Investigating large-scale Internet content filtering

Sebastian Wolfgarten, sebastian@wolfgarten.com

August 2006

M.Sc. in Security & Forensic Computing 2005/2006, Dublin City University, Ireland

Abstract

This paper analyzes large-scale, countrywide Internet content filtering from a technical point of view and investigates the current situation in the People's Republic of China. Additionally it discusses techniques to effectively defeat censorship and based on various tests conducted by the author, comments on their applicability in the Chinese part of the Internet.

1. Introduction

Nowadays the Internet has become an essential element of the world's media landscape and our everyday lives. Thus for many people sending and receiving emails, chatting with friends, researching information or even purchasing goods is almost as common as watching TV or listening to the radio. Interestingly without being further challenged it is generally taken for granted in the Western world that based on human rights, constitutions, legal systems and moral values, access to the Internet is provided freely, unlimited and most importantly unfiltered. But in reality the situation for millions of users world-wide is completely different: "Chat rooms monitored. Blogs deleted. Websites blocked. Search engines restricted. People imprisoned for simply posting and sharing information" [1]. In an attempt to create virtual frontiers in cyberspace countries such as China, Vietnam, Tunisia, Iran, Saudi Arabia and Syria [1] have installed a multiplicity of technical and non-technical controls to censor the Internet and prevent their citizens from accessing or publishing information the government regards as illegal. Therewith these countries are denying essential human rights to their citizens and specifically violate article 19 of the Universal Declaration of Human Rights which states that "everyone has the right to freedom of opinion and expression; this right includes freedom to hold opinions without interference and to seek, receive and impart information and ideas through any media and regardless of frontiers" [2]. In order to gain a further understanding of the functionality and the extent of such censorship, this paper investigates large-scale, countrywide Internet content filtering from a technical point of view. Therefore at first it discusses various means of filtering a government might enforce to perform censoring. Next it investigates the current situation of Internet filtering in the People's Republic of China and presents the implications for Chinese users by providing concrete examples. Finally this paper particularly highlights techniques to circumvent Internet censorship focusing on practical and easy to use solutions that are applicable in China.

2. Filtering techniques

2.1. Introduction

First it must be noted that the rapid advancement of technology and especially the common availability of high bandwidth Internet connections, as well as the dynamic nature of Internet content, pose a severe technological challenge to anyone trying to countrywide filter and control access to certain pieces of information. However if one was to consider a country in which, due to a lack of legal and moral restrictions, the government would have unrestricted access and full control over central network infrastructure elements, the state would be able to observe, analyze and possibly alter all data that is being transferred unencrypted. Given a tremendous amount of money, technology, knowledge and human labour, the government would then be able to perform large-scale Internet content filtering and subsequently control and prevent its citizens from accessing, exchanging or publishing information it regards as illegal. In order to study the practical implications of such large-scale Internet content filtering, Dornseif [3] and Clayton [4] analyzed the blocking of Nazi and pedophile websites in Germany and England. They identified a variety of techniques operating at different levels of the Open Systems Interconnection Reference Model (OSI model) that all vary in terms of their costs, implementation, granularity and effectiveness. Therefore the next section introduces the filtering techniques available at the different levels.

2.2. Network-level filtering

Network-level or packet filtering operates on layer 3 and 4 of the OSI model. Each packet is inspected in real-time as it passes through the filtering device (e.g. router) and based on the content of its header the device either forwards or silently discards the packet. This type of filtering is well-known for a long period of time and is implemented in a majority of the devices available today.

Hence in theory there is no need for providers to purchase additional hardware or introduce new technology to perform this type of filtering. According to Dornseif, the downside of network-level filtering is that there is no way of informing the user about the filtering simply because the filtering device silently drops the user's packets. This also holds true for unintentionally blocked information, a phenomenon often referred to as "overblocking" in which content or services are "not intended to be blocked but actually are blocked because of the coarse granularity in IP-filtering" [3]. Dornseif also differentiates between two different types of network-level filtering techniques (layer 3 and layer 4 filtering) that both vary in terms of their "granularity and resource consumption on the filtering device" [3].

