Internet Censorship Based on Bayes Learning Model

Ajeesh Ramanujan and Blesson Andrews Varghese

1 Introduction

The Internet is a global computer network through which people around the globe communicate with each other, share files, learn and entertain themselves. It was originally intended for military and research purposes. There is no doubt that internet has brought together the entire world as a single global village. Internet, as a technology, has the enormous potential to be of benefit to human lives than any other invented technology in the world. However, a lot of people are using it to spread hatred, terrorism [18], and obscenity in the world. Therefore, it is necessary that the internet should be censored. Internet censorship refers to any process by which information that is publicized or viewed on the internet is controlled. Internet censorship in India typically occurs as DNS filtering which is often selective and is not entirely effective. The public often protests against censorship as they consider it as a violation of their freedom of speech. The lack of any common standards aggravates the issue. They support their claim by providing examples of authoritarian regimes which use censorship to suppress opposition parties and minority religious groups. However, censorship is the only tool available in the virtual world which can effectively counter social evils like terrorism, piracy, defamation, and fake news. National security can be assured only through censorship. It also protects each individual's right to be forgotten. Thus there is no denying that the internet should be censored to prevent unfortunate events.

The internet does not have any physical boundaries. Any precious information once posted online cannot be completely deleted due to replication. Restricting the

A. Ramanujan · B. A. Varghese (✉)
CSED, College of Engineering Trivandrum, Thiruvananthapuram, India
e-mail: ajeesh@cet.ac.in

M. Palesi et al. (eds.), *Second International Conference on Networks and Advances in Computational Technologies*, Transactions on Computational Science and Computational Intelligence, https://doi.org/10.1007/978-3-030-49500-8_5

posted information is also difficult as the internet is vast and provides anonymity. In this paper, we are giving an overview of current internet censorship methods in India. We also explain various circumvention tools used by citizens to overcome these sensors and the consequences of using the same. Later, we introduce a censorware having machine learning capabilities. It tries to identify whether a website should be filtered based on its content's degree of harmfulness. To the best of author's knowledge, such a study is not conducted earlier.

2 Internet Censorship

Internet censorship can be applied at national/ISP/institutional/user levels. It targets to block contents harmful/objectionable to authorities. It is implemented using several technologies like firewalls, proxies, and DNS filters. No single solution provides complete coverage. Therefore censoring organizations deploy several technologies together to achieve desired results. Censorship can be done technically as well as non-technically. Some of the most common technical methods are

1. **Internet Protocol (IP) Blocking**
   The most common internet filtering technique used by countries. Most of the censorware keep a regularly updated blacklist of IP addresses which are known to spread malicious/illegal content. All communications send to/received from such IP addresses are completely blocked. Usually, censorware employs a validation tool like VirusTotal to confirm that an IP address is a legitimate one to avoid the overhead of maintaining a blacklist. IP blocking only targets IP-based protocols like HTTP, FTP, and POP. IP blocking is usually circumvented by using a virtual private network (VPN) or finding uncensored proxies. Some websites own multiple IP addresses. In order to censor such websites, we need to block all such IPs. A major shortcoming of IP blocking is that, if the website to be censored is deployed on a shared web-server, all websites on the same server will be blocked. In India, most ISPs use IP blocking to block access to websites [2, 5–8]. All such blocks are not entirely effective as they do not prevent tech-savvy users from accessing such websites [9].
2. **Domain Name System (DNS) Filtering and Redirection**
   There will be a DNS authority in every country. Officials can deregister a domain that has illegal/harmful materials. Whenever a user tries to connect to such websites, DNS will not be resolved or incorrect IP address is returned using DNS hijacking or other means [3]. Users could circumvent DNS filtering easily by accessing foreign search engines and DNS servers.
3. **Uniform Resource Locator (URL) Filtering**
   URL filters verify that hyperlinks and URLs do not contain any malicious commands, keyword, or code [11]. URL filtering mainly targets HTTP based protocols. Attackers use encrypted protocols like VPN and TLS/SSL to circumvent such filters. Using escape characters in the URL will also confuse filters. Nowadays, URL filtering is used by web and email scanning engines to identify harmful emails and search results.

