to "UoW WARP" in the remainder of this paper relates to all WARPS operated by the University of Wolverhampton.

### 3.1 West Midlands Councils WARP (WMCWARP)

Initial activity was focused to identify the resources and technical infrastructure essential in order to successfully operate a WARP. Membership of the WARP was offered as a six month free-trial. Following slow uptake, this was extended to twelve months. The WARP then became a subscription service. At the end of 2008 there were 11 subscribing members. At the time of writing, WMCWARP has 26 council members from a possible constituency of 33 Local Councils in the region.

### 3.2 East Midlands Government WARP (EMGWARP)

The East Midlands Government WARP was founded in 2006. This WARP was established as a partnership between the Local Government Association for the region, Leicester City Council and Mid-Yorkshire Chamber of Commerce (MYCCI). In 2007, MYCCI ceased involvement with WARP. Since this time, the University of Wolverhampton has been contracted to provide warnings and alerts for the EMGWARP. At the time of writing, EMGWARP has 29 council members from a possible constituency of 46 Local Councils in the region.

### 3.3 South East Government WARP (SEGWARP)

In 2009, the organisation responsible for supporting the activity of Local Government in the South East of England, South East Employers (SEE) decided to create a WARP. SEE contracted with the University of Wolverhampton as an experienced WARP Operator. SEGWARP achieved early rapid growth, recruiting over 20 subscribing members during the first six months. These remain the core members. At the time of writing, this WARP has 25 members.

### 3.4 National Health Service WARP (NHSWARP)

The NHSWARP commenced in 2008 as a pilot programme for NHS organisations, initially for the West Midlands region. Six members were involved in the pilot stage, half of whom became paying members. This WARP has not achieved the same level of maturity compared to the other WARPs operated by the University. The primary reason for this is the absence of available funding. However, developments have been closely monitored by information assurance leaders in the Department of Health. And the NHSWARP has attracted interest from the private health sector.

Consideration may therefore be given to changing the focus from a WARP that is for the NHS alone to an inclusive Health-Sector WARP. US-CERT has expressed an interest in this WARP.

## 4 WARP Resource Requirements

It is not a pre-requisite for a WARP to operate an automated system. However, the Author's experience suggests that it is both necessary and practical to create a professional WARP. The essential requirements of this system are; creation of user accounts and preferences, creation of alerts, filtering of alerts and issuing alerts. The system needs to be accessible by members from multiple locations. Hence a web based service is highly desirable.

Most WARPs use the Filtered Warning Application (FWA). This was originally developed at Microsoft, with licensing and intellectual property belonging to CPNI. The FWA has undergone a number of revisions and security assessment.

Different staffing models are operated by different WARPs in order to administer the system and issue alerts. The Wolverhampton model is primarily based on an academic member of staff supported by a Technical Assistant. Security is observed in recruitment process by ensuring that references are sought and this is supplemented by a Criminal Records Bureau (CRB) check.

## 5 Identifying Security Issues and Trends

This section summarises the main issues discussed within the WARP membership since 2007. The majority of the "Advice" and "Reporting" aspects of WARP are currently achieved via regular closed meetings. For UoW WARPs, these meetings occur quarterly. Each meeting operates a standard format with an agenda and minutes. The agenda includes a roundtable where members advise the group on any incidents that have occurred and how they have been addressed. It is also an opportunity for the participants to discuss their current work and provide feedback on the WARP itself. Each meeting includes a guest presentation. They will be from an expert speaker. The topic will be something of particular interest to the members, chosen by them. It will be an "agnostic" presentation: discussion of specific products or services is not allowed. The presenter is not allowed to attend for any other agenda items unless requested by the members.

There do not appear to be any difficulties in encouraging members to share information. The greater difficulty is achieving participation in meetings by the wider membership. Those who do attend a meeting will usually re-attend. However, there are members who do not attend meetings. Hence the maximum attendance at meetings is typically half of the membership.

The majority of problems reported by WARP members relate to the accidental loss of data. Typically this involves usb memory sticks, smart phones and laptops. Awareness in tackling this problem has increased. Most reports now state that devices were encrypted, whereas in the past this was not the case. Another trend (which has been encouraged through WARP) is the reporting of incidents to the Information Commissioners Office (ICO). Again, this is now more routine than exceptional.

The ICO have presented at a number of WARP forums with the intention of developing a relationship with WARPs in order to ensure that the most productive actions are taken in the event of a data loss.

Staff related issues are also common. These may range from reports of staff storing sound and video files on work based storage, through to the storage of pornographic material and harassment via email.

Compliance is another aspect of importance since the WARP began in 2007. The main requirement in Local Government has been the Code of Connection (CoCo). This requires the implementation of a detailed set of controls in order to connect to the Government Secure Extranet (GSx). It is necessary for Councils to do this in order to share information with Her Majesty's Revenue and Customs (HMRC). The current focus is moving towards the Public Sector Network (PSN). This aims to provide secure networks to a private cloud in which the public sector can operate. PSN has a separate code of connection.

2011 has experienced significant reductions in public sector funding. For some councils this has resulted in a loss of staff across all areas including IT. There is also an increase in the sharing of services. This for example, may involve one IT Department providing services for two or more councils. The other trend which has implications for security is the increased involvement of the private sector in public business. Contracting out IT Departments means for some councils that their whole IT function is provided by a private sector company. All of these issues have a potential impact for WARPs. Budget reduction could threaten the sustainability of the programme, sharing IT services may introduce a desire to share the WARP subscription and an outsourcing company may not wish to be a member of a local government WARP.

Many of the issues discussed by WARP members have related to the requirements of the CoCo. Hence some of the key requirements that members have needed to address include: the need for classification of data, securing remote access, penetration testing and log management. In more recent discussions, the use of Social Networks has been identified as an issue. The key conclusion from these discussions identified the necessity for each organisation to have a social network usage policy.

## 6 Other WARP Initiatives

Uow WARP has been involved in a number of trials and has provided alerts to other organisations via peer to peer links. These organisations have included the London WARP and the Law Society. UoW WARP has furthermore, engaged in activities to promote information assurance in the smaller business sector.

### 6.1 "Olympic WARP"

In 2008, the author engaged in the development of the WARP concept to support the 2012 Olympic Games. The intention was to strengthen information assurance for the games through the provision of a facility for security information sharing involving all parties rather than the main contractors alone. Whilst ddiscussions have taken place with several key stakeholders, it has not been possible to find a sponsor for a WARP initiative to support the 2012 Games.

### 6.2 International WARPs

The WARP programme is a UK initiative. However, it has attracted considerable interest from overseas. In 2007 employees of the electronics giant Hitachi visited the University of Wolverhampton. Following this, a decision was taken to create a WARP for the Hitachi Corporation's internal operations. A WARP has been created in the Irish Republic in lieu of a national CERT and in South Africa, the University of Johannesburg have created a WARP. More recently, a WARP has been registered for Flemish ICT Companies in Belgium [12].

In addition to these formally registered WARPs there has also been considerable interest from other countries. In Holland, there was a project to examine the use of WARPs with schools. More recently interest was expressed from a Chinese organisation. Overseas WARPs raise some interesting questions; how much should international WARPs be promoted / encouraged and what information (if any) should be shared with them? It is also necessary to consider how engaging with foreign organisations may affect the involvement of UK Government with WARP (e.g. CSIRTUK, GovCERTUK).

## 7 European Information Sharing Initiatives

An EU funded programme has been undertaken to address critical infrastructure protection through information sharing. The National & European Information Sharing & Alerting System (NEISAS) is an EU funded project created in 2009 to enhance critical infrastructure protection through trusted sharing of information [13]. Some initial findings of the project identify the need to provide a "true" exchange of information rather than simply a "push" web portal, enable the owner of the information to choose who can read it, support 'peer to peer' exchange between national platforms (with no central system) and enforce the Traffic Light Protocol (TLP) [14] for compliance for distribution.

NEISAS identifies key definitions and describes the community within which information sharing takes place as a "Trust Circle" which is facilitated by a "Trust Master". It provides a good example of how a member of a trust circle can share sensitive information without damaging reputation (i.e. the member discusses the issue with the trust master who then raises the issue without reference to that individual member). In developing a prototype application for cross-border information sharing, it has also defined some of the key requirements for such a system. One of the main challenges for this is overcoming the problem of losing control of distribution once an email has been sent. NEISAS suggests overcomes this by implementing Information Rights Management .

Another EU-funded project is the Framework for Information Sharing and Alerting (fisha).This aims to improve the security awareness amongst home users and smaller businesses by the creation of a European information sharing and alerting system [15]. The partners in this project are CERT Polska, CERT-Hungary and the University of Gelsenkirchen.

## 8 Conclusions

The techniques and concepts related to information security emerge from a highly secure world of secrecy. For example, cryptography was largely an application exclusive to the military and security services less than 20 years ago. The incredible growth of the internet and the rapid pace at which both internet applications and hacking techniques have developed, has made it necessary for these techniques and concepts to be applied increasingly in general use. Along with this has been the development of ways to share information across the public and private sector in response to increasing globalisation and the use of technology. This places a requirement for an environment where "secrecy" is counter-productive but where openness needs to be achieved in a "managed" way (because no one wishes to declare their vulnerabilities to a potential attacker or to be the subject of negative publicity due to the disclosure of an incident). The technical solutions to security issues have existed for some time and continue to be developed in order to meet changing requirements. However, they can only provide partial success because of a lack of effective information sharing and awareness.

The findings of this work suggest that information sharing systems for information security are still in their infancy. National CERTs have spread around the world (although there remain many countries where they remain absent e.g. many countries in Africa). They are largely, closed organisations often operated by the security agencies of their respective countries. In working largely independently, WARPs are able to achieve a more advanced level of information sharing with their communities. However, they do require development in order to fully achieve their goals. strategy. The NEISAS project has addressed some of the main issues for information sharing and show how they can be built into a software application.

UoW WARP sources the vast majority of information from vendors and independent review sites and evidence suggests that this is common for the information security community as a whole. For a national CERT to be effective in protecting their online citizens it is necessary for an ongoing dialog with organisations such as WARPs. Sharing information in order to improve security is still perhaps a difficult concept for some. It is evident that there is a great opportunity for a range of activity in this area and a requirement for greater openness in order for all to benefit from the knowledge and experience that exists.

## References

1. Spafford, E.H.: Purdue Technical Report CSD-TR-823. The Internet Worm Program: An Analysis. Illinois: Purdue University, Department of Computer Sciences (1999), `http://spaf.cerias.purdue.edu/tech-reps/823.pdf` (accessed: March 29, 2011)
2. U.S. Government: The 9/11 Commission Report, 416 p. WW Norton & Co. (August 2004), `http://govinfo.library.unt.edu/911/report/911Report.pdf` (accessed: March 29, 2011)
3. CPNI, Who we work with, Information Exchanges (2011), `http://www.cpni.gov.uk/about/who-we-work-with/information-exchanges/` (accessed: March 29, 2011)

4. Leicestershire County Council, Information Sharing Protocol (2009), http://www.leics.gov.uk/information_sharing_protocol.pdf (accessed March 29, 2011)
5. US-CERT, About Us (2010), http://www.us-cert.gov/aboutus.html (accessed March 29, 2011)
6. UNIRAS, Unified Incident Reporting and Alert Scheme (2006), http://web.archive.org/web/20010418174646/http://www.uniras.gov.uk/ (accessed January 01, 2011)
7. Cyber Security Strategy of the United Kingdom (Cm 7642). TSO, London (2009), http://www.official-documents.gov.uk/document/cm76/7642/7642.pdf (accessed February 11, 2011)
8. FIRST, First Members (2010), http://www.first.org/members/map/ (accessed February 11, 2011)
9. GOVCERT.NL, SETTING UP A CSIRT (2005), http://www.govcert.nl/render.html?it=86 (accessed January 08, 2011)
10. Proctor, T.: WARP speed ahead (2008), http://www.bcs.org/server.php?show=ConWebDoc.18595 (accessed March 30, 2011)
11. CPNI, Information sharing concept (2010), http://www.cpni.gov.uk/ProtectingYourAssets/informationSharing.aspx (accessed January 08, 2011)
12. WARP 2011, Warp Directory (2011), http://www.warp.gov.uk/directory.html (accessed March 30, 2011)
13. NEISAS 2011 (2011), https://www.neisas.eu/ (accessed March 28, 2011)
14. New Zealand Centre for Critical Infrastructure Protection (CCIP) 2011, Home, Incidents, TLP (2011), http://www.ccip.govt.nz/incidents/tlp.html (accessed March 28, 2011)
15. fisha 2011, The Project (2011), http://www.fisha-project.eu/the-project (accessed March 28, 2011)
16. CPNI, About CPNI (2011), http://www.cpni.gov.uk/about/cni/ (accessed July 22, 2011)

# The Contribution of NEISAS to EP3R

David Sutton[1], John Harrison[2], Sandro Bologna[3], and Vittorio Rosato[3]

[1] tacit.tel Limited, High Wycombe, UK
[2] Landitd Limited, Martlesham, UK
[3] ENEA - Energy and Environment Modeling Unit C.R. Casaccia - Via Anguillarese 301, Rome, Italy
david@tacit.tel, johnh@landitd.com,
{sandro.bologna,vittorio.rosato}@enea.it

**Abstract.** This paper summarises key attributes of the National and European Information Sharing and Alert System and its contribution to the European Public-Private Partnership for Resilience project, sponsored by the European Commission's Directorate-General Home Affairs as part of its European Programme for Critical Infrastructure Protection.

**Keywords:** NEISAS; EP3R, Information Sharing, Anonymisation, Traffic Light Protocol, Information Rights Management, Cross-Border Sharing.

## 1 Background to the NEISAS Project

In 2004, the European Commission (EC) published a Communication [1], proposing the framework for the European Programme for Critical Infrastructure Protection (EPCIP). This was developed to produce a more detailed proposal [2], setting out the objective, principles and framework for the programme.

In 2006/2007 on behalf of the EC, the European Network and Information Security Agency (ENISA) carried out a study in order to assess the "Feasibility for a Europe-wide Information Sharing and Alerting System" (EISAS) [3], targeted at Small to Medium Enterprises (SMEs) and citizens, which resulted in an EC Call for Proposals for a multi-lingual EISAS. The proposal submitted in response to the Call presented the project objective as:

> "The development of a Model and a Pilot Platform for a National and European Information Sharing and Alerting System, which would be based on the results of various European Union (EU) funded projects including the Information Assurance Messaging Standard (MS3i) [4], the ENISA Feasibility Study referred to above, the ENISA Data Collection Framework Study [5] and the Availability and Robustness of Electronic Communications Infrastructures Study (ARECI) [6]"

But, although the call was purely for a European model, the consortium submitting the proposal believed that effective trusted information sharing could not be created without providing a compatible National model, which should be

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 175–186, 2013.

applicable to any region, country or sector, and was henceforth referred to as NEISAS.[1]

The focuses of project were to be the definitions of:

- A National Information Sharing and Alerting Model and System
- A European Information Sharing and Alerting Model and System, conceived as a European Network of National Information Sharing and Alerting Systems

Also, the 2007 ARECI Study made a number of key recommendations:

- Recommendation 1, which outlines the requirement for a European Public-Private Partnership for Resilience (EP3R)
- Recommendation 4, which acknowledges the need for secure information sharing mechanisms, and recommends the setting up of a European Information Sharing and Alert System (EISAS)

Since one of the most important drivers for a successful Public-Private Partnership (PPP) is that of secure and trusted information sharing between public and private sectors, it is clear that the two are mutually inter-dependent.

## 2 The Challenge

Launched in 2009, the NEISAS project had three objectives:

1. To develop a NEISAS European Framework, which would also help all EU Member States to implement a National System or in case they already have one, to connect it to other Member States in a trusted way.
2. To develop an EISAS prototype (a software platform) based on end user requirements capture as well as the learning experience from other trusted information sharing models such as the existing Warning, Advice and Reporting Point (WARP) platform developed by United Kingdom (UK) Centre of the Protection of the National Infrastructure (CPNI). The new software platform would be implemented in three Member States: the UK, the Netherlands and Italy.
3. To develop a sustainable Business Model that would allow the prototype to scale up and through the creation of an Independent Body meet the requirements of more Member States.