2.2.1. Layer 3

The network layer (layer 3) of the OSI model is primarily responsible for logical addressing and routing of data [5] and contains information (e.g. IP addresses) about the source and the destination of a packet. By using this information, one can define rules to block certain packets based on the embedded source and/or destination address and thus prevent any communication to or from a given host. The following sample rules are typical Access Control Lists (ACLs) for Cisco devices [6] that will deny all TCP and UDP traffic to or from the IP address (212.58.224.81) associated with the website of the BBC (www.bbc.co.uk):

```
deny ip host 212.58.224.81 any
deny ip any host 212.58.224.81
```

If these rules are added to a central networking device, there would be no way of accessing the website of the BBC unless the filtering is somehow circumvented. This holds also true for any other service listening on that particular host. The advantage of layer 3 filtering is that, in theory, processing such rules requires only minimal resources on any networking device [3] and can be done very efficiently. However, in practice, given the vitality of IP addresses and websites these rulesets often tend to become very large in size and cause a huge performance loss. Additionally managing, distributing and synchronizing them among all network devices involved is another difficult challenge for the operator of the network infrastructure. Lastly due to the lack of granularity in the filtering mechanism itself, layer 3 filtering does not provide a way of limiting the blocking to a specific service or port. Consequently the filtering might be too broad and may unintentionally block access to a particular host or service (overblocking).

2.2.2. Layer 4

Layer 4 (transport layer) is "primarily responsible for the formatting and handling of the transport of data in a transparent manner" [5]. It provides "reliable and accurate delivery of the data to the next layer" [5] and uses protocols such as TCP, UDP as well as ICMP. The TCP and UDP protocols both include information (i.e. a port number) about the type of service (e.g. port 80 for HTTP) a packet was most likely generated by or is destined for. Together with the source and destination addresses of a packet, this application-specific information provides a finer distinction and division of network traffic when compared with layer 3. An example of layer 4 filtering in Cisco-syntax [6] would be:

```
deny tcp any host 213.133.109.150 eq 25
```

In this example, traffic from any host with any source port to destination port 25 (SMTP) on 213.133.109.150 is denied. If such a rule is deployed, any host affected by this filtering would be unable to communicate with host 213.133.109.150 on port 25 (i.e. send an email to that host). Although layer 4 filtering offers greater flexibility and precision in terms of the scope of the filtering, it may also block access to resources it should not block (overblocking). For instance Dornseif [3] mentions the HTTP protocol in which one server with a single IP address may host several (up to hundreds or thousands) other websites (so-called "name virtual hosting"). Hence if access to the web server is blocked, then access to all other websites that are hosted on the same server is also blocked.

2.3. Application-level filtering

Unlike network-level filtering, application-level filtering is applied at layer 7 (application layer) of the OSI model. Therefore it is possible to inspect and analyze the payload or content of a packet and hence "perform the most detailed inspection on data before making a filtering decision" [5]. This allows the filtering to be applied at the protocol rather than at the network level and hence provides a greater granularity in terms of the filtering. Additionally unlike network-level filtering, application-level filtering often provides ways of informing the user about the filtering. However as each packet has to be inspected, analyzed and possibly executed or sometimes even reassembled, application-level filtering cannot be done in real-time and especially in high-bandwidth environments requires an enormous amount of highly expensive technical equipment in order to remain practicable. Furthermore if an appropriately encrypted protocol such as Secure Socket Layer (SSL) or Secure Shell (SSH) is used, application-level filtering becomes mostly impossible as the payload of the transferred network traffic is encrypted and thus cannot be inspected anymore.

2.3.1. Proxies

Application proxy firewalls (often simply referred to as "proxies" or "proxy servers") operate at the application

layer of the OSI model and "act as intermediary by literally intercepting and responding to requests between hosts" [5]. Therefore they operate by interposing themselves "in the middle of the application protocol and interpreting it while applying security controls to the application commands and data, where appropriate" [7]. An application-level proxy "as part of its normal operation executes the protocol. Instead of having to follow along and try to figure out what the application protocol is doing, the proxy is the application protocol: protocol anomalies represent error conditions that the proxy detects" [7]. Consequently an application proxy firewall allows for the inspection and classification of network traffic into allowed and disallowed (or malicious/non-malicious) data and provides filtering mechanisms based on this classification. As such, an application proxy firewall is, for instance, able to differentiate between "normal" HTTP traffic and HTTP traffic generated by a network worm such as CodeRed or Nimda and apply the filtering accordingly. Unfortunately this flexibility leads to "higher hardware requirements (generally needing faster processors and more memory) as well as higher development costs" [3] and causes a huge negative performance impact as application proxies "spend more time processing the packet, which results in increased latency in the delivery of data" [3] when compared with the aforementioned network-level filtering. Furthermore, in order to be most effective, an application-level proxy needs to fully understand each protocol it is to decode and analyze. Therefore it requires filters for each protocol it needs to analyze but "most proxies can handle only a relatively small number of applications. This limitation means that the other applications are not permitted, or that you have to use a generic service proxy (which may not provide the required functionality), or that the proxy handles the additional traffic as a packet-filtering firewall (making the firewall a hybrid application proxy firewall)" [5].