4. **Packet Filtering**
   All communication on the internet happens through packets. Packet filtering uses deep packet inspection to identify any forbidden content in the packet and drop it. Packet filtering targets all TCP-based protocols like HTTP, FTP, and POP. However, if the packet is encrypted, the filter will not identify any forbidden content.
5. **Network Disconnection**
   In highly sensitive situations, rather than trying to censor the network, authorities fully block the network with the help of network disconnection. They disconnect power to all routers or block communication to them. Highly privileged users can still use satellite ISPs to gain access to the internet in such scenarios.
6. **Network Attacks**
   Rather than fully blocking the network, we can target malicious websites and launch attacks like DoS to break it. Thus access to that website is prevented for a limited period.
7. **Search Result Removal**
   Government and legal authorities may force a major web portal or search engine to block a malicious website. Thus malicious website is excluded from their search results. As a result, the site is invisible to people who do not know where to find it. When a major website like Google does this, it has the same impact as censorship.

Internet content, like any other media, can be censored using nontechnical censorship methods like

1. **Legal Prosecution**
   Based on complaints received, Court will pass laws prohibiting various types of content and/or order the removal of content [13, 14, 16]. All publishers, authors, and ISPs are liable to remove, alter, or block access to those specific content. They may defend the judgment by going for an appeal and obtaining stay orders [10].
2. **Detention**
   Those publishers, authors, and ISPs who fail to remove illegal content will be arrested [12]. Later they may be punished with fines and imprisonment. As an author, he/she may be banned from further publishing for some duration. Businesses may be closed down by revoking their licenses.
3. **Blackmail and Other Criminal Practices**
   Publishers, authors, and ISPs who publish uncensored content may be threatened and attacked by people who were affected by this content. This may even lead to murder. People may employ hackers who will threaten ISPs and local authorities on behalf of them to work according to their interests.
4. **Bribes, Promotion, and Other Forms of Payment**
   Individuals/websites may be given incentives for supporting certain claims and viewpoints. They will be promoting articles and comments in support of one group or attacking opposition groups without notifying the readers.

5. **Controlling Network Access**
   Some social networking sites have mandated verifying phone numbers while registering. This has reduced anonymous attacks in social networks to some extent [15].

Internet censorship should be done in such a way that the internet remains a great source of reliable information. At the same time, we should protect those vulnerable to internet exploitation. As nobody has complete control over the internet, it is very difficult to punish a person for internet crimes like defamation, copyright infringement, and hate crimes. The uncensored internet can negatively affect the lives of several people. The censorship of internet can protect people from malware, ransomware received via internet making their internet life more safe and simple. Internet censorship prevents inappropriate information flow and ensures that critical information do not reach the wrong people. Internet censorship helps in preventing a large number of financial frauds and identity thefts

In recent years, internet bullying and violence has become a major concern [17]. Users can be anonymous on the internet and information spreads rapidly over the internet. Some users take advantages of such properties of the internet to create violence. The users may abuse, defame each other, and expose others privacy bringing great harm to them. Many celebrities are victims of such internet violence. There are incidents of internet users being cheated by other users through social networking sites. The occurrence of all these incidents and similar incidents make the internet censorship absolutely necessary and demanding.

Censoring the internet is not a simple process. Often, censorware suffers from several drawbacks like

1. **Overblocking/Over-Censoring**
   Overblocking refers to a scenario where legitimate content is getting blocked by censorware. For example, some health-related information may be censored unintentionally believing it to be porn material. Sometimes authorities prefer overblocking rather than risking allowing access to undesirable sites.
2. **Underblocking/Under-Censoring**
   Underblocking refers to a scenario where content that needs to be censored according to censorship policy is not censored properly using censorware. It happens when censorware fails to identify the content as undesirable. Whenever a new category of malicious information is uploaded to the internet, censorware will not censor the content unless updated quickly and accurately.
3. **Violation of Constitution**
   If any government try to censor a particular moral or political issue without valid reasons, it is considered as a violation of democracy and will be disapproved. Without adequate governmental supervision/permission, no censorware should be ideally deployed in a public network. Any form of internet censorship taking place should be informed to visitors using error 451.
4. **Legal Necessities**
   Internet censorship faces various legal actions in several countries. Censorship doesn’t face many legal actions in aristocratic regimes like North Korea and

China whereas a large number of cases are filed against and file for censorship in democratic countries like India. In order to not face any legal actions, all censorware developers should ensure that their censorware does not suffer from overblocking and underblocking before mass distribution. They should document properly what all software standards were followed while development and testing. They should also mention any limitations identified in documentation.