## 3 Communities of Interest

The concept of the NEISAS platform is based around the existing use of information security communities of interest or *Trust Circles* in which there is an over-riding requirement for explicit trust between all *Members* of the community,

[1] See the NEISAS web site: `http://www.neisas.eu`

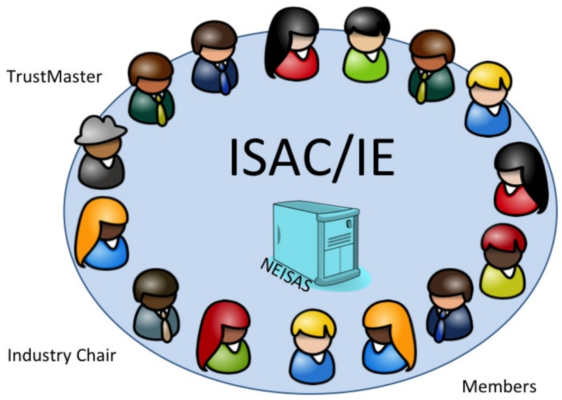


**Fig. 1.** A Single National Trust Circle

and in which a trusted individual acts as an impartial mediator or *TrustMaster*. Typically in those Trust Circles already in existence, the Chair of the Trust Circle is an industry representative, while a government representative organises and hosts meetings and takes the role of TrustMaster.

This role is crucial to the successful operation of the Trust Circle, as it is the TrustMaster who binds the circle together, invites new Members into the circle and acts as an intermediary between those Members who may not wish to identify themselves or their organisation in discussions on a particular topic, normally in order to avoid commercial sensitivities. Figure 1 illustrates a single national Trust Circle representing a particular critical infrastructure community.

Figure 2 illustrates multiple national Trust Circles, each logically independent of the others, but sharing the same physical NEISAS infrastructure. This permits the sharing of information between different critical infrastructure sectors within a Member State.

Figure 3 illustrates the situation in which multiple NEISAS systems are interconnected to permit the cross-border sharing of information, either between similar Trust Circles in different member States, or between different critical infrastructure sectors across Member States

Such communities of interest are often referred to as *Information Security Advisory Centres* (ISACs) or *Information Exchanges* (IEs). ISACs tend to specialise in sharing the analysis of information, whereas IEs focus on sharing the raw information itself. Both types have been set up with the express purpose of allowing Information Security professionals to exchange knowledge about security threats and vulnerabilities and to promulgate good practice in their own particular Critical Infrastructure sectors.

Information is normally shared at face-to-face meetings, but these may not occur at sufficiently frequent intervals, and in order to fill the gap between meetings (but not to replace them) some form of secure electronic information sharing is essential to the ongoing success of the Trust Circle.

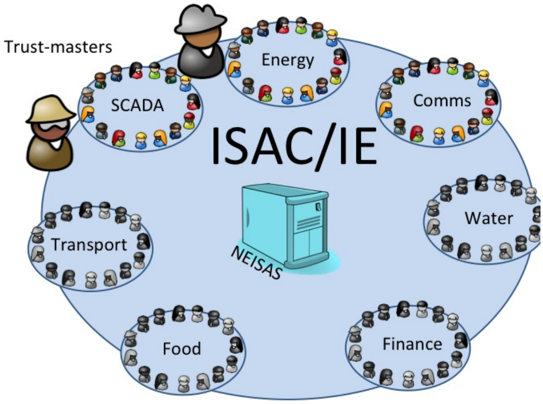


**Fig. 2.** Multiple National Trust Circles

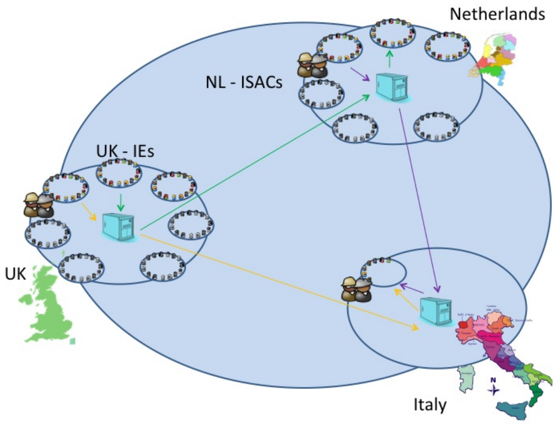


**Fig. 3.** Multiple International Trust Circles

In 2009, ENISA produced a Good Practice Guide [7] to setting up *Network Security Information Exchanges*, which lists the Information Exchange objectives as including:

- Learning more about intrusions into and vulnerabilities affecting the Public Network
- Developing recommendations for reducing network security vulnerabilities
- Assessing network risks affecting network assurance
- Acquiring threat and threat mitigation information
- Creating measures to prevent serious disruption or failure of public communications networks and services
- Taking measures to rectify any disruption or failure as soon as possible and with as little damage to critical interests as possible.

It also defines an Information Exchange:

> "An Information Exchange is a form of strategic partnership among key public and private stakeholders. In the Network Information Service (NIS) field, these can sometimes be referred to as *Network Security Information Exchanges* (NSIEs) although it is recognised that alternative names can also be used."

IEs and ISACs operate on the basis of person-to-person trust, and in most cases new Members can only be introduced with the unanimous agreement of the existing Members. Rules for membership may also include written undertakings that anonymise the source (individual or company) of the original information and that recipients of information shared must follow the *Traffic Light Protocol* (TLP).

## 4 Key Requirements

### 4.1 A Common Approach to Secure Information Sharing

Although standards exist governing the security of information *within* organisations,[2] there are currently none that define a common approach to securing the sharing of information *between* organisations. This is being addressed by the development of a new draft standard ISO/IEC 27010, which is due for publication towards the end of 2012.

### 4.2 The Traffic Light Protocol

The Traffic Light Protocol is a policy used to categorise information as *White* (unrestricted information), *Green* (community-wide, but not to be released outside the community), *Amber* (limited distribution on a need-to-know basis), and *Red* (personal, for named recipients only). The concept is already widely used within many different types of Public-Private Partnership.

### 4.3 Anonymity

Some topics that arise in an ISAC or IE meeting can be highly sensitive, and could potentially cause embarrassment to the originator's organisation from a commercial perspective. In cases such as this, the TrustMaster has a key role to play. The originator of the information may ask the TrustMaster to advise other Members of the circle about a particular topic, but to conceal their identity in order to avoid any possible commercial embarrassment. In doing this, the TrustMaster must anonymise not only the originator's identity, but also that of the information itself so that it cannot inadvertently be traced back to the originating individual or organisation.

[2] e.g. ISO/IEC 27001 and ISO/IEC 27002.

This potentially places a heavy burden on the TrustMaster, who would ideally ask the information originator to provide a precise form of words so that responsibility for confidentiality is retained by the originator.

In those cases where information may be passed between one Trust Circle and another (for example between the Energy and ICT sectors), the TrustMasters of both Trust Circles would act as intermediaries, protecting the identities of the membership of both Trust Circles.

### 4.4 Information Rights Management (IRM)

Although the Traffic Light Protocol provides a sound policy for the categorisation of sensitive information, it does not actually enforce it. Information Rights Management on the other hand permits the owner of a document to apply strong encryption to it, protecting the document from unauthorised opening. With appropriate IRM protection, the sharing of information is possible in the sure knowledge that documents cannot be digitally copied for onward distribution, and is much more secure than simple password protection.

### 4.5 Cross-Border Sharing

Whilst the ability to share information within a Trust Circle is implicit within the NEISAS system, it must also demonstrate the ability to share information not only between different Trust Circles (e.g. Information Communication Technology, Energy, Transportation) within a national NEISAS platform, but also across borders between Trust Circles on NEISAS platforms in different Member States.

### 4.6 Additional Requirements

From the UK, the CPNI raised three additional requirements, stating that NEISAS must add value to the information sharing process, for example, by productivity gains; inform and influence the development of new national systems and demonstrate the benefits of cross-border sharing in Europe.

The CPNI noted that they felt that it would be more likely that information exchange would take place between similar IEs in different Member States than it would between different IEs within them.

## 5 The Project

The project team consisted of consortium partners The Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA), Booz & Co., the Italian Government, the Dutch centre for cybercime (CPNI.nl) and Landitd (the software developer). The first phase of the project undertook an evaluation of information sharing systems and methodologies currently in use, followed by a period of collection of end-user requirements gained by interviews with stakeholders from the three Member States involved in the project.

From the 94 user requirements identified, 49 so-called *Use Cases* were developed, allowing the platform to be build around logical units. Development of the system began in April 2010, and a prototype wireframe model was available for basic functional testing by June, followed by a more useable and almost fully-functional build by September.

An intermediate build followed in November with minor bug fixes and enhancements and the final build was complete by early 2011, at which time a System Administrator's Guide and a User Guide were also produced. Testing of the software was carried out at ENEA's Usability & Media Laboratory (ULAB) in Rome, and consisted of the following stages – functionality testing, to prove that the system met the specified User Requirements; penetration testing, to verify the security aspects of the system, and usability testing, to verify the ease of use

Following all testing of the final build, trialling of the system began in Italy and the Netherlands with highly positive results, although for logistical reasons, cross-border trials were unable to be carried out.

### 5.1 Physical Implementation

Figure 4 illustrates the NEISAS architecture. Behind an external firewall there exist a total of five virtual servers for each NEISAS platform. For reasons of practicality, some of these may be located on the same physical hardware.

A front-end server running the Apache web server interface and the Drupal 6 environment that allows the publication, management and organisation of a wide variety of content on a website; an application server running Active Directory Rights Management Services (AD RMS), Active Directory Federation Services (AD FS) and the database.

Behind an internal firewall, the back-end servers include the Vasco IdentiKey server, used to authenticate NEISAS users (both Members and TrustMasters);

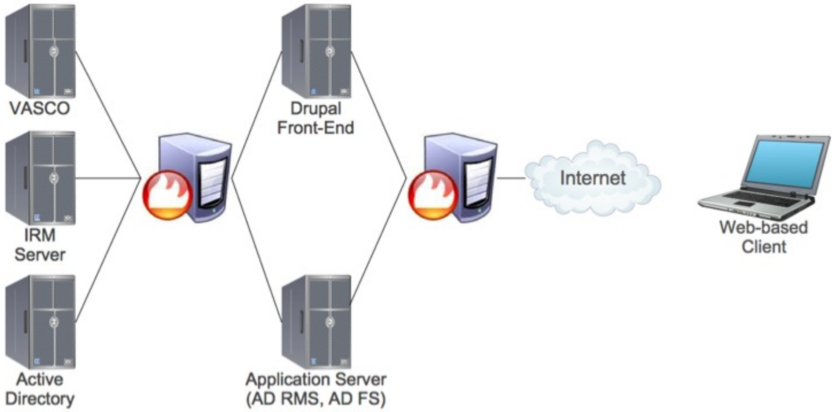


**Fig. 4.** The NEISAS Architecture

an Information Rights Management (IRM) encryption server, and an Active Directory (AD) domain controller

The client interface is specified to use Microsoft Windows XP/Vista/7. The browser is unspecified but the application has been tested with Internet Explorer, Safari and Firefox.

# 6 The Results Delivered

## 6.1 ISO/IEC 27010

NEISAS has already provided feedback into ISO/IEC 27010 development and will continue to do so, ensuring that potential stakeholders in PPPs are aware of the close links between NEISAS and the Standard, which should further encourage their engagement.

The result of this is that those PPPs who wish to implement their own Information Sharing platform can still exchange information securely with users of NEISAS-type systems, provided that their platform matches the Simple Object Access Protocol (SOAP) web services that the Controlled Information Distribution (CID) will make available to the web front-end, and that it is also (Draft) ISO/IEC 27010 compliant.

NEISAS has already provided feedback into ISO/IEC 27010 development:

> "Anonymisation is an important tool for creating effective information sharing communities. However, the control as presented here is inadequate. It is important that the sanitisation process looks at message content as well as the message source, because analysis of the content may reveal the identity of the source. It is also good practice to ask the source where possible to review the anonymised information and the list of intended recipients before it is distributed."

## 6.2 Anonymity

When posting information on the NEISAS platform individual Members may choose to self-anonymise information. For postings to other national Trust Circles or to Trust Circles on NEISAS systems in other Member States, the Trust-Master is able to set rules for the exchange of information which pre-define with which Trust Circles information may be exchanged, and whether the receiving Trust Circle Members are permitted visibility of either the Trust Circle or the originator of the posting.

## 6.3 Information Rights Management

NEISAS implements Information Rights Management using the Microsoft$^{TM}$ Active Directory Rights Management Service (AD RMS) in such a way that information is held and distributed securely whether within the NEISAS environment or outside it. Although the IRM-protected document may be sent outside the Trust Circle, the recipient may not open the document without a decryption key obtained by providing the correct credentials in the form of a valid NEISAS user name and password.

Additionally, IRM content may be sent by email to Trust Circle Members, who may access the content remotely by providing their NEISAS credentials when attempting to open the document.

## 7 Additional Benefits of NEISAS

In addition to meeting the key user requirements, NEISAS has delivered a number of additional benefits that should contribute to the success of EP3R:

### 7.1 Promotion of a Culture of Information Sharing

There is the need for a culture of information sharing to be promoted at the national government level within all Member States. Without the support of their central governments, the take-up of an information sharing mechanism of any kind is less likely to be successful, and could possibly provoke a negative reaction within the private sector which might see little advantage in sharing potentially confidential information without the presence of a trusted third party (i.e. the TrustMaster). For this reason, national Regulators must be kept fully aware of the Trust Circle's objectives, even if they are not themselves Members.

The recent pan-European exercise *Cyber Europe 2010* [8] generated a number of interim findings and recommendations, one of which was:

> "The exercise was only the first step towards building trust at pan-European level. More co-operation and information exchange is needed."

In all, 30 countries took part in Cyber Europe 2010, which demonstrates that at national government level at least, there appears to be considerable motivation towards working together to share information to resolve cyber issues.

The NEISAS project has demonstrated that secure information sharing is not only possible, but also can work successfully at a critical infrastructure community level, at a national level between diverse critical infrastructure communities and at an EU level between critical infrastructure communities. This should serve to assist national governments in promoting a culture of information sharing between themselves and their relevant private sector organisations.

### 7.2 Promotion of a Culture of Mutual Trust

There is the need to begin building trust in those critical infrastructure communities that would benefit from information sharing. This includes private sector trust in the national government (assuming that it is to be the government that will act as the TrustMaster) as well as building trust between private sector organisations within those communities, many of whom are in fierce competition with one another.

Trust is not built overnight, and it may take months or years to reach the point where private sector organisations feel comfortable with sharing sensitive information with each other and with their national governments. This is compounded by private sector organisations that operate across national boundaries, exhibiting differing degrees of trust with the respective governments.

In other cases, the parent companies of private sector organisations may be based outside the EU, and may have different views regarding trust and information sharing than the national operating companies they own.

The NEISAS project has demonstrated that information can be shared anonymously, so as to protect the identity of organisations that may feel that they could be at a commercial disadvantage if they share information more openly, and that it should help to build trust between fledgling communities.

### 7.3 Promotion of a Culture of Mutual Benefit

There are benefits both to the private and public sectors in the use of an information sharing mechanism:

- Private sector organisations will receive timely warnings of threats and vulnerabilities and examples of good practice from specialists in both the private and public sectors that will allow them better to protect their areas of the critical information infrastructure.
- Public sector organisations will benefit from an increased confidence that private sector organisations are actively pursuing due diligence and taking appropriate steps to maintain the integrity of their parts of the critical infrastructure as detailed in Article 13a of the Regulatory Reform Package [9], especially paragraph 2 which states:

  > "Member States shall ensure that undertakings providing public communications networks take all appropriate steps to guarantee the integrity of their networks, and thus ensure the continuity of supply of services provided over those networks."

Much of the work of the EP3R project will be to promote the cultures of information sharing, mutual trust and mutual benefit, and the NEISAS project has shown that it is possible to share information in a secure manner, to engender trust between both private and public sector organisations, and to anonymise information where required and to benefit from information shared.

## 8 The Business Model

Throughout the EU, the majority of the Critical Information Infrastructure is in private hands, but serves both the private and public communities. In order to ensure optimum protection of these, information on threats and vulnerabilities must be shared, and therefore exporting the Information Sharing Model to other Member States is of pivotal importance.

The project delivered a Business Model aimed at recommending an independent body to adopt NEISAS and to develop the platform for exploitation within existing information sharing communities within the EU, and with a view to helping those Member States that do not currently enjoy the benefits of such communities to develop them.

### 8.1 The NEISAS Business Model Recommendation

The NEISAS Business Model states that the prototype platform has been developed in such a way that it would be possible for willing public and private partner organisations to adopt the platform for their information sharing use or to utilise the potential commercially. Of the ten options evaluated by the project team, the most compelling choice was to adopt NEISAS within EP3R as a solution for trusted information sharing, initially for its working groups. EP3R could then facilitate Critical Information Infrastructure Protection (CIIP)-related Member State PPPs to take on the responsibility of hosting and maintaining a platform within their own countries. The proposal concluded:

> *"Of all the partner options, the EP3R Model ranks at the top because of its alignment with NEISAS and the positive impact it will have on EP3R functioning, because it has been designed to meet specific requirements of existing PPP which co-incidentally is the stakeholder community of EP3R. Moreover EP3R does not have a collaborative platform now and will benefit from the NEISAS platform."*

### 8.2 Other Possibilities for NEISAS Deployment

Although much of the work achieved on NEISAS to date is focused on its applicability to use within EP3R, two things are clear:

- That some Member States may wish to develop their own solutions to secure information sharing, as in the case of CPNI in the UK. This however does not necessarily preclude them from carrying out information sharing, as there is always the option to interconnect their systems with NEISAS systems, provided that their platform matches the SOAP web services that the CID will make available to the web front-end, and that they also comply with the forthcoming ISO/IEC 27010 Standard.
- That there are opportunities to exploit NEISAS beyond the scope of EP3R, and into communities that would wish to share information in a secure manner, but that are not necessarily connected in any way with the Commission's EPCIP programme, e.g. Police, Customs.

