A Comparative Study of Statistical Models with Long and Short-Memory Dependence for Network Anomaly Detection

Tomasz Andrysiak, Łukasz Saganowski and Adam Marchewka

Abstract Protection of systems and computer networks against novel, unknown attacks is currently an intensively examined and developed domain. One of possible solutions to the problem is detection and classification of abnormal behaviors reflected in the analyzed network traffic. In the presented article we attempt to resolve the problem by anomaly detection in the analyzed network traffic described with the use of five different statistical models. We tested two groups of models which differed in autocorrelation dependences. The first group was composed of AR, MR and ARMA models which are characterized by short memory dependences. The second group, on the other hand, included statistical attempts described with ARFIMA and FIGARCH models which are characterized by long memory dependences. In order to detect anomalies in the network traffic we used differences between real network traffic and its estimated model. Obtained results of the performed experiments show purposefulness of the conducted comparative study of exploited statistical models.

Keywords Anomaly detection · Statistical models · Network traffic prediction

1 Introduction

For many years, there have been used safety systems based on formerly isolated and classified patterns of threats, named signatures. Anti-virus software, systems for detection and breaking-in counteraction and protection against information leaks are just examples from a long and diversified list of application of those techniques.

T. Andrysiak (✉) · Ł. Saganowski · A. Marchewka
UTP University of Science and Technology Institute of Telecommunications,
ul. Kaliskiego 7, 85-789 Bydgoszcz, Poland
e-mail: andrys@utp.edu.pl

Ł. Saganowski
e-mail: luksag@utp.edu.pl

A. Marchewka
e-mail: adimar@utp.edu.pl

R.S. Choraś (ed.), *Image Processing and Communications Challenges 7*,
Advances in Intelligent Systems and Computing 389,
DOI 10.1007/978-3-319-23814-2_29

Nevertheless, there is one aspect in common, namely, they are able to protect systems and computer networks from known attacks described by the mentioned patterns. However, does lack of traffic matching known signatures mean there is no threat?

A means to defend from novel, unknown attacks is a rather radical change in operation concept. Instead of searching for attack signatures in network traffic it is necessary to browse for abnormal behavior which is a deviation from the normal traffic characteristic. The strength of such an approach is visible in solutions which are not based on knowledge a priori of attack signatures but on what does not respond particular norms, profiles of the analyzed network traffic. The techniques based on the above mentioned assumptions should be able to detect both: simple attacks of DoS type (Denial of Service) or DDoS (Distributed Denial of Service), and intelligent network worms up to hybrid attacks which are a combination of numerous different destruction methods. The consequence of such kind of attacks is inception of network anomalies, which creates a possibility to detect them, or even prevent from unwanted actions. The hardest challenge, however, is differentiation between dangerous behavior and normal movement in its initial stage in order to limit the usage of network resources. Anomalies are abnormalities, variations from the adopted rule. Anomalies in network traffic can signify device damage, an error in software or attack on resources and network systems. The essence of anomaly disclosure in computer networks is therefore detecting abnormal behaviors or actions which in particular can constitute a source of a potential attack [6]. One of possible solutions to the presented problem is implementation of Anomaly Detection Systems. They are currently used as one of the main mechanisms of safety supervision in computer networks. Their action consists in monitoring and detecting attacks directed onto information system resources on the basis of abnormal behaviors reflected in parameters of network traffic. Anomaly detection methods have been a topic of numerous surveys and review articles. In works describing the methods there were used techniques consisting in machine learning, neural networks, clustering techniques and expert systems. At present, anomaly detection methods that are particularly intensively developed are those based on statistical models describing the analyzed network traffic [6]. The most often used models are autoregressive ARMA or ARIMA, and Conditional Heteroscedastic Models ARCH and GARCH, which allow to estimate profiles of a normal network traffic [18]. In the present article we propose using estimation of statistical models AR, MR, ARMA, ARFIMA and FIGARCH for defined behavior profiles of a given network traffic. The process of anomaly detection (a network attack) is realized by comparison of parameters of a normal behavior (predicted on the basis of the tested statistical models) and parameters of real network traffic. This paper is organized as follows. After the introduction, in Sect. 2 we present the definition of long and short memory dependence. In Sect. 3 the different statistical models for date traffic prediction are described in details. Then, in Sect. 4 the Anomaly Detection System based on AR, MR, ARMA, ARFIMA and FIGARH model estimation is shown. Experimental results and conclusion are given thereafter.