Internet censorship circumvention refers to various processes of bypassing internet censorware and gaining access to censored materials. Usually, the common people lack expertise and knowledge to circumvent censorship but for most of the technologically savvy users, circumventing internet censorship is just a piece of cake. Circumventing works because censorship does not necessarily remove content from the internet but just makes it difficult to access it. Whenever a new blocking technology is introduced, anti-censorware developers reverse engineer it and find a new circumvention technology which can bypass it [4]. Different tools and strategies are used for internet censorship circumvention, including

1. **Cached Web Pages**
   Search engines like Google keep snapshots of web pages from an earlier point of time. Cached pages are identical to the original page in most cases. Even if the original website is blocked, cached web pages may still be accessible. The advantage of this technique is that no additional software needs to be installed
2. **Mirror Sites**
   Mirror websites or mirrors are replicas of other websites. Therefore even if the original website is blocked, copies of the website are still present at mirror sites which are not blocked. Using such sites, blocked content can still be accessed.
3. **Web to Email Services**
   Web to email services will return the contents of web pages with or without images as an email message. The content of a blocked web page can be accessed as an email using this service.
4. **Feed Aggregators**
   A feed is a Web document that is a shortened version of a Web page. Feed aggregator or RSS aggregator collects feeds from different web pages and shows it in a desktop window or web browser. Using such aggregators, blocked content can be retrieved directly.
5. **Direct IP Addresses**
   Several sites may own multiple domains. Only a few such domains or URLs may be blocked. Others will still be available. Trying to access an IP address directly will sometimes allow access to a blocked site. Some censorware can be fooled by entering the IP address in a base other than 10.
6. **Alternative DNS Servers**
   DNS server contains a database of public IP addresses and their associated hostnames. It helps in translating domain names to IP addresses as requested [1]. DNS servers are usually owned by ISPs and other private business organizations. Using DNS servers other than those supplied by default by an ISP may bypass DNS-based blocking.

7. **Proxy Websites**
   Proxy websites are the fastest way to circumvent censorship. They act as an intermediary between the user and the blocked website. User visits the proxy website and requests access to a blocked website by submitting the URL of the blocked website and initiating a connection. The proxy website will fetch the requested content and displays it.
8. **Reverse Proxy**
   A website may have several web servers behind a proxy. A reverse proxy server takes client requests from the internet and forwards it to one or more servers. These resources are then returned to the client as if they originated from a single server. Websites can avoid censorship by rerouting traffic using reverse proxies. Reverse proxies can also protect original characteristics and existence of actual web servers thereby making censorship difficult.
9. **Virtual Private Networks (VPNs)**
   Using VPN, censored users can create a secure connection to a country with relaxed censorship rules. Once connection is established, they can browse the internet as if they are in that country. Thus all blocked content can be accessed easily and safely.
10. **SSH Tunneling**
    SSH tunneling created an encrypted SSH connection. Users can transport all their traffic through this connection. Thus, both outgoing requests for blocked sites and the response from those sites are hidden from the censors.
11. **Sneakernets**
    A sneakernet refers to transferring information from one place to another by physically carrying electronic data on a storage media. Since we are not using any computer networks for transfer, no censorship is applicable to such transfer.
12. **Hybrid Combinations**
    Circumvention methods mentioned above can be combined to form hybrid methods which are more effective against censorship. For example, we can combine alternate DNS server technology together with VPN to create a smart DNS proxy server.

The above circumvention techniques differ in ease of use, speed, security, and risks. They target to achieve an uncensored internet connection. Rather than using the above techniques, using alternate protocols like FTP, telnet, or HTTPS will bypass some censorware. Some censorware can be fooled by conducting searches in a different language. Some countries have strict laws against circumvention. Yet, people are using several nonsecure ads based circumvention software. Internet censorship transparency is necessary to avoid confusions and negative attitude towards internet censorship. Only very few countries in the world openly admit that they practice internet censorship. Most of them would not even disclose censorship techniques employed, list of blocked websites, etc. leading to public protests against censorship.

The sensors may target nodes, users, or links. They may employ multiple strategies to filter malicious content. All censorware in the market should enforce censors at all scenarios without affecting performance. They must be scalable and cheap. They should provide accurate results with minimum false positives and false negatives. Every censorware should be capable of adapting against new circumvention techniques.