## 9 Conclusions

NEISAS has demonstrated clearly that it can meet the needs of ISACs and IEs both now and in the future. However, the EP3R project has now entered a phase in which both public and private organisations are beginning to intercommunicate on a regular basis and there is currently no formal mechanism for exchanging information between stakeholders. The NEISAS project team have strongly recommended that ENISA should make full use of NEISAS as a secure communications mechanism for EP3R, both as a project enabler, and as a means of publicising its benefits to the ISAC and IE communities within the EU.

## References

1. COM (2004) 702 Communication from the Commission to the Council and the European Parliament Critical Infrastructure Protection in the Fight Against Terrorism (European Commission) (2004), http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=COM:2004:0702:FIN:EN:PDF
2. COM (2006) 786 Communication from the Commission on a European Programme for Critical Infrastructure Protection (European Commission) (2006), http://eur-lex.europa.eu/LexUriServ/site/en/com/2006/com2006_0786en01.pdf
3. A Europe-wide Information Sharing and Alerting System (ENISA) (2007), http://www.enisa.europa.eu/act/cert/other-work/eisas_folder/EISAS_finalreport.pdf
4. Messaging Standard for Sharing Security Information (MS3i) (Symantec) (2009), https://www.neisas.eu/wp-content/uploads/2010/01/MS3i_main_Report_v3.0.pdf
5. Examining the Feasibility of a Data Collection Framework (ENISA) (2007), http://www.i-gov.org/images/articles/6079/data_collection_report_20080214.pdf
6. Availability and Robustness of Electronic Communications Infrastructures (ARECI) Final Report, March 2007 (Bell Labs and Professional Services) (2007), http://ec.europa.eu/information_society/policy/nis/docs/studies/areci_study/areci_report_fin.pdf
7. Resilient e-Communications Networks. Good Practice Guide Network Security Information Exchanges, ENISA, June 2009 (ENISA) (2009), http://www.epractice.eu/files/Resilience%20of%20public%20eCommunications%20Networks%20-%20Good%20Practice%20Guide-%20Network%20Security%20Information%20Exchanges%20(NSIE).pdf
8. 2010 Pan-European Cyber Exercise (2010), http://www.enisa.europa.eu/media/press-releases/cyber-europe-2010-a-successful-2019cyber-stress-test2019-for-europe(ENISA,2010)
9. Regulatory framework for electronic communications in the European Union (European Union) (2010), ISBN 978-92-79-14964-1, http://ec.europa.eu/information_society/policy/ecomm/doc/library/regframeforec_dec2009.pdf

# Policies to Improve Resilience against Major Industrial Accidents

Leire Labaka, Josune Hernantes, Ana Laugé, and Jose Mari Sarriegi

University of Navarra, TECNUN,
Paseo Manuel Lardizabal 13, 20018 San Sebastián, Spain
{llabaka,jhernantes,alauge,jmsarriegi}@tecnun.es

**Abstract.** A major industrial accident is an unpredictable event which triggers a disruption in a Critical Infrastructure (CI). This disruption can spread through other sectors, affecting not only the CI where the triggering event takes place but the whole society as well. In the case of major industrial accidents, system resilience consists of both the resilience of the CI (internal resilience) and resilience of society (external resilience). Resilience is the system's ability to reduce the probability of failure, the consequences from failure and the response and recovery time. However, little is known about how to achieve a high resilience level. In this paper, using the information gathered from experts and examining several major industrial accidents, we derive twelve policies that enhance the system's resilience level. The definitions of these policies are clarified through real case examples where the consequences of their use or lack of use are explained.

**Keywords:** Resilience, Critical Infrastructure, Crisis Management, Major Industrial Accidents, Resilience Policies.

## 1 Introduction

A major industrial accident can be defined as a crisis that starts in a Critical Infrastructure (CI) due to a disruption in the infrastructure or an element, such as an oil spill, a power outage, an aircraft crash or a nuclear accident. One of the main characteristic of current CIs is their interdependency. A crisis that starts in one sector may spread through the CIs' networks rapidly. For example, if a blackout occurs, the hospitals cannot carry out their current activities and the industries have to stop their production unless they dispose an autonomous power generation. Therefore, a crisis that starts in a particular CI spreads through the whole society affecting a great amount of people.

According to Rinaldi [1] there are four different types of CI interdependencies:

1. *Physical*: If the state of each CI depends upon the material output(s) of other CI.
2. *Cyber*: If the state of a CI depends on information transmitted through the ICT (Information and Communication Technologies) infrastructure.

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 187–199, 2013.

3. *Geographic*: If local environmental changes affect the CIs in that region, e.g., when the flooding of a reservoir knocks out a generator, this implies close spatial proximity.
4. *Logical*: If the state of each CI depends upon the state of another one via policy, legal, regulatory or some other type of governmental mechanism.

Thus, CIs cannot be considered as isolated entities but as a network of interconnected and interdependent elements. Bearing the importance of proper functioning of CIs for society's welfare in mind, we enhance the need for preparation and prevention measures.

Normally, the crisis is caused by an unpredictable event which can not have been foreseen. We cannot know when the triggering event will occur, which part of the system will be damaged and how it will spread through other sectors. Thus, this makes crisis prevention and preparation a challenging task.

This paper's main purpose is to break down the identified resilience types into resilience policies that crisis managers can implement in order to build up the system's resilience level. We do this through a study of major industrial accidents and also considering the information gathered from three workshops with experts.

The second section introduces the resilience concept and defines the two types of resilience that we have identified. Resilience policies that enhance the resilience level are presented in the third section and in the fourth one the influence of each policy on the crisis impact is defined. Finally, the main conclusions of the paper and the future work are proposed.

## 2 Resilience

Resilience is an essential concept when managing crises. It can be defined as the ability of the system to reduce the probability of failure, reduce the consequences from failure and reduce the time taken to cover all the response and recovery actions [2].

Some authors [2,3] break resilience down into four dimensions:

- *Technical resilience*: this refers to the ability of the organization's physical system to perform properly when subject to a crisis.
- *Organizational resilience*: this refers to the capacity of crisis managers to make decisions and take actions that lead to a crisis being avoided or at least to a reduction of its impact.
- *Economic resilience*: this refers to the ability of the entity to face the extra costs that arise from a crisis.
- *Social resilience*: this refers to the ability of society to lessen the impact of a crisis by helping first responders or acting as a volunteer.

Taking this definition and the various dimensions into account, we could say that a high level of resilience contributes to preventing the occurrence of a crisis and reducing its impact if one does occur.

The aim of crisis managers is to boost the system's resilience level to reduce the impacts from a crisis. However, how can we build up a resilient system? What actions should be implemented in order to improve the system's resilience level? Despite having a very clear general definition of resilience, there are many difficulties in breaking down this general perspective and putting it into practice.

System resilience is built up by implementing some preventive and preparatory measures such as improving the design of infrastructures and increasing maintenance levels or training operators to respond in the most effective and coordinated way. Resilience policies refer to the actions implemented in order to increase the system's resilience level. By applying these resilience policies, the system's resilience level will be enhanced, and consequently it will be able to reduce the potential impact. But, what are the policies that can be applied? How does each policy help to reduce the impact of a crisis? Due to resource scarcity, however, deciding how much should be invested in mitigation is a very challenging task. Furthermore, the influence of each policy varies depending on the triggering event.

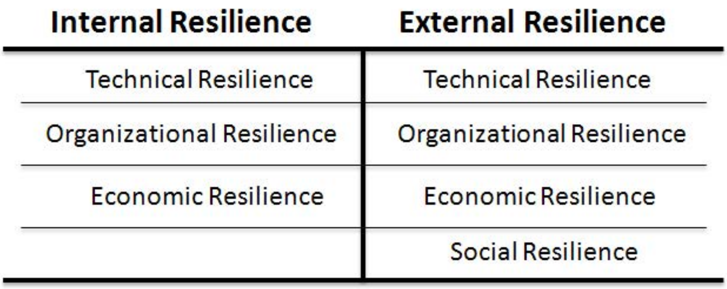
### 2.1 Resilience Dimensions in Case of Major Industrial Accidents

In the case of major industrial accidents, there is some focal asset where the triggering event occurs: a ship, a nuclear plant, a power grid plant, the chemical industry, etc. Additionally, as crises may become serious and affect a large number of people, the government needs to cooperate with the damaged industry or even lead the crisis resolution in the most appropriate way. Therefore, we divide the resilience level of an overall system into two different resiliencies: an internal resilience, which refers to the resilience level of the owner of the focal element/CI, and an external resilience, which corresponds to the resilience level of the rest of involved agents (the government, first responders, other CIs, and society).

Based on this classification, we identified some dimensions within each type of resilience. We divided internal resilience into three dimensions: technical resilience, organizational resilience, and economic resilience. External resilience, on the other hand, has been broken down into four dimensions: technical resilience, organizational resilience, economic resilience, and social resilience (see Fig. 1).

## 3 Resilience Policies

We organised three workshops in San Sebastian (Spain) to gather information with experts from different institutions such as energy companies, first responders, civil protection, health care, and organizations for CIs protection. Furthermore, we analyzed real cases from literature to get further information. Based on all this data we have identified different resilience policies that can be applied in order to improve the resilience dimensions. Afterwards, we have refined these policies to make clear the definition of each of them, which have been subsequently validated using some real examples.



| Internal Resilience | External Resilience |
|---|---|
| Technical Resilience | Technical Resilience |
| Organizational Resilience | Organizational Resilience |
| Economic Resilience | Economic Resilience |
|  | Social Resilience |

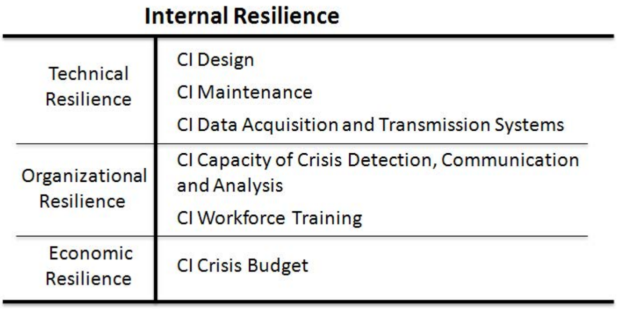
**Fig. 1.** Resilience types and dimensions in the case of a major industrial accident

The real cases analyzed illustrate the consequences of having a low or high level -or degree of effective implementation- of each policy. In some cases it can be seen that a low level of some policies led to the accident, whereas in others, a high level of them helped in its resolution.

It is important to highlight that it is much more complicated to obtain evidences about the efficiency of the policies when they have been correctly implemented and they have been successful avoiding or reducing the impacts of the crises.

### 3.1 Policies Applied to Internal Resilience

For technical resilience, three different resilience policies have been defined for each CI: CI Design, CI Maintenance, and CI Data Acquisition and Transmission Systems. To enhance organizational resilience, the following two policies can be implemented: CI Capacity for Crisis Detection, Communication and Analysis and CI Workforce Training. Finally, only one policy, which is called CI Crisis Budget, has been defined to build up the economic resilience of an organization (see Fig. 2 ).



| Internal Resilience |  |
|---|---|
| Technical Resilience | CI Design<br>CI Maintenance<br>CI Data Acquisition and Transmission Systems |
| Organizational Resilience | CI Capacity of Crisis Detection, Communication and Analysis<br>CI Workforce Training |
| Economic Resilience | CI Crisis Budget |

**Fig. 2.** Resilience policies within the internal resilience

**CI Design.** CI Design refers to the level of quality, robustness, redundancy and security of the design and construction of the infrastructure or element that the CI is responsible for. The infrastructure should meet all normative specifications and requirements. To know what specifications the element's design should meet, it is essential to precisely define its purpose, the risk level of the area against any potential threat, the aspects and characteristics of the surroundings, and how these surrounding aspects contribute to the security level of the infrastructure. Moreover, to increase the security level of the system, many infrastructures include additional security systems that should be designed in order to properly work in critical situations. Therefore, the design and construction of these security systems should be carried out consciously to make sure they are operational during the crisis. Finally, not only should the infrastructure design be reliable and robust but also care has to be taken not to introduce new vulnerabilities into the system when updates are introduced.

Two real cases that illustrate the potential catastrophic consequences due to inappropriate infrastructure design are the cases of Ford Motor Company and the DC-crash in Paris. In the 70s, the Ford Motor Company launched the Pinto model to compete with Japanese models. The narrow schedule for its design in addition to considerations of trunk space and manufacturing costs led engineers to place the gas tank between the differential and the rear bumper. In this position, a rear-end collision might push the gas tank forward into the differential, where the exposed bolts could rupture the tank, possibly leading to a fire or explosion. This serious design error cost Ford millions of dollars in legal settlements to accident victims in addition to untold damage to its reputation [4]. The other example was the DC-10 crash that occurred in Paris in 1974. In this case, a defectively designed rear cargo door blew open at an altitude of 12,000 feet, triggering cabin depressurization [5].

**CI Maintenance.** Not only should the CI be well designed but high quality maintenance activities also need to be performed periodically in order to improve the system's performance and reliability. These activities include repairing damaged parts, renewing old equipment with reliable components, updating technical features to comply with new legislation, etc. In performing these activities, we make sure that the system's elements are in an adequate and reliable condition and consequently the CI's technical resilience level will improve.

The critical nature of maintenance in preventing crises is clear as can be shown in the following example. In 1979, a DC-10 crashed in Chicago because of a maintenance problem. An improper maintenance procedure caused the left engine to break loose, severing control cables in the wing, and making it impossible for the pilots to control the airplane [5].

**CI Data Acquisition and Transmission Systems.** This policy has to do with the quality, reliability, and effectiveness of the sensors and computer equipment that should be set up in order to supervise and control the CI. Setting up the required sensors to gather information from the system and implementing adequate software to control the system are some of the main activities that should be carried out in order to achieve a high implementation of this policy. Through this equipment, it is possible to collect information from the system and transfer it to the central station to guarantee the proper functioning of the system. This way, if a failure does occur, the central station is immediately alerted in order to confront the situation.

The Canadian Blackout and the Spanair aircraft accident are two real cases in which the triggering event could not be avoided because the data acquisition and transmission systems did not work properly.

The Canadian Blackout that occurred in 2003 supports the fact that this policy helps preventing crises from occurring [6]. During a period of hot weather, many air conditioners were being used and the electricity demand increased considerably, leading to peaks in the electricity supply. The communication systems did not work as expected, and consequently grid managers did not receive the information about what was happening in real-time. As a result, managers were not aware of the critical state of the power grid and therefore, were not able to take action to prevent or mitigate the blackout.

In the same vein, in the case of the Spanair aircraft accident that occurred in Spain (2008) killed 154 passengers. The data from investigation showed that the takeoff manoeuvre took place with the flaps and slats retracted because the early warning system that should have detected the incorrectly positioned wing flaps failed to alert the crew to the problem [7].

**CI Capacity of Crisis Detection, Communication and Analysis.** CI Capacity of Crisis Detection, Communication and Analysis corresponds to the capacity of operators to detect, communicate, and analyze a crisis, proposing new preventive measures for the future. The activities carried out when this policy is implemented are training courses so operators are able to detect anomalous signals, communicate them to crisis managers, and then analyze them to establish new preventive measures. These operators are in charge of verifying the proper functioning of the whole system. Firstly, the operators should be able to detect and interpret the data provided, identifying the problem. Then, the incident will be communicated to crisis managers who will analyze its origin and consequences in order to identify the measures that must be taken to solve it and to prevent it from happening again.

The following two real cases manifest the importance of this policy to avoid the occurrence of a crisis. In 1977, the runway collision in Tenerife happened because of an occurrence of uncontrollable circumstance and an accumulation of human errors. The control tower and the crews of both planes were unable to see one another due to a sudden fog. Miscommunication between the tower and one of the airplanes caused the airplanes to collide [5].

In the case of the Italian power outage of 2003, the operators were unconscious of the urgency regarding the overload of the San Bernardino line. They were unaware of the fact that the overload on San Bernardino was only allowable for about fifteen minutes. Ten minutes after the trip ETRANS (Swiss network operator) called GRTN (Italian network operator) to decrease imports by 300MW. This measurement was completed by GRTN within 10 minutes. Despite the efforts, it was insufficient to relieve the overload and consequently San Bernardino line disrupted [8].