2.3.2. Deep Packet Inspection

Another technique to perform content filtering at the application level is to use "deep packet inspection". Deep packet inspection refers to "the capabilities of a firewall or an Intrusion Detection System (IDS) to look within the application payload of a packet or traffic stream and make decisions on the significance of that data based on the content of that data" [7]. Therefore in order to apply on-the-fly filtering "deep packet inspection typically includes a combination of signature-matching technology along with heuristic analysis" [5]. However unlike application-level proxies the actual protocol is never executed in deep packet inspection. Initially used as a technology to detect and defend against known and unknown network-based attacks, deep packet inspection is also a suitable technique for performing content filtering if an appropriate set of signatures and keywords is employed.

2.3.3. DNS manipulations

The Domain Name System (DNS) is a globally deployed hierarchical database to resolve hostnames (e.g. www.dcu.ie) into the corresponding IP addresses (e.g. 136.206.1.2). Although it was never intended to be used as a filtering mechanism, it nowadays "seems to be the preferred way of blocking" [3] due to the simplicity and yet effectiveness in which manipulations can be done. A popular example of state-decreed DNS manipulations was a blocking order published in February 2002 by the district government of North Rhine-Westphalia in Germany which forced 78 providers to block access to two Nazi-related websites hosted in the United States. Dornseif was the first to study this order in 2003 and identified six techniques for performing DNS tampering [3]:

- Refused: The easiest way to stop users from connecting to a certain host is to simply refuse to resolve that given domain. Therefore the DNS standard defines the reply "REFUSED" which means that "the name server refuses to perform the specified operation for policy reasons" [3]. Consequently this is likely to cause a "host not found" or "connection refused" error message.

- Nxdomain: A manipulation in which the existence of a particular domain is denied ("NXDOMAIN, non-existing domain") by the recursive DNS server of the provider. To invalidate a domain, the provider has to pretend to be authoritative for that domain and hence breach the DNS standard. For the user this forgery will also cause a "host not found" error message and will prevent the user from connecting to the target host.

- Name hijacking: Refers to a deliberate modification in which the user's request to resolve a certain domain is answered with bogus data. This will typically result in the user being unintentionally redirected ("hijacked") to another site.

- Name invalidation: A technique similar to "name hijacking" in which resolving a domain results in invalid rather than bogus replies. This will cause a "could not connect" error message. Dornseif refers to this method as "name astrayment".

- Silence: Another way of refusing to resolve a particular domain is silently not to respond to such a request at all. This will result in a delay or even a timeout and will eventually cause a "host not found" error.

- Provoked server failures: This type of tampering will cause a server-generated error message to be send to any client trying to resolve a certain domain. Hence the user will experience some sort of

error message (e.g. "could not connect") and will be unable to resolve or connect to the destination domain.

Although trivial to circumvent, these six techniques will typically prevent non-experienced users from resolving a particular domain and thus from connecting to a target host. Interestingly at the time of writing the aforementioned blocking order is still active and will prevent people using a provider based in North Rhine-Westphalia, Germany from successfully resolving a small number of domains. For example, trying to resolve the Nazi-related website www.stormfront.org with the local provider ISIS Multimedia Net will cause name invalidation as described above:

```
# host www.stormfront.org dns1.isis.de
Using domain server:
Name: dns1.isis.de
Address: 145.253.2.80#53
Aliases:

www.stormfront.org has address 127.0.0.1
```

Here the name server of the local provider resolves the domain www.stormfront.org to the IP address 127.0.0.1 (localhost). This is a generic address and refers to each computer's own loopback network interface. Hence a connection attempt to this host will fail with a "could not connect" error message.

3. Filtering by example: China

3.1. Introduction

According to official statistics published by the Chinese government [8], the number of Internet users in China has grown dramatically over the last couple of years and reached an estimated size of approximately 123 million in June 2006. Therewith the communist country has got the world's second largest number of Internet users and is only outbalanced by the United States of America. Although the degree of reliance of such statistics is questionable, these figures are still very impressive given they are equivalent to just 9-10% of China's entire population and subject to growth of about 18-20% per year. Therefore from a technical and non-technical point of view it is incredible to believe that the Chinese government operates "the most extensive, technologically sophisticated, and broad-reaching system of Internet filtering in the world" [9] and effectively manages to prevent Chinese users from accessing or publishing information it regards as bad, critical, subversive or illegal. Forbidden and thus blocked information are including but not limited to [10]

- Any information contradicting the constitution of the People's Republic of China.
- Any information disclosing state secrets, violating national security, subverting the government or destroying the unity of the country.
- Any information damaging the honour and the interests of the state.
- Any information disturbing social order or undermining social stability.
- Any information spreading or instigating lewdness, pornography, gambling, violence, murder or terror.