2 Definition of Long and Short-Memory Dependence

Long memory dependences manifest themselves in the existence of autocorrelations of elements creating the given time series. In most cases it is high-order autocorrelation. This means that in the examined series there is a dependence between the observations—even those distant in time. This phenomenon is called long memory and was discovered by a British hydrologist Hurst [13]. In case of long memory existence the autocorrelation function is slowly falling at hyperbolic pace. The time series with long memory feature has in the spectral domain distribution of low frequency. Short memory time series, however, show essential autocorrelation of low frequency only. This means that observations that are separated even by a relatively short time period are no longer correlated. Short memory series are easy to recognize due to the fact that in the time domain the autocorrelation function disappears quickly, and in the spectral domain there are distributions of high frequency. It is said that the stochastic process has long memory with parameter d if its spectral density function $f(\lambda)$ meets the condition

$$f(\lambda) \sim c\lambda^{-2d}, when \quad \lambda \to 0^{+}, \tag{1}$$

where c is constant, and symbol $\sim$ means that the relation of left and right side is heading to one. When the process meets that condition and when $d > 0$ then its autocorrelation function is disappearing in hyperbolic manner [3, 4, 18] i.e.

$$\rho_k \sim c_\rho k^{2d-1}, when \quad k \to \infty. \tag{2}$$

Parameter d describes the memory of the process. When $d > 0$, the spectra density function in unlimited in surrounding 0. It is then said that the process has a long memory. When $d = 0$, the spectral density is limited in 0, and the process is described as having short memory. When $d < 0$, then the spectral density equals 0 and the process shows negative memory and is named anti persistent [10, 12].

3 Statistical Models for Network Traffic Prediction

The tested network traffic is represented by means of time series describing variance of parameters characterizing the number of received and sent TCP, UDP and ICMP packages within a time unit. A natural way of describing such series are statistical models which are based on autoregression and moving average in relation to differently realized data variances and autocorrelation of elements creating the given time series.

3.1 Short-Memory Models

In order to describe the properties of short memory time series (essential autocorrelations of low order only) the approach that is often applied is the use of solutions known as autoregression model AR, moving average MR and mixed models ARMA. They can be used for modeling stationary series, i.e. series where there are only random fluctuations around the average, or non-stationary reducible to a stationary form. Their composition is based on autocorrelation phenomenon, i.e. on correlation of the predicted variable value with values of the same variable but delayed in time [5].

Autoregressive Model Numerous time series are composed of interdependent observations which means that it is possible to estimate the models coefficients which describe the following elements of the series on the basis of the delayed in time previous elements of the series $\left(Y_{t-1}, Y_{t-2}, \ldots, Y_{t-p}\right)$, and random component ε_t in current period t. The above can be presented with the use of equation of autoregression of the order (p) as $AR(p)$

$$Y_t = c_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t, \tag{3}$$

where $\phi_1, \phi_2, \ldots, \phi_p$ describe the models parameters, c_0 is invariable and $\varepsilon_t \left(0, \sigma^2\right)$ is the white noise process with zero mean and variance σ^2. AR is a process with memory of previous realizations of the series. Such a process is called stationary, when there is a condition $p > 1$ and all roots of the polynomial $W(z) = 1 - \phi_1 z - \phi_2 z^2 + \cdots - \phi_p z^p$ of each module are greater than one. For such a model the prediction is built step by step by recurrent substitution of successive values. With stationary processes $AR(p)$ such a prediction is heading for the average value of the process, and the error variance of the forecast aims at the variance of the process.

Moving Average Model It is a linear model in which the realization of Y_t in the current period depends on realization of the random component ε_t in the current period and q in subsequent previous periods. It can be presented by means of an equation of moving average of order q as $MA(q)$

$$Y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}, \tag{4}$$

where $\theta_1, \theta_2, ..., \theta_p$ describe the models parameters, and $\varepsilon_t \sim \left(0, \sigma^2\right)$ is the white noise process with zero mean and variance σ^2. MA is a process with memory of previous values of the random component. Every MA process which can be reduced to a stationary autoregressive process is called invertible. In general case this condition is fulfilled when the roots of the polynomial $W(z) = 1 + \theta_1 z + \theta_2 z^2 + \cdots + \theta_p z^q$ lie outside the unit circle. The prediction made with the use of $MA(q)$ model is obtained in the recurrent way, as it seeks the average value.