**CI Workforce Training.** Workers at the CI must be adequately trained prior to the occurrence of a crisis so they know how to respond when a crisis does occur. Workers should take training courses to know the procedures and protocols that should be followed when something unexpected occurs and to gain the skills they need to improve their response. In addition to this, they also have to train their sensemaking capacity in order to be able to understand the unexpected event, adapt to it, and make the correct decisions in a stressful situation and without much information. Responding on-time and working in a coordinated manner can significantly reduce the time needed to respond to a crisis, and consequently fewer negative effects will appear.

The Italian Blackout and the Chernobyl accident are two clear examples in which the negative consequences due to human errors can be illustrated. In 2003 in Italy, some electrical grid operators took inappropriate and ineffective measures, which added nine minutes to the time it took to solve the problem. This mismanagement was consequence of lack of training and it led to the disruption in the Sils-Soazza line and the Mettlen-Lavorgo line, disconnecting them from the grid [8,9].

Human error was one of the main causes of the Chernobyl accident. Technicians wanted run an experiment on the main reactor's main turbine in order to verify whether in the event of a power cut turbines would be able to supply enough power to the pumps before the standby diesel generators took over. When they began with the test, they suddenly realized that the reactor was working in unstable conditions but they ignored the situation and carried out the experiment until the reactor exploded [5,10].

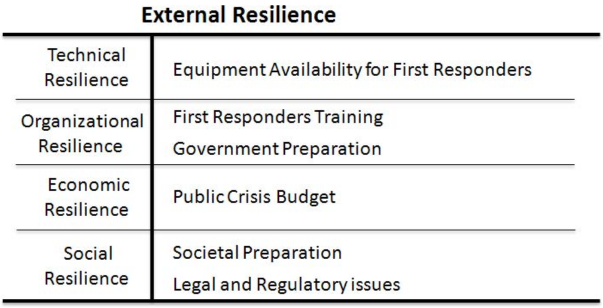
**CI Crisis Budget.** CIs should have resources set aside in order to cover repairs and replacements, should a crisis occur. This allows entities to increase their economic resilience level and consequently to buy new components, repair damage sooner, and temporarily hire workers and equipment, thereby reducing the response and recovery times. When this pool of money is reduced or even emptied, the response to the critical situation will take longer.

The recent BP Oil Spill is a good example that shows how the CI should possess some extra resources to be able to face the extra costs that arise from an accident.

The pool of money that BP has for emergencies seems to be enough to cover this severe incident. At the time of the Gulf oil spill, BP set up a $20 billion trust fund in order to satisfy the claims, which is plenty since until May 2011 they have only had to pay around $6 billion [11].

### 3.2 Policies Applied to External Resilience

Within the external resilience level we defined four dimensions. The policy that could help to improve technical resilience is having technical equipment available to first responders. First Responder Training and Government Preparation allow crisis managers to improve the organizational resilience. Having a large Public Crisis Budget for extra costs arising from a crisis allows all the expense of recovery and response activities to be covered. Finally, training society for crisis management and having well defined and updated regulations enhance the social resilience level (see Fig. 3).



**External Resilience**

| | |
|---|---|
| Technical Resilience | Equipment Availability for First Responders |
| Organizational Resilience | First Responders Training<br>Government Preparation |
| Economic Resilience | Public Crisis Budget |
| Social Resilience | Societal Preparation<br>Legal and Regulatory issues |

**Fig. 3.** Resilience policies within the external resilience

**Equipment Availability for First Responders.** The availability, quality, redundancy, reliability and security level of the technical equipment of the public bodies, first responders and society is essential in order to face a crisis, repair the damages, respond to emergency situations, introduce alternative emergency devices to replace the damaged ones, etc.

Purchasing the necessary equipment, maintaining them properly and updating them are some examples of the activities that should be carried out in this policy. Having high quality equipment allows first responders, government, and society to respond rapidly, reducing the impact of the crisis.

The following three examples expose how important is the availability of this equipment not to worse the critical situation and to increase the technical resilience level of the society.

During the gas leak in Bhopal, first responders realized there were serious problems because there were not effective emergency medical facilities or adequate transport for emergency evacuations [12]. The Exxon Valdez oil spill is another example in which there was a lack of equipment to deal with an oil spill of such magnitude and a long time was needed to get it [13].

Mendez-Martnez [14] claim that in the case of Prestige oil spill, the lack of adequate systems for prevention and response, led the Spanish government to accept several equipment offers from other nations which caused delays and a less efficient response.

**First Responder Training.** First Responder Training has to do with how first responders (fire fighters, emergency units, policemen, etc.) are prepared to face a crisis. Prior to the occurrence of a crisis, they should be trained to know how to respond to and solve a crisis and the procedures and protocols they must follow.

Actions such as how to act in dangerous places and how to organize themselves and coordinate with each other need to be defined before the critical event takes place. After a crisis, everything that went wrong must be identified, and measures should be enacted so they do not occur again.

First responders must be prepared and trained to act independently and effectively in dire circumstances. They must feel capable of operating with initiative and performing their tasks. They should be instilled with a set of core values, ethics, and priorities that will guide them in their decisions and actions. Potential responders should be trained to assess when emergency plans need to be activated.

The Bhopal accident and Exxon Valdez oil spill are two accidents where first responders lacked training and as a result the impact increased.

According to Bisarya and Puri, when the Bhopal accident occurred, the Mayor and the Chief Police of Bhopal recognized that they were not prepared to face such a crisis. They did not have the proper information about the storage of hazardous and dangerous materials in the plant or about their side effects. Furthermore, they found lack of coordination among company and emergency services [12]. In the case of Exxon Valdez oil spill, the first responders lacked the training to handle such major spills, and as a result the response time was longer than expected and consequently there was a large adverse impact [13].

**Government Preparation.** In a crisis, a government's main roles are to properly communicate the situation to the public and give advice about how they should behave, and to lead and coordinate all the entities that take part in dealing with and solving the crisis. Proper communication between the government and the public, where the government tells the public what they should do and how the resolution of the crisis is progressing, will diminish the public's anxiety, and as a result, the impact. When leading a crisis it is essential to increment their sensemaking capacity because crises are uncertain and complex. Therefore, crisis managers need to understand the critical situation and adapt to it rapidly [15]. Coordination among different entities is also essential to reduce the response and recovery time and the possible impact. All the entities taking part in managing the crisis should act in the most coordinated way in order to effectively reduce its impact.

The following three examples describe the importance of the government preparation in the effective crisis management. The government's inability to communicate and coordinate all the stakeholders related with crisis response and to get help from other nations will result in longer recovery times and greater impact, as was the case during the Exxon Valdez oil spill [13]. In the case of Prestige oil spill, although many experts said that the best alternative was to move the ship to the coast because of adverse weather, the stormy sea and the critical condition of the vessel, the Spanish authorities instead ordered to it be removed from the shore, and as a result the strength of the high seas crashed the vessel completely, spilling all the oil and increasing the resulting environmental damage [14].

**Public Crisis Budget.** As in the case of CI Crisis Budget, the public institutions should have a pool of money set aside in case a crisis occurs in order to help the stakeholders and society. This extra funding allows organizations, society and first responders to get resources in a reasonable way. If this pool of money is reduced because it is used, the government should fill it again although it might take some time to happen.

Two mining accidents explained below illustrate how having extra public money allocated for crises can lead to a satisfactory resolution.

The government's level of commitment may lead to totally different consequences for similar accidents. In the case of the San José mining accident that occurred in Chile in 2010, the high amount of resources invested by Chile's government allowed a rescue system to be built, which consequently saved the lives of all the miners. On the contrary, the Mexican government's attitude was different in the Pasta de Conchos mining accident. In this case, the government did not help in the rescue, and as a result 65 miners died [16].

**Societal Preparation.** Not only should the government and first responders prepare to respond to a crisis but society can also play an important role in crisis resolution. In the event of a crisis, elderly people may need assistance, hospitals can become overcrowded and so they need more personnel resources and some volunteers to repair damage.

Training the public would allow citizens to assist society during a crisis, thus reducing possible adverse effects. Society's awareness is very important factor in order to society prepare for the crisis. Having a good level of public preparation in the face of a crisis directly influences social resilience and in turn, reduces the impact.

In the case of the Prestige oil spill, the good practice of this policy enhanced in the response. Not only did volunteers help to clean the Galician cost, but they also brought about greater involvement from institutions and the government [17].

**Legal and Regulatory Issues.** Legal and Regulatory issues relate to the maturity level of the crisis regulations in order to take preventive measures and define protocols to know the responsibilities that each entity has when facing a crisis.

The regulations that private companies should meet, the regulations for the first responders and regulations for the public would allow everyone to be more prepared for the crisis and reduce possible impact. Indeed, not only should the regulations be defined, but it is also necessary to update them continuously. Having well defined and updated regulations would allow each agent to know what its responsibilities are in order to respond in the most coordinated and effective way.

The Chernobyl accident and the Italian Outage are two examples that highlight the need of a policy to improve crisis management.

The ability to deal with the Chernobyl accident was affected by the lack of proper regulation and the unstable political situation in the country. The regulations were mainly focused on the immediate response and lacked information about the post emergency period. Thus, the tragedy's consequences needed more time to be solved leading to a significant increase of the impact [18].

The Italian Outage [9] shows that having different regulations in Swiss and Italy lead to a longer resolution period. Thus, a unified legal and regulatory framework throughout Europe is necessary to ensure the security of grid operation and supply in Europe. Having different regulations in Swiss and Italy led to a longer resolution period.

## 4 The Influence of Policy Implementation Level on Crisis Impact

Not all the policies implementation level affects a crisis at the same point of its lifecycle. Even though all of them have some influence during the whole process, several policies are more successful at preventing a crisis whereas others mostly influence in the response and recovery period and also reduce impact (see Fig. 4).

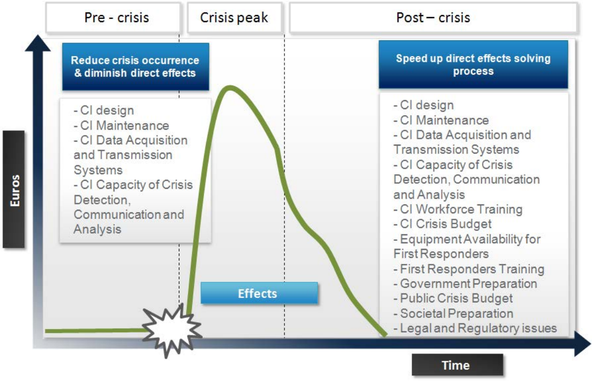

Pre - crisis
Crisis peak
Post – crisis
Reduce crisis occurrence & diminish direct effects
- CI design
- CI Maintenance
- CI Data Acquisition and Transmission Systems
- CI Capacity of Crisis Detection, Communication and Analysis
Speed up direct effects solving process
- CI design
- CI Maintenance
- CI Data Acquisition and Transmission Systems
- CI Capacity of Crisis Detection, Communication and Analysis
- CI Workforce Training
- CI Crisis Budget
- Equipment Availability for First Responders
- First Responders Training
- Government Preparation
- Public Crisis Budget
- Societal Preparation
- Legal and Regulatory issues
Effects
Euros
Time

**Fig. 4.** The influence of the policies throughout the whole lifecycle of a crisis

High level of CI Design, CI Maintenance, CI Data Acquisition and Transmission Systems, and CI Capacity of Crisis Detection, Communication and Analysis help to prevent the occurrence of a crisis. If our CI design is robust and secure enough and it is well maintained, it may be able to withstand some major hazards and prevent the triggering event from taking place. Moreover, if our data acquisition and transmission system is the appropriate one, we will be able to detect early warning signals and take measures to keep a crisis from occurring. Finally, the good level of detection, communication and analysis policy helps us to correctly interpret the signals we are receiving from the system and communicate threats to the managers so they can take the corresponding measures.

In the case of reducing impact, all the policies have an influence. Having a good level of all policies will allow all stakeholders to be more prepared to respond and recover from a crisis.

## 5 Conclusions and Future Work

Resilient Critical Infrastructures reduce the probability of incidents and crises occurring, and if they do occur, the impact will not be so significant. As a consequence, building resilience has become the most promising strategy in crisis management. This work-in progress research attempts to present and illustrate how this can be done with examples of twelve policies that contribute to this resilience building process. Bearing in mind these policies and their consequences will provide new insights to CI security managers.

However, this research is still incipient as we have not defined how each policy implementation level and system's resilience level can be quantified yet. Moreover, the influence of each policy into the overall system's resilience level needs to be also evaluated. As resources are scarce, in most cases it is impossible to implement all the policies. Therefore, knowing before the crisis which policy is the most efficient in diminishing impact would allow prioritizing them.

## References

1. Rinaldi, S.M.: Modeling and simulating critical infrastructures and their interdependencies. In: Proceedings of 37th Hawaii International Conference on System Sciences. IEEE Computer Society, Washington, DC (2004)
2. Bruneau, M., Chang, S., Eguchi, R., Lee, G., O'Rourke, T., Reinhorn, A., Shinozuka, M., Tierney, K., Wallace, W., von Winterfelt, D.: A framework to quantitatively assess and enhance seismic resilience of communities. Earthq. Spectra 19, 733–752 (2003)
3. Multidisciplinary Center for Earthquake Engineering Research (MCEER): Engineering Resilience Solutions (2008)
4. Fleddermann, C.B.: Engineering Ethics. Prentice Hall (2004)
5. Manion, M., Evan, W.M.: Technological catastrophes: their causes and prevention. Technology in Society 24, 207–224 (2002)
6. US-Canada Power System Outage Task Force: Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations (2004)
7. Comisión de Investigación de Accidentes e Incidentes de Aviación Civil: Accident involving aircraft McDonnell Douglas DC-9-82 (MD-82), registration EC-HFP, operated by Spanair, at Madrid-Barajas airport on 20 August 2008 (2008)
8. Union for the Coordination of Transmission of Electricity (UCTE): Final Report of the Investigation Committee on the 28 September 2003 Blackout in Italy (2004)
9. CRE and AEEG: Report on the Events of September 28th, 2003 culminating in the separation of the Italian Power System from the other UCTE Networks (2004)
10. Dörner, D.: The Logic of Failure. Addison-Wesley, Massachusetts (1997)
11. British Petroleum: Public Claims Status, `http://responsedata.bp.com/files/PublicClaimsStatusTracking05052011v2.pdf` (retrieved on 2011)
12. Bisarya, R.K., Puri, S.: The Bhopal gas tragedy - A perspective. J. Loss Prev. Process Ind. 18, 209–212 (2005)
13. Skinner, S.K., Relly, W.K.: The Exxon Valdez Oil Spìll (1989)
14. Méndez-Martínez, C.: Libro Blanco sobre el Prestige. Gobierno del Principado de Asturias, Oviedo (2003)

15. Boin, A., McConnell, A.: Preparing for Critical Infrastructure Breakdowns: The Limits of Crisis Management and the Need for Resilience. J. Conting. Crisis Manag. 15, 50–59 (2007)
16. Taniguchi, H.: El rescate de mineros en Chile revive las heridas del accidente en México, México (2010)
17. García-Mira, R., Real, J.E., Uzzell, D.L., San Juan, C., Pol, E.: Coping with a threat to quality of life: the case of the Prestige disaster. Revue Européenne de Psychologie Appliquée/European Review of Applied Psychology 56, 53–60 (2006)
18. Demin, V.F., Yatsalo, B.I.: Chernobyl Lessons Learned for Post-Emergency Response. IRPA 10 (2000)

# Fuzzy Input-Output Inoperability Model
## (Short Paper)

Gabriele Oliva[1], Roberto Setola[2], and Stefano Panzieri[1]

[1] University Roma Tre, Via della Vasca Navale 79, 00146 Roma, Italy
oliva@dia.uniroma3.it, panzieri@uniroma3.it
[2] University Campus Bio-Medico, Via A. del Portillo 21, 00128 Roma Italy
r.setola@unicampus.it

**Abstract.** In this paper the Input Output Inoperability Model (IIM) is extended in order to model perturbations by means of Fuzzy numbers, thus allowing to set up the model by means of vague and linguistic data.