This leads to a situation in which for instance "Chinese citizens seeking access to Web sites containing content related to Taiwanese and Tibetan independence, Falun Gong, the Dalai Lama, the Tiananmen Square incident, opposition political parties, or a variety of anti-Communist movements will frequently find themselves blocked" [9]. Resistance against this filtering or publishing forbidden information is severely punished by the Chinese government and according to Reporters Without Frontiers [11] has already led to the imprisonment of at least 50 Chinese individuals since 1999. For example the cyber-dissident Li Jianping (40) is currently on trial on a charge of "inciting the subversion of state sovereignty" by publishing critical articles and comments on foreign websites. If convicted, he will have to face an imprisonment of 15-years [12] in China. Technically the censorship is believed to be mainly build on technology provided by well-known Western companies such as Cisco or Juniper Networks [13] and "comprises multiple levels of legal regulation and technical control. It involves numerous state agencies and thousands of public and private personnel" [9]. Interestingly "unlike the filtering systems in many other countries, China's filtering regime appears to be carried out at various control points and also to be dynamic, changing along a variety of axes over time. This combination of factors leads to a great deal of supposition as to how and why China filters the Internet. These complexities also make it very difficult to render a clear and accurate picture of Internet filtering in China at any given moment" [9]. Thus in order to study the implications and extent of this filtering in a real-world environment, the author rented a dedicated server from a Chinese hosting company for a period of four weeks in July/August 2006. This Linux-based server was located in Shanghai where it was directly connected to the backbone of China Telecom, the largest Internet Service Provider (ISP) in China. The next sections provide a brief snap-shot of the implications of Internet filtering in China as experienced by the author.

3.2. The Domain Name System

As mentioned before, manipulations of the Domain Name System (DNS) are probably one of the most easiest, fastest and yet effective methods to prevent users

from accessing particular websites. Thus in order to discover whether this type of manipulation is used by the Chinese government at all, 50 sample domains were resolved simultaneously in an automated manner on both a Chinese and a German server. The results of these tests were then compared with each other to discover any discrepancies and possible manipulations. The sample dataset contained the addresses of well-known political or religious organizations as well as television channels, newspapers and other popular domains which have been reported to be blocked in China [9] or which are very likely to be blocked due to their content. For instance among the tested domain names were:

Domain	Description	Result
www.falundafa.org	Spiritual movement	SERVFAIL
www.amnesty.org	Human rights org.	SERVFAIL
www.bbc.co.uk	Television channel	SERVFAIL
www.wikipedia.org	Online encyclopedia	SERVFAIL
www.cnn.com	Television channel	SERVFAIL
www.greenpeace.org	Non-profit organis.	SERVFAIL
www.gov.tw	Taiwanese governm.	Timeout
www.worldpress.org	News	Timeout

Although all domains were successfully resolved by the German server, it was found that more than 20% of the tested domains either caused a "SERVFAIL" or timeout error when being resolved on the server located in China. Hence given the nature of the blocked domains (e.g. BBC, Falun Dafa/Falun Gong, Free Tibet Campaign, Amnesty International), it appears that the Chinese government employs DNS manipulations to prevent users from accessing certain websites.

3.3. Search engines

Search engines are largely responsible for the variety and amount of information available to the users about a particular topic. Consequently if search engines are systematically manipulated to hide or even alter certain results, they would be the perfect instrument for censorship and enable an adversary to easily dictate the way users experience the web. In August 2006 the organization Human Rights Watch (HRW) published a paper [13] describing the involvement of multinational companies such as Google, Yahoo and Microsoft in assisting the Chinese government and their attempts to censor the Internet by manipulating major search engines. The document highlights two different ways in which the search results tend to be falsified in China:

1. Website de-listing: A manipulation in which an undesirable website is deliberately removed (delisted) from the list of search results.