Autoregressive Moving Average Model For a stationary series, instead of applying separate models of AR and MR classes, in order to describe the connections between observations from the subsequent periods we use autoregressive models of moving average [3], i.e. $ARMA(p, q)$ models with delay order (p, q) written as

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} \quad (5)$$

where $\phi_1, \phi_2, \ldots, \phi_p$, and $\theta_1, \theta_2, \ldots, \theta_p$ describe the models parameters, and $\varepsilon_t \sim \left(0, \sigma^2\right)$ is the white noise process with zero mean and variance σ^2. As a result, by means of a lower number of AR and MR components than separately for AR model and MR model, any linear process can be described, which is beneficial from the perspective of the models estimation possibility and its use in predicting. $ARMA$ process contains properties of both AR and MR which is most easily visible in decomposition of the ACF function. $ARMA$ model generates stationary process if its components are: stationary AR and reversible MA. The prediction made by means of $ARMA(p, q)$ model is obtained in a recurrent way.

3.2 Long-Memory Models

An interesting approach towards describing the features of long memory time series was the use of solutions with movable autoregressive averaging in the process of fractional differentiation. In a result $ARFIMA$ (Fractional Differenced Noise and Autoregressive Moving Average) model was created [10], which is a generalization of $ARMA$ and $ARIMA$ models. Another approach towards time series description was including conditional variance dependence of the process from its previous values using the $ARCH$ model (Autoregressive Conditional Heteroskedastic Model) introduced by Engel [6]. Generalization of this approach was the $FIGARCH$ model (Fractionally Integrated $GARCH$), which autocorrelation function of squared residuals of the model decreases in a hyperbolic way. Such a behavior of an autocorrelation function enables naming $FIGARCH$ a model of long memory in the context of the autocorrelation function of squared residuals of the model.

3.3 Introduction to ARFIMA Model

The Autoregressive Fractional Integrated Moving Average model called $AR-FIMA$ (p, d, q) is a combination Fractional Differenced Noise and Auto Regressive Moving Average which is proposed by Grange, Joyeux and Hosking, in order to analysis the Long-Memory property [10, 12].

The $ARFIMA(p, d, q)$ model for time series Y_t is written as:

$$\Phi(L)(1-L)^d y_t = \Theta(L)\varepsilon_t, \quad t = 1, 2, \ldots\Omega, \quad (6)$$

where y_t is the time series, $\varepsilon_t \sim (0, \sigma^2)$ is the white noise process with zero mean and variances σ^2, $\Phi(L) = 1 - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p$ is the autoregressive polynomial and $\Theta(L) = 1 + \theta_1 L + \theta_2 L^2 + \cdots + \theta_p L^q$ is the moving average polynomial, L is the backward shift operator and $(1 - L)^d$ is the fractional differencing operator given by the following binomial expansion:

$$(1 - L)^d = \sum_{k=0}^{\infty} \binom{d}{k} (-1)^k L^k \tag{7}$$

and

$$\binom{d}{k} (-1)^k = \frac{\Gamma(d+1)(-1)^k}{\Gamma(d-k+1)\Gamma(k+1)} = \frac{\Gamma(-d+k)}{\Gamma(-d)\Gamma(k+1)}, \tag{8}$$

where $\Gamma(*)$ denotes the gamma function and d is the number of differences required to give a stationary series and $(1 - L)^d$ is the d^{th} power of the differencing operator. When $d \in (-0.5, 0.5)$, the $ARFIMA(p, d, q)$ process is stationary, and if $d \in (0, 0.5)$ the process presents long-memory behavior.