**Keywords:** Critical-Infrastructures, Fuzzy Systems, Vague Data.

## 1 Introduction

Modeling Interdependent Critical Infrastructures is a challenging task; in fact in this field the models are hard to validate, because of the lack of quantitative data. This is one of the reasons that made the *Input-Output Inoperability Model* (IIM) [1] gain large attention in the research community, because of its economic origin. The model introduces the concept of *inoperability*, as the inability (in percentage) of each infrastructure to correctly operate. The inoperability of each infrastructure is assumed to linearly depend on the inoperability of the others and on external failures:

$$x(k+1) = Ax(k) + c \tag{1}$$

where $x$ represents the vector of the inoperabilities of the different infrastructures and $c$ is a constant vector representing induced failures. The elements $a_{ij}$ of matrix $A$, often called the *Leontief Coefficients*, are all non-negative and represent the degree of dependency of infrastructure $i$ on the $j$-th one. However, economy is just one dimension along which analyze interdependency phenomena. In [3] it is suggested to consider also information elicit from experts, thus introducing ambiguity. In [4] there is an attempt to introduce uncertainty in IIM using a probabilistic framework; however this approach seems not adequate to handle the ambiguity related to linguistic expressions. In [5] a fuzzy model has been presented, while in [7,11] it is shown how a fuzzy IIM model is able to codify uncertain data. This paper introduces a formal IIM model with fuzzy perturbations. The paper is organized as follows: Section 2 introduces the Discrete-time Fuzzy Systems and their stability; the proposed Fuzzy IIM model, as well as some simulative results, are presented in Section 3 and 4, respectively, while some conclusive remarks are collected in Section 5.

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 200–204, 2013.

## 2 Linear Discrete-Time Fuzzy Systems

In this section the stability of Linear *D*iscrete-Time Fuzzy Systems (DFS) is discussed, extending the scalar approach in [2,9,10,8]. A fuzzy subset of $\mathbb{R}$ is described by a membership function $\mu : \mathbb{R} \to (0,1]$ which assigns to each point $p \in \mathbb{R}$ a grade of membership in the fuzzy set. For each $\alpha \in [0,1]$, the $\alpha$-level set $[\mu]^\alpha$ of a fuzzy set is the subset of points $p \in \mathbb{R}$ with membership grade $\mu(p) \geq \alpha$. Let $\mathbb{E}$ be the space of all bounded compact and convex fuzzy subsets $\mu$ of $\mathbb{R}$ [2]; such sets are often called *Fuzzy Numbers* (FN). A *triangular* fuzzy number (TFN) $\mu \in \mathbb{E}$, in particular, is described by an ordered triple $\{\mu_l, \mu_c, \mu_r\} \in \mathbb{R}^3$ with $\mu_l \leq \mu_c \leq \mu_r$. Define the distance $d_{\mathbb{E}}(x,y)$ as the maximum difference in membership grades between the elements in $x$ and $y$ [2]. Define a linear and stationary *Discrete-Time Fuzzy System* (DFS) as follows:

$$x(k+1) = Fx(k);\ x(0) = x_0 \tag{2}$$

where $x, x_0 \in \mathbb{E}^N$ and $F$ is a $N \times N$ matrix. Extending the scalar case [2], it is possible to prove that a linear and stationary DFS (2) is stable if there exists:

- a stable crisp system $z(k+1) = Gz(k)$, $z(0) = z_0$, where $G \geq 0$ and $z, z_0 \in \mathbb{R}^N$;
- a defuzzyfication function $V(x) \geq 0$ such that for $z(0) = V(x(0))$ it follows that $V(x(k+1)) \leq z(k+1)\ \forall k$;
- a continuous, positive and monotone non-decreasing functional $a(\cdot)$ such that $a(d_{\mathbb{E}^N}[x(k), 0]) \leq ||V(x(k))||$, where $d_{\mathbb{E}^N}$ is the distance in $\mathbb{E}^N$ defined as the sum of the distances for each component [2].

Notice that the stability of a fuzzy system can be derived from the stability of a non-fuzzy system. It is possible to prove the following result, for which only a sketch of proof is given.

**Theorem 1.** *Let a linear and stationary DFS, such that $F \geq 0$. Then, the DFS is stable if the crisp system $z(k+1) = Fz(k)$ is stable.*

*Proof.* It is sufficient to show that there exists $V(x)$, $a(\cdot)$ that satisfy the above conditions; this is true if $V$ is the *zero-membership function* (i.e., the degree of membership of 0 in $x$) and if $a(\cdot) = d_{\mathbb{E}}(\cdot,\cdot)$.

In the linear case, the evolution of a DFS can be easily evaluated level-wise. Let $F^+$ and $F^-$ be the positive and negative parts of $F$, respectively. For each $\alpha$-level, the system can be represented in the following form [6]:

$$\begin{bmatrix} \underline{x}^\alpha(k+1) \\ \bar{x}^\alpha(k+1) \end{bmatrix} = \begin{bmatrix} F^+ & F^- \\ F^- & F^+ \end{bmatrix} \begin{bmatrix} \underline{x}^\alpha(k) \\ \bar{x}^\alpha(k) \end{bmatrix} \tag{3}$$

Under the non-negativity assumption made for $F$ it follows that $F^+ = F$ and $F^- = 0$.

## 3 Fuzzy IIM Model

Define a discrete-time IIM fuzzy system (IIMF) as follows:

$$x(k+1) = Ax(k) + c,\, x_0, c \in \mathbb{E}^N,\, a_{ij} \geq 0 \; \forall i, j = 0, \ldots, N \tag{4}$$

where $A \in \mathbb{R}^{N \times N}$. In such a model the inoperability of each infrastructure $x_i$ is described by means of a fuzzy variable, i.e., by a set of values with different degree of believeness, instead of a "crisp value". If each state variable and input is described by a triangular fuzzy number, the center of the triangle represent the trajectory associated with the maximum belief, while the left and right endpoints represent the best/worst cases.

**Corollary 1.** *Let a discrete-time IIM fuzzy system (4), and suppose that A is stable; then the system is stable.*

*Proof.* Since the perturbation $c$ is constant, it is possible to rewrite the model (4), including the $c$ inside the state of the system; the result is

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{c}(k+1) \end{bmatrix} = \begin{bmatrix} A & I \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{c}(k) \end{bmatrix} \tag{5}$$

Due to the non-negativity of the Leontief Coefficients, it follows that the conditions required by Corollary 1 are verified. It remains to prove that the reshaped dynamic matrix is stable in the usual crisp sense. The dynamic matrix of Eq. (5) is block triangular, and is stable if $A$ is stable.The proof is complete. □

## 4 Simulation Results

We consider the scenario developed in [7], which analyzes the 11 Italian Critical Infrastructures (see Figure 1 for details) ; the considered Leontief Coefficients are intended for a outage scenario lasting from 1 to 6 hours, and are reported in Table 1; such a system satisfies the hypotheses of Corollary 1, since the coefficient are non-negative and the sum of the elements on each row is less than one. In our simulation, starting from a full operability condition a "severe impact" is induced on the electrical grid (i.e., $c_2 = [0.5, 0.6, 0.7]$) and in the same time a "moderate impact" on the wired telecommunication network, i.e. $c_3 = [0.2, 0.3, 0.35]$ (we used the fuzzy codification procedure detailed in [7]). Figure 1 shows the evolution of the crisp system bounded by the left and right extrema of the support of the fuzzy system. Notice that the distance between the best and worst curves that characterizes each infrastructures. This data provide us an estimation of the "uncertainties" that characterize the estimation, that as evident from the Figure 2, is not uniform on all the infrastructure. However, the most interesting aspect to observe is the relative position of the curve associated with the crisp system with respect to those associated with the best and worst case. As evident, even if the curve associated with the crisp system is always enclosed between the other curves, it does not represent the mean value. The same considerations can be deducted by the Figure 1(right) which shows the triangular shape of the fuzzy system state at steady state that, as evident, is not isosceles.

**Table 1.** Leontief Coefficients for the case study (source [7])

| | # 1 | # 2 | # 3 | # 4 | # 5 | # 6 | # 7 | # 8 | # 9 | # 10 | # 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **# 1** | 0.000 | 0.134 | 0.460 | 0.308 | 0.033 | 0.024 | 0.011 | 0.001 | 0.02 | 0.007 | 0.310 |
| **# 2** | 0.000 | 0.000 | 0.023 | 0.010 | 0.002 | 0.003 | 0.000 | 0.008 | 0.002 | 0.179 | 0.004 |
| **# 3** | 0.006 | 0.083 | 0.000 | 0.013 | 0.004 | 0.003 | 0.005 | 0.002 | 0.001 | 0.004 | 0.005 |
| **# 4** | 0.002 | 0.109 | 0.120 | 0.000 | 0.002 | 0.005 | 0.002 | 0.002 | 0.002 | 0.004 | 0.007 |
| **# 5** | 0.005 | 0.050 | 0.020 | 0.009 | 0.000 | 0.005 | 0.008 | 0.005 | 0.008 | 0.008 | 0.020 |
| **# 6** | 0.001 | 0.233 | 0.109 | 0.104 | 0.005 | 0.000 | 0.007 | 0.002 | 0.006 | 0.001 | 0.005 |
| **# 7** | 0.003 | 0.100 | 0.100 | 0.030 | 0.007 | 0.003 | 0.000 | 0.003 | 0.003 | 0.007 | 0.008 |
| **# 8** | 0.006 | 0.036 | 0.039 | 0.039 | 0.026 | 0.024 | 0.017 | 0.000 | 0.017 | 0.018 | 0.029 |
| **# 9** | 0.008 | 0.500 | 0.050 | 0.100 | 0.050 | 0.020 | 0.020 | 0.020 | 0.000 | 0.000 | 0.008 |
| **# 10** | 0.002 | 0.030 | 0.005 | 0.009 | 0.005 | 0.000 | 0.002 | 0.005 | 0.005 | 0.000 | 0.005 |
| **# 11** | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |

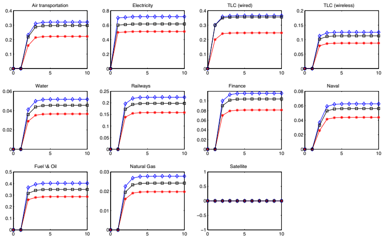


**Fig. 1.** Evolution of the different inoperabilities for the crisp system (black boxes) and evolution of left (in red asterisks) and right (in blue diamonds) extrema of the fuzzy system for $\alpha = 0$ (i.e., the support)

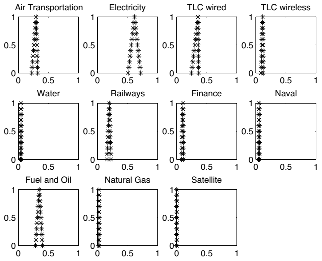


**Fig. 2.** Equilibrium state reached by the IIM fuzzy variables

## 5 Conclusions

In this paper the IIM model has been extended, allowing to manage vague and ambiguous quantities by introducing a Fuzzy version and studying its stability. Further work will be devoted to introduce ambiguity on the parameters (i.e., the leontief coefficients), as well as considering models with some non-linearities (e.g., hysteresis, saturations, logical conditions).

## References

1. Haimes, Y., Jiang, P.: Leontief-based Model of Risk in Complex Interconnected Infrastructures. Journal of Infrastructure Systems, 1–12 (2001)
2. Lakshmikantham, V., Mohapatra, R.N.: Theory of fuzzy differential equations and inclusions. CRC Press LLC (2003)
3. Lewis, T.G.: Critical Infrastructure Protection in Homeland Security: Defending a Networked Nation. Wiley (2006)
4. Orsi, M.J., Santos, J.R.: Incorporating Time-Varying Perturbations Into the Dynamic Inoperability InputOutput Model. IEEE Tran. on Systems, Man and Cybernetics - Part A: Systems and Humans 40(1), 100–106 (2010)
5. Panzieri, S., Setola, R.: Failures propagation in critical interdependent infrastructures. Int. J. Modelling, Identification and Control 3(1) (2008)

6. Pearson, D.: A property of Linear Fuzzy Differential Equations. Appl. Math. Lett. 10(3), 99–103 (1997)
7. Setola, R., De Porcellinis, S., Sforna, M.: Critical infrastructure dependency assessment using the input-output inoperability model. Int. Journ. of Critical Infrastructure Protection 2, 170–178 (2009)
8. Oliva, G., Setola, R., Panzieri, S.: Distributed Consensus under Ambiguous Information. International Journal of Systems of Systems Engineering 4(1), 55–78 (2013), doi:10.1504/IJSSE.2013.053504
9. Oliva, G., Panzieri, S., Setola, R.: Distributed Synchronization Under Uncertainty: A Fuzzy Approach. Fuzzy Sets and Systems 206, 103–120 (2012), doi:10.1016/j.fss.2012.02.003
10. Oliva, G.: Stability and Level-wise Representation of Discrete-time Fuzzy Systems. International Journal of Fuzzy Systems 14(2), 185–192 (2012)
11. Oliva, G., Panzieri, S., Setola, R.: Fuzzy Dynamic Input-Output Inoperability Model. International Journal on Critical Infrastructure Protection 4(3-4), 165–175 (2011), doi:10.1016/j.ijcip.2011.09.003

# Dependencies Discovery and Analysis in Distributed Systems
## (Short Paper)

Emiliano Casalicchio

Department of Computer Science, University of Roma "Tor Vergata", Roma, Italy
emiliano.casalicchio@uniroma2.it

## 1 Introduction

The welfare of our daily life depends, even more, on the correct functioning of complex distributed applications. Moreover, new paradigms such as Service oriented computing and Cloud computing encourage the design of application realized coupling services running on different nodes of the same data center or distributed in a geographic fashion. Dependencies discovery and analysis (DDA) is core for the identification of critical and strategical assets an application depends on, and it is valid support to risk and impact analysis [10].

The goal of this research, framed in the context of the MOTIA [1] project, is to define methodologies and metrics to quantitatively and qualitatively evaluate service level dependencies in critical distributed systems.

In literature there is a pletora of network monitoring tools, working at layer 2 and 3, that offer discovery dependencies features and that allow to building a dependency map of the observed system (an updated list could be found here [9]). On the contrary few works concentrate their attention on application level DDA [5, 4, 7, 6, 1–3]. Often, DDA is used as a tool for distributed application management and typically gives a qualitative picture of system dependencies. At the best of our knowledge there are no examples of works oriented to application level dependency quantification that is, no indicators has been defined to quantify how much two services are dependent.

This paper briefly describe DeDALO, the DEpendency Discovery and AnaLisys using Online traffic measurement framework we have designed and implemented.

## 2 The DeDALO Framework

The architecture of the DeDALO framework is sketched in figure 1. DDA is realized through four main steps (implemented by related software modules):

[1] This research is partially supported by the MOTIA project, http://www.motia.eu, funded by the European Commission - Directorate-General Justice, Freedom and Security: JLS/2009/CIPS/AG/C1-016.

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 205–208, 2013.

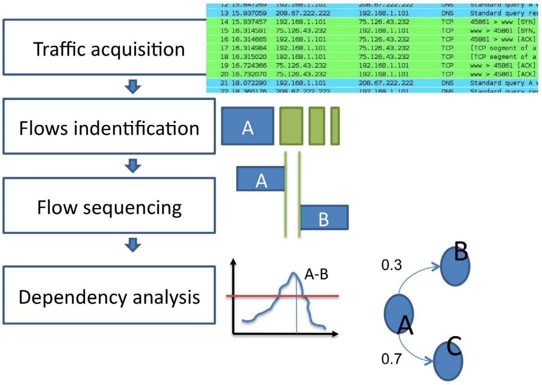
Traffic acquisition, Flow identification, Flow sequencing and Dependency analysis. In the following we give a brief description of each phase and therefore we concentrate our attention on Flow Identification and Dependency Analysis.

*Traffic acquisition.* In this phase DeDALO accesses a network interface card or a PCAP file to extract an IP packet and to convert it in a manageable data structure. The DeDALO observation system, inspired to existing works, has been implemented using an agent-based architecture, deploying several agents to observe and acquire the system activities. To implement the network traffic collection module we use the `libpcap` library.

*Flow identification.* As second step, DeDALO works on the extracted packet trying to match it with a suitable group of packets (flow) according to its header. Each flow is an instance of an access, performed by a client node to obtain a service from a server node.

*Flow sequencing.* For any starting flow, DeDALO match it with all ended flows, originated by the same client node, saving the corresponding interarrival time. For any couple of flows, DeDALO keeps a frequency distribution which counts the occurrencies of a given time.

*Dependancy analysis.* In this phase DeDALO analizes all distributions to evaluate a possible dependency between the observed flows, applying an inference engine to all samples. The result of this analysis is a qualitative evaluation of dependencies (let say Yes/No/Maybe). The analysis phase is completed by the evaluation of a dependency metric we propose and that measure the intensity of a dependency. As result we DeDALO builds a wheighted graph with the flows represented as vertex and the dependencies as weighted edges.



```
12 15.647269  192.168.1.101   208.67.222.222  DNS   Standard query A
13 15.937059  208.67.222.222  192.168.1.101   DNS   Standard query res
14 15.937457  192.168.1.101   75.126.43.232   TCP   45861 > www [SYN]
15 16.314591  75.126.43.232   192.168.1.101   TCP   www > 45861 [SYN,
16 16.314665  192.168.1.101   75.126.43.232   TCP   45861 > www [ACK]
17 16.314984  192.168.1.101   75.126.43.232   TCP   [TCP segment of a
18 16.315020  192.168.1.101   75.126.43.232   TCP   [TCP segment of a
19 16.724366  75.126.43.232   192.168.1.101   TCP   www > 45861 [ACK]
20 16.732070  75.126.43.232   192.168.1.101   TCP   www > 45861 [ACK]
21 18.072290  192.168.1.101   208.67.222.222  DNS   Standard query A
22 18.360176  208.67.222.222  192.168.1.101   DNS   Standard query res
```

**Fig. 1.** Phases of the DeDALO framework

### 2.1 Flow Identification and Sequencing

We define *data flow* $F_i$ a continuos sequence of information exchanged by two nodes. A service $S_i$ consists of specific software components, running on a networked node, and providing its functionalities though the network.