2. Keyword censorship: A technique that prevents users from searching for specific keywords.

In order to identify the degree of the manipulations, Human Rights Watch performed sample searches for 25 sensitive and non-sensitive terms on the Chinese variants of Google (www.google.cn), Yahoo (cn.yahoo.com) and MSN (search.msn.com.cn) as well as on Baidu (www.baidu.cn), the leading domestic search engine in China. Afterwards the search results were compared with the U.S.-based counterparts of the search engines. The tests revealed that depending on the search terms used, all search engines in China are subject to filtering. These results are identical to independent experiments conducted by the author. However it must be noted that generally speaking "Chinese Internet users have access to significantly more information with Google.cn and the censored MSN operating in China. However, it appears that Yahoo! is censored at approximately the same level as Baidu, the domestic search engine leader" [13]. Furthermore the organization discovered that unlike Baidu, Google, MSN and Yahoo China will notify users in different ways if information has been censored. Consequently it appears that users in China are required to use some method of circumvention in order to get unfiltered search results.

3.4. Web browsing

Browsing the web is probably one of the major activities when using the Internet. Unfortunately in China it is also the quickest and yet most disturbing way of experiencing censorship. In addition to the aforementioned manipulations of the Domain Name System and popular search engines, the Chinese government also monitors all web browsing activities of their users. As Clayton discovered, the censoring works by inspecting web traffic for certain keywords (e.g. Falun Gong, Tibet, Taiwan) and once such a keyword has been identified, deliberately breaking the connection between the client and the server by sending forged RST packets to both endpoints. Additionally "once blocking has begun, it will remain in place for many minutes and further attempts by the same client to fetch material from the same website will immediately be disallowed by the injection of further forged resets" [14]. These results are equivalent to the author's impressions of browsing the web in China. It was found that websites such as www.amnesty.org are inaccessible.

4. Circumventing the filtering

4.1. Introduction

Circumventing or even attempting to circumvent Internet censorship is likely to infringe a countries law. Thus in order to avoid detection and ultimately legal consequences for the individuals involved, all circumvention techniques presented in this paper (and in fact any way of circumvention) should only be exercised with high caution. With that being said, the first step to circumvent large-scale Internet filtering is to attempt to identify the kind of filtering (e.g. DNS tampering) that is being em-

ployed by an adversary. Although the filtering is typically a black box at first sight, there are still a number of tests one could perform to make an educated guess of the filtering mode of operation. These tests are including but not limited to

1. Attempting to access information, services and websites that are likely to be blocked and studying the results.
2. Generating arbitrary TCP/IP packets with payloads possibly subject to censorship to enumerate the magnitude and strictness of the filtering.
3. Performing various DNS queries on local as well as multiple, randomly distributed public DNS servers from different countries and comparing the results.

If used in combination with a network sniffer such as Wireshark (formerly known as Ethereal) or tcpdump, these tests will most likely provide an individual with a high level of technical knowledge with basic information about the functionality and the magnitude of the filtering. Furthermore one could record and analyze all locally generated and received network traffic and try to identify inconsistencies (e.g. forged RST packets) within the traffic that are possibly caused by an intermediate filtering mechanism. Once sufficient information about the filtering has been gathered, one may choose an appropriate circumvention technique which is most likely to defeat the filtering mechanism in place. A relatively complete list and description of suitable techniques was published as part of a guide for bloggers and cyber-dissidents by the organization Reporters Without Borders [15]. Various other sources are also dedicated to bypassing Internet filtering [16]. However as the guide correctly mentions, there are also a number of disadvantages and risks associated with the majority of circumvention techniques including

- Primary point of Internet access: Depending on the nature of the primary point of Internet access (e.g. private computer, public computer in Internet cafe etc.), an individual may regardless of his level of technical knowledge be unable to use certain circumvention techniques simply because he is unable to install a particular piece of software or change the computer's settings in such a way that a filtering could be bypassed. Additionally depending on the type of service (e.g. instant messaging or Voice over IP communication) a user wants to use, he may be unable to bypass the filtering system as the required software product does not support alternative, non-standard configurations.
- Level of technical expertise: Generally speaking, users with a higher level of technical expertise are more likely to be able to circumvent a particular filtering mechanism. Additionally these kinds of users tend to have access to a greater number of circumvention techniques often involving the use of cryptography and special software or servers. Contrary inexperienced and non-technical users are typically limited to a few easy-to-use and well-known techniques that are unlikely to fully circumvent a countrywide filtering mechanism. Consequently the level of technical expertise a user possesses is vital for his success rate.
- Privacy and anonymity: Although some circumvention methods enhance a user's level of privacy and anonymity, they mostly do not provide perfect privacy or anonymity. Thus if a user was to reach a very high level of privacy and anonymity, he has to use a combination of various circumvention techniques and must also possess a high level of technical knowledge.
- Availability of trusted remote systems: If a user has access to a trusted computer system which is possibly located in a different country and thus unaffected by any filtering mechanism, he may use that system as a gateway to freely access any piece of information. Furthermore the use of strong encryption will prevent an adversary from observing the actions of an individual in any great detail. Unfortunately this is an option which is typically only available to a very small number of users. Hence the majority of users are limited to using publicly accessible servers which can easily be blocked by the operator of the filtering.
- Expected penalty and enforcement: Breaking a filtering system may put an individual's personal security at stake. Therefore if the enforcement of the filtering and the expected penalties are very strict, an individual must use particular caution when trying to bypass a filtering. In addition he must ensure to use a system which is secure and also provides the highest level of privacy as well as anonymity.