Forecasting $ARFIMA$ processes is usually carried out by using an infinite autoregressive representation of (1), written as $\prod(L) y_t = \varepsilon_t$, or

$$y_t = \sum\nolimits_{i=1}^{\infty} \pi_i y_{t-i} + \varepsilon_t, \tag{9}$$

where $\prod(L) = 1 - \pi_1 L - \pi_2 L^2 - \cdots = \Phi(L)(1 - L)^d \Theta(L)^{-1}$. In terms of practical implementation, this form needs truncation after k lags, but there is no obvious way of doing it. This truncation problem will also be related to the forecast horizon considered in predictions (see [12]). From (9) it is clear that the forecasting rule will pick up the influence of distant lags, thus capturing their persistent influence. However, if a shift in the process occurs, this means that pre-shift lags will also have some weight on the prediction, which may cause some biases for post-shift horizons [8].

3.4 FIGARH Model

The model enabling description of long-memory in variance series is $FIGARCH$ (p, d, q) (Fractionally Integrated $GARCH$) introduced by Baillie et al. [1]. The $FIGARCH(p, d, q)$ model for time series y_t can be written as:

$$y_t = \mu + \varepsilon_t, \quad t = 1, 2, \ldots\Omega, \tag{10}$$

$$\varepsilon_t = z_t \sqrt{h_t}, \quad \varepsilon_t | \Theta_{t-1} \sim N(0, h_t), \tag{11}$$

$$h_t = \alpha_0 + \beta(L)\, h_t + \left[1 - \beta(L) - [1 - \phi(L)](1 - L)^d\right] \varepsilon_t^2, \qquad (12)$$

where z_t is a zero-mean and unit variance process, h_t is a positive time dependent conditional variance defined as $h_t = E\left(\varepsilon_t^2|\Theta_{t-1}\right)$ and Θ_{t-1} is the information set up to time $t-1$.

The $FIGARCH(p, d, q)$ model of the conditional variance can be motivated as $ARFIMA$ model applied to the squared innovations

$$\varphi(L)(1 - L)^d \varepsilon_t^2 = \alpha_0 + (1 - \beta(L))\, \vartheta_t, \quad \vartheta_t = \varepsilon_t^2 - h_t, \qquad (13)$$

where $\varphi(L) = \varphi_1 L - \varphi_2 L^2 - \cdots - \varphi_p L^p$ and $\beta(L) = \beta_1 L + \beta_2 L^2 + \cdots + \beta_q L^q$ and $(1 - \beta(L))$ have all their roots outside the unit circle, L is the lag operator and $0 < d < 1$ is the fractional integration parameter. If $d = 0$, then FIGARCH model is reduced to GARCH; for $d = 1$ though, it becomes IGARCH model. However, FIGARCH model does not always reduce to GARCH model. If GARCH process is stationary in broader sense, then the influence of current variance on its forecasting values decreases to zero in exponential pace. In IGARCH case the current variance has indefinite influence on the forecast of conditional variance. For FIGARCH process the mentioned influence decreases to zero far more slowly than in GARCH process, i.e. according to the hyperbolic function [1, 18]. In practical implementation of prediction FIGARH model see [18].

4 Parameters Estimation and Choice of Model

The aim of searching for a useful forecasting model is not utilization of the greatest number of parameters which will most accurately describe variance of the analyzed time series. It is due to the fact that too big matching may embrace the description not only of the part of the process called signal but also of random noise, for which in finished trials one can discern as random regularity. The objective of the research is rather discovery of such a model which will describe the most important properties of the analyzed time series by means of a finite number of statistically essential parameters [7]. The most often used method of parameter estimation of autoregressive models is the Maximum Likelihood Estimation (MLE). The basic problem appearing while using this method is the necessity to define the whole model and consequently sensitivity of the obtained estimator to the presumptive errors in the specification of polynomials AR and MA, which are responsible for the process dynamics [9]. There is no universal criterion for the choice of the model. Usually, the more complex the model, the bigger is its likelihood function. Therefore, there is a searching for a compromise between the number of parameters occurring in the model and the value of the likelihood function. The choice of a sparing form of the model is performed on the basis of information criteria such as Akaike (AIC) or Schwarz (SIC). In our article, for parameter estimation and choice of the model, we utilized the Maximum

Likelihood Method. It was due to its relative simplicity and computational efficiency. In order to estimate the order of the AR and MA models we used the autocorrelation function ACF and $PACF$. For $ARMA$ model, however, we used Box-Jenkins procedure [2]. For $ARFIMA$ model we applied HR estimator (described in the Haslett and Rafterys work [11]) and automatic models order selection algorithm based on information criteria (see Hyndman and Khandakar [14]). For estimation of $FIGARCH$ model we used the methodology described in the article [18].