Each data flow considers a tuple composed by the two end-point nodes and the time interval in which they exchanged data. The end-points are called *origin* and *destination*, identifying the source of the request and the element offering the service. Two data flows, let say $F_{i,j}$ and $F_{i,k}$, originated by the same node $i$, towards two different destinations ($j$ and $k$), may occur in a sequence interleaved by an interarrival time $\lambda_{i,j,k}$. It is possibile to study the correlation between the two data flows by analyzing the distribution of $\lambda_{i,j,k}$.

The assumption made by DeDALO is that the relation of dependency between two services $S_j$ and $S_k$ is a condition related to a systematic and sequential use of both services inside a common application context (of course, this assumption impose some limitation we will discuss later on). If a sequence of data flow (representing service accesses) is repeatly spaced by similar interarrival time, than we assume that the two flows (and by consequence, the two services) are bound by a dependency relation. For example, if data exchanged through the first flow $F_{i,j}$, toward $S_j$, are vital to the origin end-point $i$ to successfully access $S_k$, access identified by $F_{i,k}$. Therefore $S_j$ and $S_k$ are dependent, as well as the application/service $S_i$, $S_j$ and $S_k$.

A common example is the domain name resolution process (very simplified here). The sequence of a DNS flow and a HTTP flow analyzed from the vantage point of a node running a web browser is likely to be found many times, because the node will need to know the HTTP server IP address, querying the DNS. If the DNS is unavailable the HTTP server will be not reachable and, even if it is properly working, the client will perceive a denial of service.

DeDALO's goal is to retrieve couples of IP services bound by a dependency relation. It carries on the analysis by fetching every IP packet exchanged between an origin and two or more destination services. The algorithm considers only packets headers and timings, discarding the payload.

DeDALO identifies a data flow by aggregating packets with same connection information (IP address, TCP ports, protocol) saving the timing of first and last packet of each flow, in order to identify the interarrival time. It's important to point out that all timing information are coherent, as they are referred to packets originated by the same node, and there's no need for syncronization.

As a new data flow $F_{i,j}$ is identified, DeDALO tries to correlate it with all flows $F_{i,*}$ previously originated by the same node. Each of these flows is coupled with $F_{i,j}$, by calculating the interarrival time between it and $F_{i,j}$, meaning that a sequence between two data flows has been detected with a given interarrival time. When two flows are correlated, we're also correlating the nodes which are responsibles for the offered services. For each couple of flows (and nodes) is computed the interarrival time frequency distribution, counting the occurencies of timing samples. The distribution represents the frequency with which one or more client nodes access two services subsequently. Any distribution can have different shapes and can be characterized by a specific mean, standard deviation and sparseness.

### 2.2 Dependency Discovery Model

DeDALO uses all interarrival time frequency distributions to identify a dependency between the corresponding services. It looks for one or more interarrival times with a frequency higher than a given *dependancy threshold d*.

To build our discovery model we considering that only data flows $F_{i,j}$ and $F_{i,k}$ with an interarrival time $\lambda_{i,j,k} \leq 3$ seconds are considered to contribute to possible dependencies. The distribution of interarrival time is computed binning observed interarrival times with a granularity of $10^{-2}$ seconds. Therefore, being $\mu_f$ the average value for interarrival time frequency and $\sigma_f$ its standard deviation, we compute the dependency threshold as $d = \mu_f + \alpha \cdot \sigma_f$, where $1.96 \leq \alpha \leq 3$, for example in [3] the authors use $\alpha = 3$.

If one or more interarrival time show a frequency higher than the dependency threshold, than such interarrival time value are evidence of a systematic behavior that is likely to represent a dependency.

## 3 Concluding Remarks

The main lessons learned from the design and implementation of DeDALO are the followings. First, there is a set of limitation to draw a map of service level dependencies in a network regardless its extension.Such limitation can be classified as structural factors (e.g. switched networks) and encoding factors (e.g. packet encryption). Second, a dependency model is strictly related to the observation points used, and therefore the it is influenced by the policy used to deploy agents.

## References

1. AggreGate. Network Manager, http://aggregate.tibbo.com/
2. Bahl, P., Chandra, R., Greenberg, A., Kandula, S., Maltz, D.A., Zhang, M.: Towards highly reliable enterprise network services via inference of multi-level dependencies. Microsoft Research (2007)
3. Chen, X., Zhang, M., Morley Mao, Z., Bahl, P.: Automating network application dependency discovery: experiences, limitations, and new solutions. Microsoft Research, University of Michigan (2008)
4. HP. Network management center, https://h10078.www1.hp.com/cda/
5. IBM. Tivoli, http://www-01.ibm.com/software/tivoli/
6. OpenNMS, http://www.opennms.org/
7. ServiceNow, http://www.service-now.com/
8. Kandula, S., Chandra, R., Katabi, D.: Whats going on? Learning communication rules in edge networks. Microsoft Research (2008)
9. Standford Linear Accelerator Center, Network Monitoring Tools, http://www.slac.stanford.edu/xorg/nmtf/nmtf-tools.html
10. Tyson Macaulay Critical Infrastructure: Understanding Its Component Parts, Vulnerabilities, Operating Risks, and Interdependencies. CRC Press Inc. (December 2009)

# Protecting Critical Infrastructures from Stealth Attacks: A Closed-Loop Approach Involving Detection and Remediation
## (Short Paper)

Stefano Avallone, Claudio Mazzariello, Francesco Oliviero,
and Simon Pietro Romano

University of Napoli "Federico II"
Via Claudio 21, 80125 Napoli, Italy
{stavallo,cmazzari,folivier,spromano}@unina.it

**Abstract.** We present an architecture capable to protect Critical Infrastructures from one of the most harmful categories of Denial of Service (DoS) attacks, commonly known in the literature as either "low-rate", or "stealth" attacks. Stealth attacks do represent the last generation of network threats, since they minimize both cost and visibility, at the same time achieving an effectiveness which is comparable to that of common brute force attacks. The study is conducted by exploiting an actual deployment of an architecture for the effective protection of Critical infrastructures, designed and developed within the INSPIRE European Project.

**Key words:** Critical Infrastructure Protection, Intrusion Detection, Remediation, Denial of Service (DoS).

## 1 Introduction

Stealth attacks tend to smartly exploit the intrinsic features — and potential vulnerabilities — of network communication protocols in order to purposely put into crisis the interacting end-points, while evading common rate-based DoS detection mechanisms. A typical example of a stealth attack is represented by a low rate malicious flow exploiting the TCP retransmission property for denying service to legitimate users. In this paper we focus on the "shrew attack" [1], which relies on short traffic bursts having both a maliciously chosen duration and a maliciously chosen low rate of occurrence. The study is conducted by exploiting the deployment of an actual architecture for the effective management and protection of a SCADA-based Critical Infrastructure, which has been designed and developed as an outcome of the INSPIRE Project[1].

The typical architecture of current CIs has a hierarchical structure, which integrates heterogeneous devices and network trunks, also via shared network connections. To achieve interoperability, open communication protocols are increasingly being used, thus exposing SCADA (Supervisory Control And Data Acquisition) systems to the same vulnerabilities which threaten general purpose Information Technology (IT) systems.

[1] http://www.inspire-strep.eu/

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 209–212, 2013.

## 2 An Intrusion Detection and Reaction System for Critical Infrastructure

In the framework of the INSPIRE project, we have implemented some solutions to guarantee both security and resilience of Critical Infrastructures. Firstly, we have devised a routing mechanism [2] that allows the communication infrastructure of a SCADA system to be resilient to both node/link failures and attacks. A first advantage of this technique is that an attacker that has been able to compromise a node will not intercept all the packets and thus will not be able to reconstruct the whole information flow. Another advantage of this technique is disclosed in the case where a node is found to be under attack. The splitting technique allows for a fast re-routing of the flows traversing the attacked node, preserving information confidentiality.

Another solution concerns real-time, on-line Intrusion Detection. To some extent, the problem of diagnosing unwanted usage patterns can be regarded as intermediate among the prevention and reaction phase. By monitoring network segments, and observing some significant traffic properties, we aim at being able to infer whether any anomalous activity is going on. In order to do that, we need to find a model which is appropriate to synthesize traffic properties allowing to distinguish between acceptable and anomalous behaviors. We propose a solution based on techniques coming from the field of pattern recognition and artificial intelligence. Starting from raw packets, the value of parameters describing traffic properties of interest must be calculated.

## 3 Stealth Attacks Detection

In this work we focus on low rate DoS attacks exploiting the TCP retransmission property for denying service to legitimate users. As stated in [1], low-rate denial of service attacks, unlike high-rate attacks, are difficult for routers and counter-DoS mechanisms to detect. In the mentioned paper, authors present what they call a *shrew attack*, which is made of short traffic bursts characterized by both a maliciously chosen duration and low frequency, specifically conceived with the aim to evade rate-controlling detection mechanisms.

Our approach to detection is the following: (i) generate synthetic traffic traces including low-rate attacks; (ii) define traffic metrics specifically suited for the behavioral modeling of a typical attack pattern [3]; (iii) based on the defined metrics, extract suitable behavior patterns for attack classification from raw traffic.

Starting from the recorded traffic traces, and based on the set of features described above, the so-called *feature vectors* characterizing our traffic mix have been computed. We have looked after traffic model extraction by means of the following supervised machine learning algorithms: (i) J48 Decision Tree; (ii) *Support Vector Machine* , i.e. a non-probabilistic binary linear classifier; (iii) Bayesian Network ; (iv) Boosting .

## 4 Trials and Experimentations

The experimental evaluation of the proposed attack detection and reaction framework has been carried out using a testbed, in order to work in an isolated, protected and controllable environment, which is depicted in Fig. 1.

In the low-rate attack detection scenario, we reproduced a realistic network environment, involving both the SCADA components and the network connecting them, as well as the interconnection with a public network exploited by both legitimate users, and by attackers performing their malicious actions. The attackers target a web server and, along the path from the attacker to the victim, attack packets flow through one of the routers shared with the SCADA system. We have tested the detection capabilities of the detection approach we propose, as well as the ability to circumvent and isolate the router involved in the attack path. In order to do that, we have deployed the following tools:

**Background Traffic Generator**[2] reproducing both TCP and UDP traffic, and traffic related to VoIP conversations.

**User Traffic Generator**[3] reproducing browsing patterns of a number of users of a web server.

**SCADA Traffic Generator** a real SCADA server connected with a number of RTU's which were active and producing traffic during the experiments

**Path Selector** selects the most convenient path for SCADA traffic. Operates by taking several criteria into account, including the presence of alerts generated by the IDS, or reports about congested routers.

**Network Configurator** practically enforcing the path selection policy provided by the Path Selector.

**Service Manager** listening for alerts coming from network monitoring probes, and relaying them to the Path Selector.

**Intrusion Detection System (IDS)** computes several parameters describing the activity of each flow, and the relationships between flows, and compares them with several traffic classification criteria calculated offline by means of artificial intelligence techniques.

**Attack Tool** Shrew attack toolkit, as described earlier.

In Fig. 1 we represent the actual hardware configuration of the testbed, which mainly consists of Linux boxes and two Juniper M10 routers. Machines identified by numbers 1 to 6 are MPLS-enabled routers, and represent the core of the SCADA communication infrastructure. Machine 7 represents the attack source, whereas machine 8 plays the role of the attack victim. The INSPIRE attack detection system is deployed close to machine 3, observing traffic flowing through it. A schematic *storyboard* of the experimental evaluation scenario can be described as follows. A SCADA system operates over an MPLS-enabled network. No impairment has been observed yet. At the RTU side, the network used by the SCADA is physically shared with other businesses of the same Organization. The attacker activates his tools and targets an application server. Attack traffic influences congestion windows of other intersecting traffic flows, and also affects the RTU. The IDS detects the ongoing attack and notifies the Path Selector of

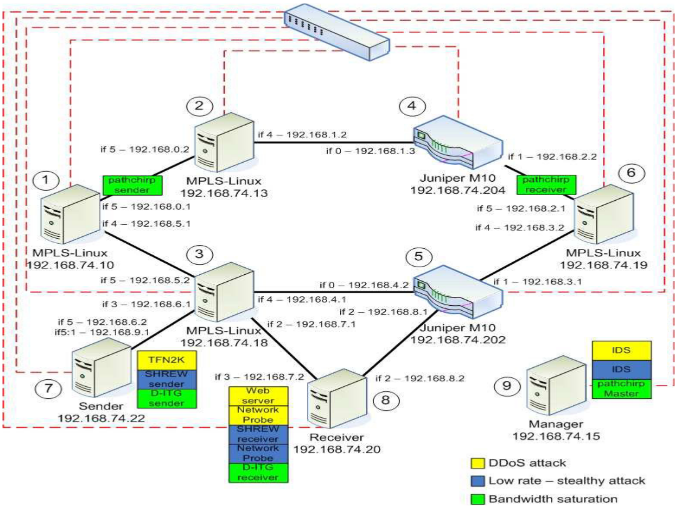


**Fig. 1.** The demo setup

the involvement of the router. Alternative paths are calculated and the traffic is re-routed along safe links. In case SCADA traffic is routed on link-disjoint paths (MPLS splitting), the path including the router involved in the attack is disabled. This countermeasure does not prevent or eliminate the attack, but it sustains business continuity of TSO's SCADA system. Furthermore, data confidentiality and integrity are ensured.

## 5 Conclusions

In this paper we presented a solution for protecting the communication infrastructures allowing information exchange among elements of a SCADA system. The developed Intrusion Detection tool has proven effective in detecting a stealth DoS attack. The detection process works as an effective trigger for flexible real-time network reconfiguration, performed by using MPLS, aimed at isolating resources under attack.

## References

1. Kuzmanovic, A., Knightly, E.W.: Low-rate tcp-targeted denial of service attacks and counter strategies. IEEE/ACM Transactions on Networking 14(4), 683–696 (2006)
2. Avallone, S., Manetti, V., Mariano, M., Romano, S.P.: A splitting infrastructure for load balancing and security in an mpls network. In: Proceedings of TridentCom 2007, Orlando, FL, USA. IEEE (May 2007)
3. Lee, W., Stolfo, S.J.: A framework for constructing features and models for intrusion detection systems. ACM Transactions on Information and System Security (TISSEC) 3(4), 227–261 (2000)

# Risk Assessment in Critical Infrastructure Security Modelling Based on Dependency Analysis
## (Short Paper)

Thomas Schaberreiter[1], Kati Kittilä[2], Kimmo Halunen[2],
Juha Röning[2], and Djamel Khadraoui[1]

[1] Centre de Recherche Public Henri Tudor, Service Science & Innovation (SSI),
29, Avenue John F. Kennedy, L-1855 Luxembourg, Luxembourg
`{firstname.lastname}@tudor.lu`

[2] University of Oulu, Department of Electrical and Information Engineering,
P.O. Box 4500, FIN-90014 University of Oulu, Finland
`ouspg@ee.oulu.fi`

**Abstract.** Critical infrastructure (CI) services are consumed by the society constantly and we expect them to be available 24 hours a day. CIs can be mutually dependent on each other and a failure in one infrastructure can cascade to another interdependent infrastructure to cause service disruptions. Methods to better assess and monitor CIs and their interdependencies in order to predict possible risks have to be developed. In this work, we present a method for CI analysis to identify critical entities in CIs at a management/organisational level as well as at a technical level supported by the PROTOS-MATINE model for dependency analysis.

**Keywords:** Critical infrastructure, risk, dependency analysis, security modelling.

## 1 Introduction

Critical infrastructures (CI) provide services that build the centre of our society and economy. To operate complex systems like CIs can be problematic and CI providers put substantial effort into keeping CIs running and reduce risks of any kind, for example the risk of failure, the risk of intrusion or the risk of incorrect operation. In related work, CI security modelling was introduced in [2],[1]. The idea of CI security modelling is to represent CIs as *critical services* and the *dependencies* between the services for on-line monitoring of the CI service state. To be able to capture the interdependencies between CI sectors, CI service risk is used as a common representation of the on-line state of CI services. The risk of a breach of confidentiality, a breach of integrity of the risk of degrading availability (CIA) are taken into account. The on-line risk is derived from observing *base measurements* in the CI systems. In this work one important aspect of the

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 213–217, 2013.