With these risks and disadvantages in mind, the following section presents and discusses some of the most popular techniques for circumventing Internet censorship. Additionally as each technique was thoroughly tested by the author on the aforementioned server, this section also comments on the applicability in the People's Republic of China.

4.2. The Clayton method

In June 2006 Richard Clayton, Steven Murdoch and Robert Watson published the first in-depth analysis of the mode of operation of the "Great Firewall of China" [14]. They discovered that the censorship employed by the Chinese government partially works "by inspecting

TCP packets for keywords that are to be blocked. If the keyword is present, TCP reset packets (viz: with the RST flag set) are sent to both endpoints of the connection, which then close". Interestingly, the original packets are not modified by this filtering and will pass the censorship unaltered. Hence the authors conclude that "if the endpoints completely ignore the firewall's resets, then the connection will proceed unhindered". Therefore they give the example of a Linux-based system in which a firewall rule such as

```
# iptables -A INPUT -p tcp --tcp-flags
RST RST -j DROP
```

will discard all packets with the RST flag set. This will lead to a situation in which the censoring is completely ineffective and in fact circumvented if such a rule is deployed on both ends of a communication channel. However it should be noted that their method also has a number of disadvantages limiting or even hindering the application of this technique in many situations:

1. In order to use this method one needs to have full control of both endpoints (i.e. client and server) of the connection to fully ignore TCP reset packets on either side. In reality having full control of these two endpoints is rarely the case as the user is usually only controlling the client but not a given remote server. If a user possesses full control of a client and a remote server, there are more effective ways (e.g. use of strong encryption, covert channels) of circumventing the censorship than simply ignoring TCP reset packets.

2. Even if an end-user has full control over one endpoint of the connection, he may often due to a lack of knowledge or architectural circumstances beyond his control, be unable to change the mode of operation of the network stack of his operating system.

3. The Internet standard document RFC 793 mentions that "as a general rule, reset (RST) must be sent whenever a segment arrives which apparently is not intended for the current connection" and lists a number of rules for RST generation and processing [17]. The document also specifies scenarios in which these types of packets must be used. Ignoring RST packets hence violates the defined standard and may cause communication problems.

In summary, it is undeniable that the aforementioned technique is a simple but yet very effective way of circumventing the way the Chinese censorship currently works. However Clayton's technique is not generic and as the censoring may be subject to change, his method may fail in the future. In addition, although it may render the blocking mechanism useless, it does not prevent the Chinese government from observing the communication between two parties. Thus the government might install other means (e.g. network-level filtering) to stop and prevent two parties from communicating with each other.

4.3. Alternative DNS servers

Given a situation in which a user's DNS server is subject to forgery, the easiest way to defeat these manipulations is to continuously use a publicly accessible DNS server which has not been tampered with. Consequently the user must not utilize the DNS servers he has been automatically assigned with by his provider but modify his network configuration to explicitly use external ones only which are not subject to tampering. A list of such alternative DNS servers can be obtained online, however it should be noted that due to security reasons some of these servers might refuse to answer queries from arbitrary clients or resolve a domain which they are not authoritative for. Once publicly accessible, unaltered DNS servers such as 213.133.99.99 or 213.133.100.100 have been identified, DNS tampering should no longer affect the user. Certainly if a provider also employs network-level filtering to prevent a user from querying alternative DNS servers, the user will have to use a more sophisticated method (e.g. tunneling) to successfully circumvent DNS tampering. In the People's Republic of China one simply has to exclusively use an unfiltered, foreign DNS server to defeat the manipulations at the DNS level and to successfully resolve domains (e.g. www.cnn.com) that are blocked by the national DNS servers. However as the government also employs additional means of filtering, these domains still remain inaccessible although they are being correctly resolved.