5 Experimental Results

Experimental results are based on traffic features set taken from SNORT [17] based preprocessor which we proposed in [16]. We have used 26 traffic features presented in Table 1. For algorithms evaluation we used Kali Linux [15] tools for simulating real

Table 1 Network traffic features used for experiments

f_1	number of TCP pockets
f_2	in TCP pockets
f_3	out TCP pockets
f_4	number of TCP pockets in LAN
f_5	number of UDP datagrams
f_6	in UDP datagrams
f_7	out UDP datagrams
f_8	number of UDP datagrams in LAN
f_9	number of ICMP pockets
f_{10}	out ICMP pockets
f_{11}	in ICMP pockets
f_{12}	number of ICMP pockets in LAN
f_{13}	number of TCP pockets with SYN and ACK flags
f_{14}	out TCP pockets (port 80)
f_{15}	in TCP pockets (port 80)
f_{16}	out UDP datagrams (port 53)
f_{17}	in UDP datagrams (port 53)
f_{18}	out IP traffic [kB/s]
f_{19}	in IP traffic [kB/s]
f_{20}	out TCP traffic (port 80) [kB/s]
f_{21}	in TCP traffic (port 80) [kB/s]
f_{22}	out UDP traffic [kB/s]
f_{23}	in UDP traffic [kB/s]
f_{24}	out UDP traffic (port 53) [kB/s]
f_{25}	in UDP traffic (port 53) [kB/s]
f_{26}	in TCP traffic (port 4444)

world attacks in controlled network environment (for example: Application specific DDos, various port scanning, DoS, DDoS, Syn Flooding, pocket fragmentation, spoofing and others).

For anomaly detection we used statistical algorithms with short and long memory dependence: $ARMA$, $FIGARH$ and $ARFIMA$. In Tables 2 and 3 there are results of DR [%] and FP [%] for mentioned three algorithms. Most promising results in terms of DR and FP were achieved for $ARFIMA$ long memory statistics (with FP less then 10 %).

Table 2 Detection Rate DR [%] for a given network traffic features

Feature	ARMA	FIGARH	ARFIMA
f_{1}	4.24	5.40	6.26
f_{2}	9.22	10.20	12.24
f_{3}	9.22	10.20	12.24
f_{4}	9.22	10.20	12.24
f_{5}	9.22	10.20	12.24
f_{6}	0.00	0.00	0.00
f_{7}	0.00	0.00	0.00
f_{8}	30.52	32.20	35.64
f_{9}	88.68	90.42	96.52
f_{10}	87.23	90.24	95.45
f_{11}	0.00	0.00	0.00
f_{12}	78.82	80.24	82.24
f_{13}	9.22	10.20	12.24
f_{14}	9.22	10.20	12.24
f_{15}	9.22	10.20	12.24
f_{16}	0.00	0.00	0.00
f_{17}	4.42	5.40	6.26
f_{18}	9.22	10.20	12.24
f_{19}	9.22	10.20	12.24
f_{20}	4.42	5.40	6.26
f_{21}	9.22	10.20	12.24
f_{22}	0.00	0.00	0.00
f_{23}	0.00	0.00	0.00
f_{24}	0.00	0.00	0.00
f_{25}	0.00	0.00	0.00
f_{26}	75.24	78.00	80.00

Table 3 False Positive FP [%] for a given network traffic features

Feature	ARMA	FIGARH	ARFIMA
f_1	6.01	5.24	4.22
f_2	5.85	5.45	4.12
f_3	5.92	5.24	4.15
f_4	5.72	5.22	4.11
f_5	5.24	4.28	3.54
f_6	4.42	3.34	2.23
f_7	7.24	6.75	5.98
f_8	6.15	5.24	4.15
f_9	6.85	6.22	5.05
f_{10}	2.53	1.46	0.48
f_{11}	4.24	3.52	2.56
f_{12}	2.18	1.04	0.05
f_{13}	6.21	5.46	4.14
f_{14}	5.12	4.45	3.24
f_{15}	5.08	4.38	3.32
f_{16}	2.11	1.24	0.02
f_{17}	2.25	1.82	0.39
f_{18}	5.12	4.55	3.82
f_{19}	5.14	4.62	3.26
f_{20}	6.22	5.34	4.55
f_{21}	5.32	4.22	3.11
f_{22}	3.17	2.46	1.60
f_{23}	5.77	4.44	3.42
f_{24}	0.00	0.00	0.00
f_{25}	1.12	0.45	0.02
f_{26}	1.12	0.45	0.02