CI security model will be covered: CI analysis. To be able to break down the complexity of CIs and be able to capture the critical services and their dependencies, CIs have to be analysed. We utilise the PROTOS-MATINE model for dependency analysis, a qualitative dependency analysis method that has the advantage of capturing the structure of a complex system quickly and adequately by combining multiple information sources. The method initially focused on the interdependencies of network protocols and produced the PROTOS-MATINE model and the semantic tool Graphingwiki [4],[5]. The method has been successfully applied to finding dependencies in antivirus software [3] and assessing the socio-technical security of a VoIP system [6].

## 2 Critical Infrastructure Analysis

This Section describes how the PROTOS-MATINE method is used to gather the necessary information sources to be able to analyze CIs and represent them as critical services, dependencies between critical services and base measurements. Furthermore, it is discussed how the constructed model can be checked for validity.

### 2.1 Gathering the Information Sources

We identify three abstract organizational classes that capture the main difference in tasks, objectives and responsibilities in a CI: The *preparedness and business continuity level*, the *process level* and the *technical level*.

The highest level of abstraction can be seen as the preparedness and business continuity level. At this level, the most important questions are: What is the mission of the CI and how does the CI function under normal circumstances, during an (security) incident and during an exceptional crisis? What are the high level connections and agreements with other companies or organizations? How do these affect the CIA of the services that the CI provides? The information gathering efforts at this level focus on actors, criticality and service levels. Potential data sources include: Passive network intelligence; Interviews, News; Documentation; Supply chain; Contracts, Policies; Laws, Mandates and Regulation; Incident data; Interviews with high-level management.

The focus at the process level is on the processes, which the organisation or system incorporates. This level considers human resources in the mid-level management who are responsible for the day to day operations and putting the high level policies and strategies into action. At this level, the most important questions are: Who does, is responsible and/or supervises the normal functionality of operations as well as backup systems, personnel or temporary replacement? How do these affect the CIA of the processes in the CI? Here the data sources include: Networks, Systems, Inventory; Documentation; Processes and life cycles; Services and related input/output dependencies; Interviews with mid-level management.

The technical level includes mostly technical details of the system and its functionality. At this level, the questions to be answered are: What is the system composed of and how are these components linked to each other? How is the security implemented and what about security controls? What about backup systems/methods? How is vulnerability management and product/service secure development life cycle taken care of? Who is responsible for the operation and maintenance of the systems/services? How do the different states of these systems/services affect the CIA of the systems/services? The data sources include: Vulnerability feeds and related product information about systems; Standards; Logs and Sensors; Documentation; Network traffic; External requirements; Interviews with technical staff.

### 2.2 Constructing the Dependency Model

**Critical Service Identification.** To be able to capture and comprehend the complex structure of a critical infrastructure it is seen as inevitable to perform some kind of decomposition of the infrastructure into smaller, easier to understand entities.

Taking into account the service oriented approach of the security model, CIs are decomposed by identifying services and the systems used to provide those services. Each identified service can be a real service provided by the CI or it can be a logical entity that enables better understanding about the system at a higher level. Each identified service can be investigated separately and it can be further decomposed into its constituent parts.

The closer a service is to the root of the tree, the more general it will be in function. Those services will be mainly composed of sub-services and less of actual infrastructure. The closer a service is to the leaves of the tree, the more specific it will be in function. Those services will be mainly composed of infrastructure and only a few additional sub-services. The lowest level of each path through the decomposition tree should contain infrastructure that is utilized when providing the service.

Identifying services with the PROTOS-MATINE method can be done in several phases and abstraction levels. First it is important to identify the major services in the whole CI and the possible constraints and service levels that laws and regulations pose to these. After this initial step, the PROTOS-MATINE method can be used iteratively to identify services and the sub-services of these until the desired level of detail has been reached. The information gathering can be done at the levels identified in the PROTOS-MATINE model. The important thing is to utilise all information sources that are available and suitable for the given abstraction level. Also updating the model throughout the process is crucial as deeper investigation can reveal information relevant to the higher level model.

**Critical Service Dependencies.** The dependency analysis in this step is based on the infrastructure decomposition tree. Each identified service is examined separately and the dependencies of this service to other services, either from the investigated infrastructure or from an external infrastructure are evaluated and identified. The result of this step is a dependency graph that contains critical services as notes and their dependencies as edges.

Identifying dependencies using the PROTOS-MATINE method can be done after service identification or in parallel (to help understand the structure for service decomposition). Dependencies can be identified at every level of abstraction and at each level the sources from which these dependencies arise are slightly different. At the preparedness and business continuity level good sources are for example service contracts and SLAs. At the technical level technical documents give valuable information about devices which are needed to perform the service in question. Interviews are a very important data source when identifying dependencies at any level. At high levels, management usually has important information about service dependencies. At low levels interviews with technical staff might reveal hidden dependencies which are impossible to perceive only from technical documents.

### 2.3 Observing the Dynamic Behaviour

**Base Measurement Identification.** This part of the process investigates, for each identified service, the CI components to examine what kind of measurements can be utilized from it in order to determine the current state of the service. Base measurements can be various measurable indicators, ranging from sensor outputs located at CI system components or software measurements from ICT (Information and Communication Technologies) systems.

Identifying base measurements in PROTOS-MATINE is done at the lowest abstraction level. Information should be gathered from expert interviews, standards and/or industry best practises. These measurements should be representative of the system behaviour and relevant to the three attributes that are to be measured. To be able to find a good subset of base measurements to determine a CI service state, they should be ranked according to their assumed importance. If there are constraints on the amount of measurements that can be utilised, then only the highest ranking measurements can be used.

### 2.4 Model Checking

As mentioned before, dependency analysis is a recursive process. After each iteration of the recursion, the model has to be checked for validity. The questions that have to be asked in the model checking process are: Does the decomposition reflect the structure of the CI adequately?, Have the dependencies been taken into account correctly?, Were the right base measurements chosen?

The model checking process can be supported by emulation and simulation. The key here is to think about the most likely incidents and incident patterns and how they should reflect risk in the model. After those are identified, scenarios can be either emulated by setting the concerned base measurements to a value that should produce a certain risk and see how this is reflected in the model, or simulated by using logged data from previous incidents to produce more realistic scenarios. Note that the model should not be trimmed to only be able to detect known incidents, but an evaluation with known data and highly likely failure scenarios can help to improve the model and detect flaws, errors and wrong assumptions.

If at some point the model behaves as expected, the recursion can be stopped and the model can be deployed. If the desired result was not reached, another iteration of the recursion should be carried out in order to refine the model.

## 3 Summary and Future Work

In this paper, we presented a method for dependency analysis of CIs. Starting from a hierarchical decomposition of a CI into services and CI systems/components that are utilized to provide those services, dependencies to other infrastructure services are identified. Those can be dependencies to internal services or to services provided by another CI. Furthermore, this method helps to extract observable infrastructure entities (base measurements) used for on-line monitoring of CI services.

The information gathering process of the CI analysis is supported by the PROTOS-MATINE model, which uses different information sources to evaluate the social and technical aspects of a CI at different levels of the organizational structure to identify services and their dependencies as well as the base measurements.

Future work will focus on the validation of the proposed approach based on an industrial case study.

## References

1. Aubert, J., Schaberreiter, T., Incoul, C., Khadraoui, D.: Real-time security monitoring of interdependent services in critical infrastructures. Case study of a risk-based approach. In: 21st European Safety and Reliability Conference, ESREL 2010 (September 2010)
2. Aubert, J., Schaberreiter, T., Incoul, C., Khadraoui, D., Gateau, B.: Risk-based methodology for real-time security monitoring of interdependent services in critical infrastructures. In: International Conference on Availability, Reliability, and Security (ARES 2010), pp. 262–267 (February 2010)
3. Eronen, J., Karjalainen, K., Puuperä, R., Kuusela, E., Halunen, K., Laakso, M., Röning, J.: Software vulnerability vs. critical infrastructure - a case study of antivirus software. International Journal on Advances in Security 2, 72–89 (2009)
4. Eronen, J., Laakso, M.: A case for protocol dependency. In: IEEE International Workshop on Critical Infrastructure Protection, pp. 22–32 (2005)
5. Eronen, J., Röning, J.: Graphingwiki - a semantic wiki extension for visualising and inferring protocol dependency. In: Proceedings of the First Workshop on Semantic Wikis – From Wiki to Semantics, ESWC 2006 (June 2006)
6. Pietikäinen, P., Karjalainen, K., Eronen, J., Röning, J.: Socio-technical security assessment of a VoIP system. In: The Fourth International Conference on Emerging Security Information, Systems and Technologies, SECURWARE 2010 (July 2010)

# Countermeasures Selection via Evidence Theory
## (Short Paper)

Giusj Digioia, Chiara Foglietta, Gabriele Oliva, and Stefano Panzieri

Dipartimento di Informatica e Automazione, University ”Roma TRE”,
Via della Vasca Navale, 79, 00146, Roma, Italy
{digioia,fogliett,oliva}@dia.uniroma3.it,
panzieri@uniroma3.it

**Abstract.** In this paper an approach to understand the possible causes of outages in different and interconnected infrastructures, based on the evidences of detected failures is provided. Moreover, causes inferred are used to estimate possible not detected failures that, together with those detected, allow to better understand the infrastructure vulnerability and the impact of outages. Such a kind of analysis is regarded as a useful support to identify effective countermeasures, in order to mitigate risks related to malfunctioning behavior of critical infrastructures.

**Keywords:** Critical-Infrastructures, Interdependency Modelling, Evidence Theory, Situation Awareness, Risk Assessment.

## 1 Introduction

The concept of Situation Awareness [1] is gaining momentum rapidly, and is more and more drawing the attention of the scientific community. It seems therefore natural to address the problem of situation awareness in the field of Critical Infrastructures [2,3].

While interdependency models are used to foresee the evolution of the state of the infrastructures, based on the actual state, Situation Awareness refers to present time and, potentially provides more insight on the actual state with respect to interdependency models; hence it may successfully complement interdependency analysis. Moreover, as shown in figure 1, both methodology can be used together in order to derive adequate countermeasures to adverse events.

Figure 1.(a) shows traditional analysis performed by means of interdependency models; field data is used as input and the expected evolution of the state of the different infrastructures is calculated. Figure 1.(b) shows a possible architecture involving both situation awareness and interdependency analysis: a ”cause identification” module is introduced with the aim to determine the causes of the experienced outage. The expected evolution is then foreseen by means of an interdependency model, fed by the sensorial data, as well as by the output of the cause identification module; finally the data elaborated are provided to a module devoted to identify adequate countermeasures to cope with the scenario

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 218–222, 2013.

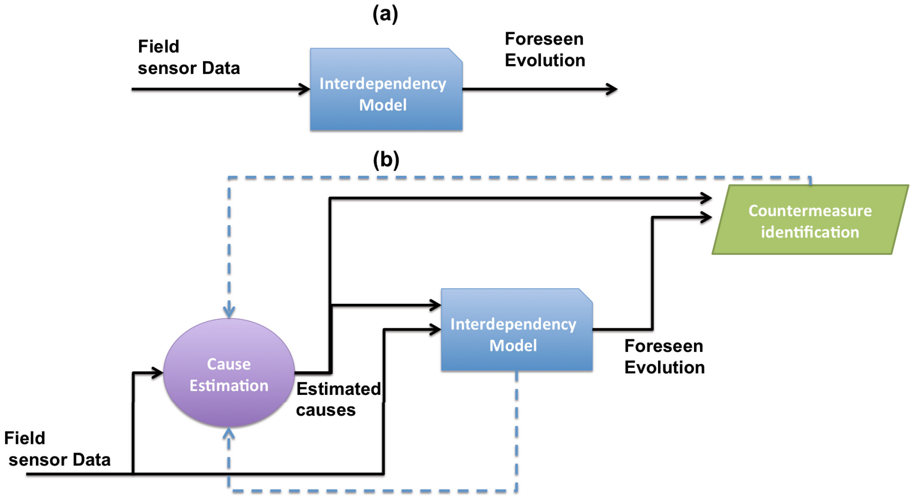


**Fig. 1.** Traditional interdependency modeling usage (a) and cooperation between interdependency modeling and situation awareness with the aim to identify suitable countermeasures (b)

at hand. Notice that a feedback towards cause identification module might be introduced at every level (i.e., blue dotted lines in Figure 1.(b)).

In this paper based on Dempster-Shafer [4] and Smets [5] theory of evidence, an approach to understand the possible causes of outages in different and interconnected infrastructures, is proposed. In the future, the results of this procedure are the inputs for a interdependence modeling of critical infrastructures that can better evaluate risks.

The paper is organized as follows: the proposed framework is detailed in Section 2, where a small but explicative case study is reported, showing the potentialities of the proposed approach; finally, some conclusive remarks and future work directions are collected in Section 3.

## 2 Case Study with Simulation Results

As stated before, Situation Awareness theory is mainly based on the experience gathered by the scientific community in the field of Multisensor Data Fusion. The reference model describing the Data Fusion process is the JDL model (Joint Directors of laboratories) described in [6]. The JDL model is a five-level architecture. Each level is a proposed functional step characterized by level of abstraction for inputs and outputs.

The idea proposed in this paper is to apply the concept of Situation Awareness to critical infrastructure domain. The Theory of Evidence is a formalism which can be used for modeling uncertainty instead of classical probability.

Let a bipartite graph $G = (V_1, V_2, E)$ where $V_1, V_2$ are the set of $m$ causes and $n$ faults, respectively and E only contains direct edges in the form $(v_i^1, v_j^2)$, where $v_i^1 \in V_1$ and $v_i^2 \in V_2$ (see Table 1). Let $x_0 \in \mathbb{R}^n$ be the vector of the faults, where the i-th component $x_{i0} \in [0,1]$ represents the severity of failure in the $i$-th element.

In order to reduce the cardinality of the power set $\Omega$ consider only the focal sets as the subset of $\Omega$ supported by a non-null faults, according to the graph $G$. In this way it is possible to reduce the size of the power set by considering only $2^{\Psi}$.

Let $\Psi_j$ be the subset of $\Psi$ supported by the j-th failure (i.e., the causes which have an outgoing edge that goes into the j-th failure node) and let $k$ be the maximal cardinality of a subset of $\Psi_j$, let $h$ be the number of sets with cardinality $k$ in $\Psi_j$ and set $\alpha = x_{j0}$; then

1. Assign a mass equal to $\frac{\alpha}{2h}$ to each set of cardinality $k$
2. Set $k = k - 1$, set $\alpha = \frac{\alpha}{2}$ and calculate $h$ as the number of set of cardinality $k$.
3. Repeat Step 1 until $k = 0$.

In order to take into account also the alternative let $\bar{\Psi}_j = \Psi - \Psi_j$ and assign the masses of its elements according to the above algorithm with $\alpha = 1 - x_{j0}$.

Once the masses are assigned to each subset of the power set, it is possible to calculate the belief function $Bel$ related to each singleton.

Consider this simple case study: two different zones of a power grid are controlled by two SCADA systems, through some RTUs. The connection between the SCADA systems and the related RTUs is granted by a telecommunication infrastructure. The vector of anomalies is composed of 5 variables: the failure on the branch of the power grid in zone 1 and in zone 2 (E1 and E2), the failure in the support SCADA system in the two zones (S1 and S2) and a cyber malicious intrusion in the TLC infrastructure (T-IDS), detected by an IDS sensor.