4.4. Alternative proxy servers

The use of an alternative proxy server is probably one of the simplest methods to circumvent loosely enforced censoring and enables even mostly unexperienced users to gain access to previously censored information. According to the aforementioned guide published by the organization Reporters Without Borders [15] there are in principle two different kinds of proxies:

- Web-based proxies
- Non web-based proxies

The first kind of proxy refers to a browser-based solution in which ad-featured access to other websites is provided. It can be used by simply accessing websites such as megaproxy.com offering this type of service and thus does not require altering the local browser settings. Contrary a non web-based proxy refers to a stand-alone web caching server which is freely accessible by the general public. In order to use a non web-based proxy one has to change his current browser settings and specify the proxy.

A frequently updated list of web-based and non web-based proxies can be found online. However as the communication with the proxy server is often not encrypted, an adversary may detect these circumvention attempts and hence block access to the proxy (e.g. using network-level filtering). Thus one should frequently switch proxies or chain them to cover their tracks. If access to all well-known proxies is forbidden, other and more sophisticated ways of circumvention (e.g. tunneling) must be deployed.

4.5. Tunneling

Tunneling refers to the process of encapsulating one protocol or a multiple of protocols inside another (often referred to as "transport protocol"). Depending on the nature of the transport protocol this encapsulation may or may not be transparent to the end-user and thus require changes to the system or application configuration. Furthermore if the transport protocol is using encryption, then the tunneled communication is also confidential until it reaches the end-point of the tunnel. However many tunneling protocols do not support encryption and hence transfer the tunneled data in plaintext unless the tunneled protocol itself is encrypted at the application layer. Typically tunneling is used in situations in which a tech-savvy user is experiencing severe limitations of the type or number of protocols that he successfully can use in a network environment to connect to a particular remote host or the Internet. If a user is for instance facing a network setup in which for whatever reason TCP and UDP connections are filtered, he may use the ICMP protocol as a transport medium to tunnel his entire network traffic to the outside world and hence bypass any filtering employed by the operator of the network. Beside such a transport protocol the user is also required to have secure and preferably privileged access to a remote system which can be used as the communication end-point of any tunnel. Thus the remote system must ideally not be subject to any filtering and allow the user to install as well as run additional pieces of networking software. If access to such a system is unavailable, one may purchase (shell) access from a provider ideally operating in a foreign country or try to use a provider such as Super Dimension Fortress (SDF) or ShellsNet that offers shell accounts for free.

4.5.1. ICMP tunneling

The Internet Control Message Protocol (ICMP) is used to "provide information about routing failures and to report about delivery error, congestion delays and other conditions on the network" [5]. Thus as mentioned above, the use of ICMP as a tunneling mechanism is one way to bypass TCP and UDP-based filtering because it is less likely to be subject to filtering. But like any other type of tunnel, an ICMP-based tunnel also requires the user to have access to a remote host which is unaffected by the filtering. Additionally one has to use a special piece of software such as "ptunnel" (ping tunnel) or "ICMPTX" (IP-over-ICMP) which are both freely available online. Ptunnel written by Daniel Stødle is an application to "to reliably tunnel TCP connections to a remote host using ICMP echo request and reply packets, commonly known as ping requests and replies" [18]. The software works by tunneling TCP connections over ICMP packets and sending these packets from the client to an intermediate host (called "proxy") which then opens a TCP connection to a previously defined remote server. Once the reply from the remote server comes in over TCP, the proxy converts the data back to ICMP echo replies and sends them on to the client that requested the connection. The mode of operation of ICMPTX is very similar but unlike ptunnel it does not tunnel TCP but complete IP connections. When being tested in China it was discovered that in general ICMP tunneling works fine and both of the aforementioned products can be used to successfully establish a tunnel with an intermediate host in Germany and thus freely communicate with any other host on the web. However with ptunnel it was found that the software resolves any target on the client rather than on the intermediate gateway. Thus it is vulnerable to DNS manipulations and does by default not provide a way to fully bypass the filtering. Only when being used in combination with foreign, unfiltered DNS servers it was discovered that ptunnel can be used to effectively bypass the Internet censorship in China. ICMPTX is not affected by this problem because it does not rely on the local DNS configuration.

4.5.2. SSH tunneling

The Secure Shell (SSH) is an application layer protocol based on public-key cryptography. In summary it provides "a powerful, convenient approach to protecting communications on a computer network. Through secure authentication and encryption technologies, SSH supports secure remote logins, secure remote command execution, secure file transfers, access control, TCP/IP port forwarding, and other important features" [19]. These advanced features of SSH such as port or X forwarding will enable a user to securely and reliably bypass any filtering. Thereby port forwarding refers to a transparent technique in which "insecure protocols running over TCP can be made secure by forwarding the connections through SSH" [19]. However like virtually all tunneling mechanisms, SSH tunneling also requires a user to have access to a remote computer system which is unaffected by any filtering and can be used as the end-point of the tunnel. Then in order to bypass a censorship, one may establish a cryptographically secure tunnel to a remote system via SSH and forward a local port to a HTTP proxy server running on the same or even a different re-