6 Conclusion

Ensuring a sufficient level of safety to resources and information systems is a question that is currently intensively surveyed and developed by many research centers in the world. A growing number of novel attacks, their global reach and level of complexity enforce dynamic development of network safety systems. Most often implemented mechanism aiming to ensure security are methods of detection and classification of abnormal behaviors reflected in the analyzed traffic. In the present article, we compare properties of predicated analyzed statistical models in terms of their effectiveness to detect anomalies in network traffic. The analyzed models were those of a long and short memory reflected in the autocorrelation strength of elements composing a given time series. Parameter estimation and identification of the range of the model were realized as a compromise between the models coherence and size of its estimation error. While realizing implementation processes of the described models there were achieved diverse statistical estimations for the analyzed signals of the network traffic. In order to detect anomalies in the network traffic we used differences between the real network traffic and its estimated model for the analyzed parameters characterizing number of received or sent TCP, UDP and ICMP packages within a time unit. The results obtained after the performed experiments show advantage of predictive models ARFIMA and FIGARCH in the network traffic anomaly detection.

References

1. Baillie, R., Bollerslev, T., Mikkelsen, H.: Fractionally integrated generalized autoregressive conditional heteroskedasticity. J. Econom. **74**, 3–30 (1996)
2. Beran, J.A.: Statistics for Long-Memory Processes. Chapman and Hall, New York (1994)
3. Box, G., Jenkins, G., Reinsel, G.: Time Series Analysis. Holden-day, San Francisco (1970)
4. Box, G.E., Jenkins, M.G.: Time Series Analysis Forecasting and Control, 2nd edn. Holden-Day, San Francisco (1976)
5. Brockwell, P., Davis, R.: Introduction to Time Series and Forecasting. Springer, New York (2002)
6. Chondola, V., Banerjee, A., Kumar, V.: Anomaly detection: a survey. ACM Comput. Surv. **41**(3), 1–72 (2009)
7. Crato, N., Ray, B.K.: Model selection and forecasting for long-range dependent pro-cesses. J. Forecast. **15**, 107–125 (1996)
8. Gabriel, V.J., Martins, L.F.: On the forecasting ability of ARFIMA models when infre-quent breaks occur. Econom. J. **7**, 455–475 (2004)
9. Geweke, J., Porter-Hudak, S.: The estimation and application of long memory time series models. J. Time Ser. Anal. (4), 221–238 (1983)
10. Granger, C.W.J., Joyeux, R.: An introduction to long-memory time series models and fractional differencing. J. Time Ser. Anal. **1**, 15–29 (1980)
11. Haslett, J., Raftery, A.E.: Space-time modelling with long-memory dependence: assessing Ireland's wind power resource (with discussion). Appl. Stat. **38**(1), 1–50 (1989)
12. Hosking, J.R.M.: Fractional differencing. Biometrika **68**, 165–176 (1981)
13. Hurst, H.R.: Long-term storage capacity of reservoirs. Transactions of the American Society of Civil Engineers **1**, 519–543 (1951)
14. Hyndman, R.J., Khandakar, Y.: Automatic time series forecasting: the forecast package for R. J. Stat. Softw. **27**(3), 1–22 (2008)
15. Kali Linux: https://www.kali.org (2015)
16. Saganowski, Ł., Goncerzewicz, M., Andrysiak, T.: Anomaly Detection Preprocessor for SNORT IDS System, Image Processing and Communications Challenges 4. Advances in Intelligent Systems and Computing **184**, 225–232 (2013)
17. SNORT IDS: http://www.snort.org (2014)
18. Tayefi, M., Ramanathan, T.V.: An overview of FIGARCH and related time series models. Aust. J. Stat. **41**(3), 175–196 (2012)