For what concerns the causes, 6 events were considered: natural disasters (i.e., earthquake) both in zone 1 and 2 (N1,N2), cyber attacks in zone 1 and zone 2 (C1 and C2) and physical attack (i.e. cut of an electric wire, causing a local damage) in zone 1 and zone 2 (P1, P2), see Table 1. Physical attacks affect both SCADA and power grid in a given zone; natural disasters attacks affect both SCADA and power grid in a given zone and cyber attacks in a given zone affect

**Table 1.** Case Study Model represents as adjacency matrix

| | $\mathbf{N_1}$ | $\mathbf{P_1}$ | $\mathbf{C_1}$ | $\mathbf{N_2}$ | $\mathbf{P_2}$ | $\mathbf{C_2}$ |
|---|---|---|---|---|---|---|
| $\mathbf{S_1}$ | 1 | 1 | 1 | 0 | 0 | 0 |
| $\mathbf{E_1}$ | 1 | 1 | 0 | 0 | 0 | 0 |
| $\mathbf{T-IDS}$ | 0 | 0 | 1 | 0 | 0 | 1 |
| $\mathbf{S_2}$ | 0 | 0 | 0 | 1 | 1 | 0 |
| $\mathbf{E_2}$ | 0 | 0 | 0 | 1 | 1 | 1 |

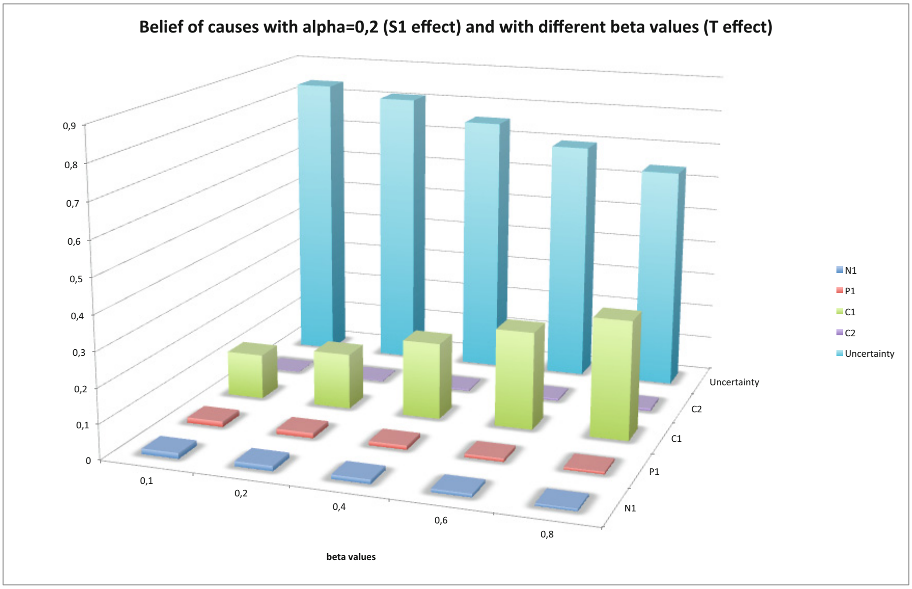


**Fig. 2.** Results for alfa = 0.2 when beta changes

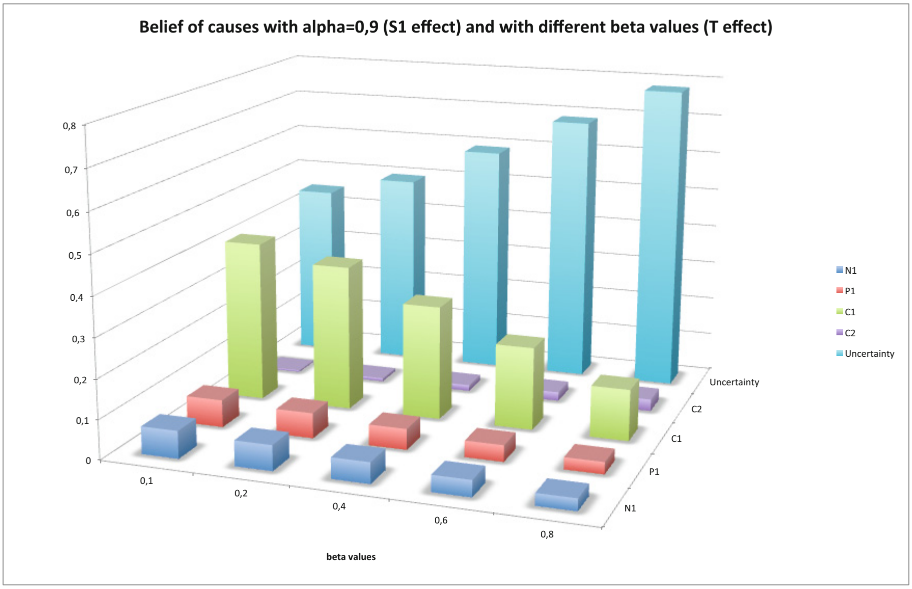


**Fig. 3.** Results for alfa = 0.9when beta changes

the SCADA system of that zone, as well as the TLC network (which is the same for both zones).

In the following simulation it can be noticed that the approach proposed is able to identify an occurring malicious intrusion in zone 1, causing the malfunctioning of the power grid, excluding the natural disasters and physical attacks as causes of the malfunctioning behavior. It has been considered that the state of $S1$ was equal to $\alpha$ while the state of $T - IDS$ was $\beta$, and some results are plotted for different values of $\beta$ (see figures 2, 3). It can be noticed that low values for $\beta$, make the uncertainty about causes high and that the belief of a cyber attack in zone 1 increases with high values for $\beta$.

## 3 Conclusions

In this paper, the Situation Awareness approach has been applied to Critical Infrastructure in order to increase the awareness about causes of malfunctioning, such as natural disasters or malicious events.

Our belief is that the framework of Situation Awareness suits the context of protection of Critical Infrastructures, in fact the understanding of malfunctioning behaviour causes allows to estimate possible not detected failures, whose identification is crucial to evaluate the vulnerability of infrastructures and the impact of outages.

## References

1. Endsley, M.: Toward a theory of situation awareness in dynamic systems: Situation awareness. Human Factors 37(1), 32–64 (1995)
2. Digioia, G., Foglietta, C., Oliva, G., Panzieri, S., Setola, R.: Moving from looking to understanding: Situation awareness and prediction. In: Flammini, F., Setola, R., Franceschetti, G. (eds.) Effective Surveillance for Homeland Security: Balancing Technology and Social Issues, pp. 74–92. CRC Press/Taylor & Francis (2013)
3. Digioia, G., Foglietta, C., Oliva, G., Panzieri, S.: Aware on-line interdependency modeling via evidence theory. International Journal of Critical Infrastructures 9, 74–92 (2013)
4. Dempster, A.: A Generalization of Bayesian Inference. In: Yager, R., Liu, L. (eds.) Classic Works of the Dempster-Shafer Theory of Belief Functions. STUDFUZZ, vol. 219, pp. 73–104. Springer, Heidelberg (2008), http://dx.doi.org/10.1007/978-3-540-44792-4_4
5. Smets, P.: Data Fusion in the Transferable Belief Model. In: Proceedings of the Third International Conference on Information Fusion, FUSION 2000, vol. 1, pp. PS21–PS33 (2000), http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=862713
6. Hall, D.L., Llinas, J.: Handbook of Multisensor Data Fusion. CRC Press (June 2001), http://www.worldcat.org/isbn/0849323797

# The Robustness of Assortativity
## (Short Paper)

Antonio Scala[1,3,4] and Gregorio D'Agostino[2]

[1] ISC-CNR, Physics Dept. University of Rome " Sapienza"
P.le Moro 5 00185 Rome Italy
[2] ENEA CR "Casaccia"
via Anguillarese 301 I-00123 Roma (RM) Italy
[3] London Institute for Mathematical Sciences, 22 Audley Street, London UK
[4] IMT Alti Studi Lucca, piazza S. Ponziano 6, 55100 Lucca, Italy

**Abstract.** Complex networks are ubiquitous in real word and represent a key model for both human made and natural systems. An important characteristics that distinguishes technological networks from biological networks is the assortativity, i.e. the correlation among the degrees of connected nodes. We apply spectral analysis to investigate how assortativity influences the robustness of a network with respect to failure propagations or epidemic spreading. We find a no free lunch situation: while disassortative networks are more robust since they have a higher failure threshold, in assortative networks there is more time for intervention before total breakdown.

## 1 Introduction

Complex Networks have been applied to a wide range of sectors, from technological fields like the Internet or power grids to biological fields like genomics or ecosystems [1,2]. A network is anything that can be represented by a set of elements called nodes connected by links representing some relationship among nodes: as an example, in social networks the nodes are people and the links between them can be relationships like friendship, political alliance or collaboration. The structure of the networks is linked to topological metrics like the degree distribution (the degree of a node is the number of its neighbours) and it plays a key role in determining the robustness, the resilience ad the response of a network [3]. Real networks in most cases show non-trivial topological correlations; in particular, many networks show ''assortative mixing'' on their degrees, i.e. high-degree vertices tend to be attached to high-degree ones, while other networks show disassortative mixing, i.e. high-degree vertices tend to be attached to low-degree ones. The network's degree–degree correlation can be quantified by a single scalar $\alpha$ called the assortativity coefficient [4] which assumes values $\alpha = 0$ for degree-uncorrelated networks, $\alpha > 0$ for assortative networks and $\alpha < 0$ for disassortative networks. Assortative correlations are typically observed in social networks [4]; on the other hand, disassortative connections are mainly found in technological and biological networks [5]. We want to investigate the consequences of the assortativity on the characteristics of a network.

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 223–226, 2013.

## 2 Monte Carlo

To randomize a networks one possible procedure consists into reshuffling links while keeping the degree of each node constant [6]; it has already been noticed that link-swap moves can be assortative, disassortative or neutral [7]. We introduced a means to sample the space of networks of different assortativity sharing the initial degree distribution. While our procedure is general, in this paper we will concentrate on initial network configurations obtained by the Barabasi-Albert preferential attachment procedure [8].

We define a fictive energy $H(G) = -\sum_{ij} k_i A_{ij} k_j$ that has the property that on average $H$ decreases if the assortativity increases and vice-versa We can therefore use the fictive energy $H$ to sample the space of assortative networks via a Monte Carlo procedure in which we assign the weight $\propto \exp\left[-\beta H(G)\right]$ to the configuration $G$ and we accept a link reshuffling move with probability $\exp\left\{-\beta\left[H(G') - H(G)\right]\right\}$. The parameter $\beta$ looks like the analogous of an inverse temperature in the canonical ensemble, but in order to be able to sample both assortative and disassortative configurations we have to allow $\beta$ to be both positive and negative. The resulting sampling of the assortativity $\alpha$ respect to the parameter $\beta$ is monotonously increasing.

## 3 Spectral Analysis

A powerful tool in assessing the general characteristic of a network is the spectral analysis of its associated matrices [9].

Formally, a network (or a graph) is defined as a couple $G = (V, E)$ where $V$ is the set of $N_V$ nodes and $E$ is the set of $N_E$ links; each link joins two nodes. To each graph $G$ we associate its adjacency matrix $A$, defined as $A_{ij} = 1$ if nodes $i$,$j$ are connected, $A_{ij} = 0$ otherwise. The networks we are considering are simple (no self loops, i.e. $A_{ii} = 0$) and undirected ($A_{ij} = A_{ji}$). The degree of node $i$ is therefore $k_i = \sum_j A_{ij}$ ; nodes are labelled for increasing degree: $k_1 \leq k_2 \leq \ldots \leq k_N$.

The eigenvalues of $A$ are real as $A$ is Hermitian (we are considering undirected networks); moreover $\Lambda_N$ is positive as $A$ is a positive matrix.

The propagation of epidemics on networks is clearly linked to the adjacency matrix $A$ that dictates which nodes can be infected by a virulent node; moreover the dynamics of epidemics in certain cases can be related to the dynamics of failure propagation. The maximum eigenvalue $\Lambda_N$ has a particular status as it is linked to the epidemic threshold. The epidemic threshold $\tau$ of a network can be thought as the fraction of nodes to immunize in order to stop an infection with a fixed disease propagation rate; Wang and coauthors have shown that in networks the epidemic threshold scales as $\tau \sim 1/\Lambda_1$ [10,11].

We find that $\Lambda_1^{-1}$ decreases with assortativiy: in the range of correlation we explore, disassortative networks show an epidemic threshold up to 20% higher than assortative ones (Fig.1). Our findings confirm the idea that avoiding direct connections between hubs (highly connected nodes) may provide protection against epidemics [12].

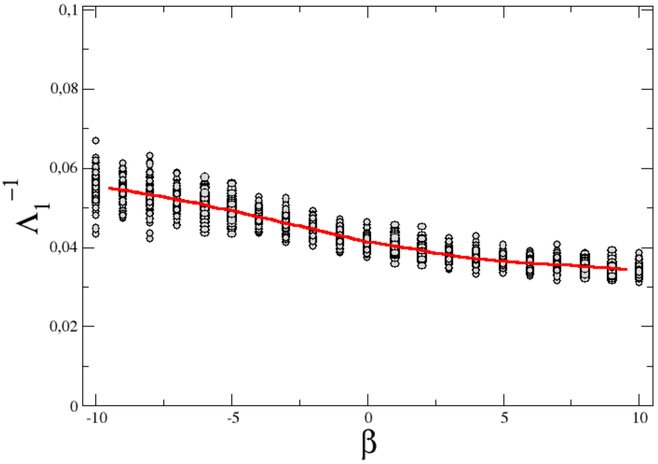


**Fig. 1.** The epidemic threshold $\Lambda_1^{-1}$ decreases with assortativity (networks of 10000 nodes)

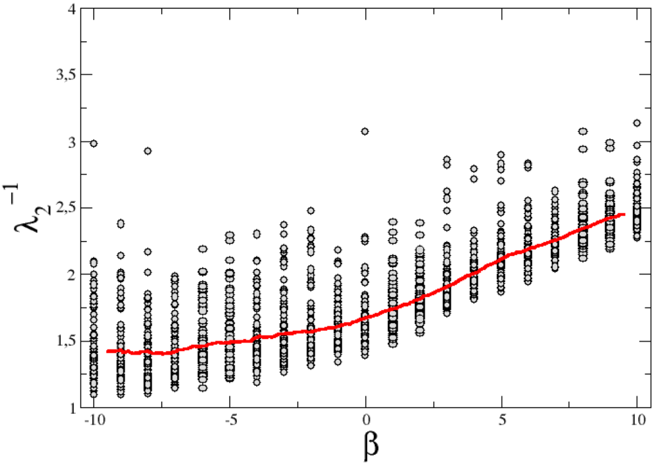


**Fig. 2.** For diffusion-like processes, the longest time to explore the network is proportional to $\lambda_2^{-1}$. For our networks of 10000 nodes, it increases with the assortativity.

The Laplacian matrix of a network is defined as $L \equiv D - A$, where $D$ is the diagonal matrix of degrees $D_{ij} = k_i \delta_{ij}$. It is the analogous of the Laplacian operator and describes the diffusion of random walkers on the network. The eigenvalues of $L$ are $\lambda_1 = 0 \leq \lambda_2 \leq \ldots \leq \lambda_N$; the eigenvector (mode) associated to the zero-th eigenvalue $\lambda_1$ is the equilibrium distribution for a diffusive process on the network. The first non-zero eigenvalue $\lambda_2$ is the inverse diffusional timescale of slowest mode, i.e. it is a measure of the longest time for a random walker to explore the whole network. Therefore, a lower value $\lambda_2^{-1}$ means that there is less time for intervention before a network is totally compromised by randomly propagating failures or epidemics; in such respect assortative networks show times up to 60% higher than disassortative ones (Fig. 2).

## 4 Conclusions

We have investigated via spectral methods some effects of the assortativity on the robustness of a network with respect to randomly propagating failures and epidemics. We have found a "no free lunch" situation: while disassortative networks have a higher failure threshold, assortative networks give more time for intervention before total breakdown.

**Acknowledgements.** AS thanks US grant HDTRA1-11-1-0048, CNR-PNR National Project "Crisis-Lab" and EU FET project MULTIPLEX nr.317532. The contents do not necessarily reflect the position or the policy of funding parties.

## References

1. Caldarelli, G.: Scale free networks. Oxford University Press, Oxford (2007)
2. Buchanan, M., Caldarelli, G., De Los Rios, P., Rao, F., Vendruscolo, M. (eds.): Networks in cell biology. Cambridge University Press, Cambridge (2010)
3. Caldarelli, G., Vespignani, A. (eds.): Large scale structure and dynamics of complex networks. World Scientific, Singapore (2007)
4. Newman, M.E.J.: Assortative mixing in networks. Phys. Rev. Lett. 89(20), 208701 (2002)
5. Newman, M.E.J.: Mixing patterns in networks. Phys. Rev. E 67(2), 026126 (2003)
6. Maslov, S., Sneppen, K.: Specificity and Stability in Topology of Protein Networks. Science 296(5569), 910–913 (2002)
7. Zhou, S., Mondragón, R.J.: Structural constraints in complex networks. New Journal of Physics 9(6), 173 (2007)
8. Barabasi, A.L., Albert, R.: Emergence of scaling in random networks. Science 286, 509–511 (1999)
9. Chung, F.R.K.: Spectral Graph Theory. CBMS Regional Conference Series in Mathematics, vol. 92. American Mathematical Society (February 1997)
10. Wang, Y., Chakrabarti, D., Wang, C., Faloutsos, C.: Epidemic spreading in real networks: An eigenvalue viewpoint. In: SRDS, pp. 25–34 (2003)
11. Chakrabarti, D., Wang, Y., Wang, C., Leskovec, J., Faloutsos, C.: Epidemic thresholds in real networks. ACM Trans. Inf. Syst. Secur. 10, 1:1–1:26 (2008)
12. Eguíluz, V.M., Klemm, K.: Epidemic threshold in structured scale-free networks. Phys. Rev. Lett. 89(10), 108701 (2002)

# Author Index