3 New Censorware Proposed

We are proposing a new censorware, Ever Learning Censorware, which learns continuously based on Naive Bayes learning technique. After every learning, it shall identify and filter harmful websites in a better way. The major components of the proposed censorware are

1. blacklist—A blacklist of domains or keywords and their degree of harm
2. classifier—A Bayesian classifier which will classify domains, keywords into harmful, moderate, and harmless
3. packet capture engine—It captures all network traffic. Whenever a site is accessed, it captures such packets, extracts domain/URL, and gives to a classifier for classification. If the packet is identified as harmful, communication is dropped.

In addition, we are storing recent harmful sites in a recent sites list to improve performance. It is periodically updated to remove entries older than 2 days. It will contain only one entry for each site with last accessed time. If a site is present in recent sites list as well as blacklist with a high degree of harm, it is simply returned as harmful.

The attacking mode chosen is to attack the harmful link in the network rather than targeting a particular node/user. The filtering approach used is as follows:

1. Drop all communications which cannot be analyzed at all.
2. IP Filtering—Drop packets send to and received from the website found on the blacklist with a high degree of harm.
3. Filtering based on classifier- Use classifier to find the nature of new domains not present in the blacklist. Add domain and degree of harm to the blacklist. If the domain is harmful, drop the communication. If moderate, do keyword filtering. If harmless, allow communication.
4. Keyword Filtering—Search for blacklisted keywords and blacklisted links in packet content. If the number of such keywords/links reaches a threshold (say 400), block communication, add/update the entry in blacklist with a high degree of harm. If all keywords have a low probability of being harmful (below a threshold), allow communication.
5. DNS Hijacking—In case of dropping communication, redirect the user to a block page confirming that content is being censored and asking for any suggestions. These suggestions can be stored in a central server and can be reviewed periodically.

Rather than completely blocking content which cannot be analyzed, we can log those domains and allow communication after getting user consent confirming communication is legal. An admin can periodically review non-analyzable sites and take remedial measures. In order to measure, the degree of harm of a site, a new censoring approach is identified based on Bayes theorem.

Particular words have particular probabilities of being harmful and getting censored. For example, keywords like $porn, drug, suicide, murder, book, pencil, movie, film, music$ have respective probabilities (degree of harm) of 1, 0.9, 0.95, 0.93, 0, 0, 0.6, 0.7, 0.4. The filter will not know these probabilities without training. For manual training, the user must manually indicate a word and its degree of harm. If a new word is encountered after training, it is assigned a random degree of harm (say 0.4). This can be reviewed by a moderator initially. The degree of harm associated with every word will be continuously updated once the number of websites containing that word increases a threshold. It is calculated by dividing the total number of sites containing that word with the total number of sites inspected. After a significant amount of testing, the degree of harm associated with each word is expected to not deviate much. The probabilities associated with each word found on the website are used to calculate the probability that a website belongs to which category. Each word in a website (including domain name) contributes to the probability that the website is harmful. The website's probability of harm is computed and if it is greater than some threshold values (say 0.9, 0.5), websites are classified as harmful and moderate. Otherwise, it is classified as harmless

Let us assume that a website contains word "videos." It may be a benign website or a malicious website. Internet censorware will try to identify whether a website is harmful from this particular word. For that it uses the formula based on Bayes theorem

$$P_r(A|B) = \frac{P_r(B|A) \cdot P_r(A)}{P_r(B|A) \cdot P_r(A) + P_r(B|\neg A) \cdot P_r(\neg A)}$$

where

- $P_r(A|B)$ is the probability that accessed website is harmful knowing that it contains this keyword
- $P_r(B|A)$ is the probability of occurrence of this word in harmful websites. It is same as the degree of harm of word calculated by censorware
- $P_r(A)$ is the marginal probability that a website is harmful. It is calculated by dividing the total number of sites identified as harmful with the total number of inspected sites.
- $P_r(B|\neg A)$ is the probability of occurrence of word W in harmless websites.
- $P_r(\neg A)$ is the marginal probability that a website is harmless. It is calculated by dividing the total number of sites identified as harmless with the total number of inspected sites.

If we determine the harmness of a website only based on the presence of a single word, it is error-prone. We need to consider several words and combine their harm to determine a website's overall degree of harm. Combining Individual probabilities, we will get the following formula for Computing the probability that a website is harmful

$$P = \frac{P_1 \cdot P_2 \cdots P_N}{(P_1 \cdot P_2 \cdots P_N) + ((1 - P_1) \cdot (1 - P_2) \cdots (1 - P_N))}$$

where

- P is the probability that a suspected website is harmful.
- $P_1, P_2 \ldots P_N$ are the probabilities that a website is harmful knowing it contains words $w_1, w_2 \ldots w_N$
- N is the total number of valid words on the website

Rather than assigning a random degree of harm to new words not present in the blacklist, the classifier can also decide to discard such words for which there is no information available. Words like the, a, $some$, is, etc. for which degree of harm cannot be defined are ignored. It is obvious that we must not assign a degree of harm to numeric data, special symbols, and spaces. Even if we ignore such harmless components, there would not be much impact. We keep a list of such words to filter them. Even if censorware automatically adds a new such word to blacklist, it is removed later during periodic moderation. We can also try grouping words rather than a single word. Thus accuracy can be improved.

A background service should run continuously and stores the latest content of critical components like the blacklist, recent sites list, non-analyzable list to stable storage. This ensures that all the learned information are safe. Rather than trying to update all entries, we can recreate lists. Thus even if the censorware is terminated due to unforeseen consequences, all learnings are not lost. Censorware can be restarted and used based on these lists in stable storage. In such scenarios, we will only lose learnings that could have been conducted in a short interval from the last update time. Keeping backups of all these stable storage components is recommended to avoid data loss in case censorware is terminated while modifying the content in stable storage.

3.1 Implementation

The language used is Java 8. Java was used because of its support for network analysis and machine learning. Java applications can be modified, rewritten, or enhanced easily and can be run on almost all operating systems. An open-source Java library jnetpcap-1.4.r1425 was used to capture HTTP packets and get URL of visited websites. Jnetpcap is a java wrapper for popular libpcap and WinPcap libraries. In order to read the contents of a website, HtmlUnit was used. HtmlUnit is a headless web browser written in Java. It can simulate browsers like chrome and can extract data from websites. All lists (blacklist, recent sites list, non-analyzable list) were stored as simple Unicode files inside the project present in the file system. The entire project was built using Maven 3.5.3 to create our java app.

Implementation Modules

1. **Driver Engine**
   This module contains the main method. It initially calls init method to populate various components like the blacklist, recent sites list, etc. If required files are not available for populating the lists, initial training is carried out. Then Driver engine continuously spawns various threads for packet capturing, classification, automated component update, etc. after confirming they are not alive.
2. **Packet Capture Engine**
   It first gets a list of network devices on the running system. Second, it opens up the selected network device. Third, we create a packet handler which will receive packets from the libpcap loop. Fourth, we enter the loop and tell it to capture 10 packets. The loop method does a mapping of pcap.datalink() DLT value to JProtocol ID, which is needed by JScanner. The scanner scans the packet buffer and decodes the headers. The mapping is done automatically. If the header is of type HTTP, the URL of the accessed website is extracted. After confirming it is not yet tested recently, it is passed to the CheckDomain method of Bayes Classifier for analysis
3. **Bayes Classifier**
   The init method is located here. It also maintains array lists corresponding to blacklist, recent sites list, non-analyzable sites list, etc. It is implemented as a thread which periodically writes these array lists to their corresponding files (blocked, recent, non-analyzable) in the file system. When a site URL is passed to the CheckDomain method of Bayes Classifier, it tries to read the content of webpage using WebScraper. If the content is read, it applies Bayes classification method defined earlier to check the degree of harm of a website. If the website is harmful, it is added to the hosts file to block its further access. If the website is found to be not harmful, hosts file is updated to remove entries corresponding to this website. The website is added/updated in blacklist with a calculated probability
4. **Web Scraper**
   It accepts a URL and gets the content of corresponding web page using htmlunit library. The browser version was given as best_supported. If a website cannot be read, it returns 403 forbidden errors. Such websites are added to nan-analyzable lists.
5. **Word**
   It indicates an entry in the blacklist. It includes corresponding word/domain and its degree of harm.

6. **Site**
   It indicates an entry in the recent sites list. It includes site URL, last accessed time, and degree of harm.
7. **Manual Training Engine**
   It allows Admin to manually train the system. Admin can add a new entry to blacklist, search for any entry and can remove an entry from the blacklist. Admin can also write entire blacklist to blacklist.txt file.

3.2 Results

The initial training was carried out manually by populating blacklist with harmful, benign websites, words, and their degree of harm. Then Network data was sniffed continuously using packet capture engine module. If the system has multiple network devices, packets coming from any network device are captured. Whenever user tries to access a website, corresponding HTTP packets are captured successfully. The URL was extracted from those HTTP packets and was sent to analyze. As explained in earlier section, Bayes classifier analyzes degree of harmness of website. If the web page is analyzed and identified as not harmful, communication is allowed. Otherwise, communication is dropped and the domain name is added to the hosts file to prevent further access. In both cases, the blacklist is updated with non-redundant pairs of (words, the degree of harm) corresponding to website content and domain. When the user tries to access this site later, he would not be able to establish a connection. The degree of harm of each website is reevaluated periodically to avoid unnecessary long-term blocking errors. Thus censorware is successfully blocking websites identified as harmful. The success of censorware entirely depends on the blacklist and initial training should be carried out extensively to cover all domain types. As the training continues, size of blacklist is increasing exponentially resulting in performance degradation. As part of future research, blacklist may be changed to an indexed database to improve performance.

4 Conclusion

Internet censorship is really necessary for today’s society. Since the internet is growing on a daily basis and has a wide range of applications, misuse of the internet can lead to drastic consequences. The censorware proposed in this project can overcome many limitations of the existing system and is far more efficient. It is also much transparent compared to existing leading to greater public support. The training can become a little cumbersome, but it can be managed. Manual intervention is needed only at the beginning. This design can be extended to implement similar censorware in routers and other internet endpoints. We can improve the proposed censorware by introducing a mechanism to analyze the https packet. One method suggested is reassembling TCP packets and analyzing the assembled packet. Another area which can be improved is the blocking policy. We can think of an intermediate DNS server or a Firewall rather than adding to the hosts file. As part of result analysis, we have identified that the size of the blacklist is increasing tremendously with each website tested. We can try to restrict size by grouping similar words, irrelevant words, etc.

References

1. Sarkar, P.K., Jain, A.: Intelligent Transport System. PHI Learning, 15 Nov 2017
2. Orlowski, A.: India Blocks Yahoo! Groups. The Register, 24 September 2003
3. Montieri, A., Pescape, A., Aceto, G.: Internet censorship in Italy: an analysis of 3G/4G networks. In: IEEE ICC 2017 Communications QoS, Reliability, and Modeling Symposium (2017)
4. Leberknight, C.S., Wong, F., Poor, H.V., Chiang, M.: A taxonomy of internet censorship and anti-censorship. In: Fifth International Conference on Fun with Algorithms (2010)
5. Raj, D.: BuyDomains.com Blocked in India for no Obvious Reason. TechBlogger (2012)
6. Government to block all porn sites in India, asks Internet providers to deny access to such websites. Mobiletor, 11 November 2014
7. India blocks 32 websites, including Vimeo and Github. India Today, 31 December 2014
8. India blocks Pakistani newspaper web site. Newswatch.in, 5 July 1999
9. Anwer, J.: Blocking Website in India: Reliance Communications Shows It Is Very Easy. Times of India, 24 December 2011
10. Pereira, L.: Singham Effect: File Sharing Sites Blocked. NDTV, 22 July 2011
11. Gupta, M., Verkamp, J.-P.: Inferring Mechanics of Web Censorship Around the World. FOCI '12, 2012
12. Pahwa, N., et al.: Updated: Indian Government Blocks Typepad, Mobango, and Clickatell. MediaNama, 4 March 2011
13. Pahwa, N., et al.: No more John Doe orders? Indian ISPs get court order for specificity in URL blocks. MediaNama, 20 June 2012
14. Pahwa, N., et al.: 219 Websites Blocked in India, after Sony Complaint. MediaNama, 7 July 2014
15. Orkut's tell-all pact with cops. The Economic Times, 1 May 2007
16. Ribeiro, J.: Delhi Court issues summons to Google, Facebook headquarters for objectionable content. PC World, 16 January 2012
17. Deibert, R.J., Palfrey, J.G., Rohozinski, R., Zittrain, J. (eds.): ONI Country Profile: India, Access Contested. OpenNet Initiative, MIT Press, November 2011, pp. 299–308
18. Sengupta, S.: India Blocks Blogs in Wake of Mumbai Bombings. The New York Times, 18 July 2006