mote host. Additionally by altering the browser configuration to exclusively send all data to the SSH tunnel listening on the local system, every request will be transferred encrypted to the remote proxy, serviced and then confidentially transferred back to the requesting client. In addition once the SSH tunnel has been established, DNS queries are only performed by the remote system. Thus the user will not be affected by possible manipulations of local DNS servers. Instead one can securely and reliably access any website which was previously blocked by an adversary and in fact, use SSH to securely tunnel almost any protocol running over TCP. The other way of forwarding is called "X forwarding". X is the most popular window system for Linux/Unix systems and can be used to run X applications remotely. These applications can then display their windows locally or vice versa, run locally and have their display exported remotely. In X forwarding, SSH secures the underlying X protocol by tunneling its communication and therewith enabling a user to securely run remote X application (e.g. a browser) on a local display (or vice versa) [19]. In the Chinese part of the Internet it was discovered that the use of SSH is not prevented by any means. Accordingly if SSH access to an unfiltered, remote system is available, port and X forwarding are both suitable techniques to completely and securely bypass the filtering of TCP connections performed by the Chinese government. Thus websites such as www.worldpress.org can be accessed without any problems. Given the strong cryptography used by SSH as well as its technical level of sophistication, this tends to be a preferable way of circumvention in China.

4.5.3. SSL tunneling

When people think about the Secure Socket Layer (SSL) protocol they mostly refer to HTTPS which is the secure transport of HTTP over SSL/TLS. However in reality the SSL protocol is not specific to HTTP at all and in fact "is an authentication and encryption technique providing security services to TCP clients by the way of a Berkeley sockets-style API. It was initially developed by Netscape Communications Corporation to secure the HTTP protocol between web clients and servers, and that is still is primary use, though nothing about it is specific to HTTP" [5]. Consequently SSL can also be used to securely tunnel other protocols or even build the foundation of a Virtual Private Network (VPN). In order to use such a SSL-based VPN to bypass Internet censoring, one again needs access to a remote host which is unaffected by the filtering. Furthermore one is required to install an additional piece of software such as OpenVPN on both ends which is used to establish and manage the tunnel. This free piece of software "is a full-featured SSL VPN solution which can accommodate a wide range of configurations, including remote access, site-to-site VPNs, WiFi security, and enterprise-scale remote access solutions with load balancing, failover, and fine-grained access-controls" [20]. The author discovered that OpenVPN can be used to successfully establish a SSL-based tunnel between China and Germany and hence build a cryptographically secure way of fully bypassing the Chinese censorship.

4.5.4. Other ways of tunneling

In principle, given enough creativity and knowledge, almost every network or application protocol can be used to transport another and build a tunnel. However in reality the selection of the protocol to use largely depends on environmental circumstances that are often beyond the user's control (e.g. network or firewall setup). Thus in order to successfully bypass a filtering mechanism such as a firewall or countrywide censorship, one must choose an appropriate and sometimes even exotic way to tunnel through an adversarial network. In addition to the protocols mentioned above, DNS tunneling as implemented by tools such as NSTX or OzymanDNS, provides for instance one way of tunneling all IP traffic through a network in which only DNS queries and replies are allowed (e.g. public wifi hotspots). Furthermore if an Internet connection is restricted by a proxy server, one may use software like httptunnel to dispose HTTP requests to bypass a filtering and connect to a computer outside of the local network. Other and more exotic ways of tunneling are including but not limited to ACK- [21] or steganography-based solutions [22], receiving blocked web pages via email or using online translators as a gateway to access blocked information [23]. Finally one could use decentralized peer-to-peer (P2P) software such as The Onion Router (TOR) or Freenet that are dedicated to anonymize "web browsing and publishing, instant messaging, IRC, SSH, and other applications that use the TCP protocol" [24] to defeat censoring.

5. Conclusion

Generously supported by Western companies such as Cisco, Google, Microsoft or Yahoo, the Chinese government operates the world's most sophisticated and comprehensive Internet censoring system. Experiments conducted by the author as well as various papers published by international researchers indicate that the extent of this filtering truly is massive and pervasive. It prevents Chinese users by a multitude of non-transparent, technical and non-technical means from accessing or publishing information the government defines as illicit. Considering the enormous number of Chinese users and the general availability of high-speed Internet connections, the level of perfection of the censorship is surely frightening. In order to maintain such a high level of control in the future, the Chinese government will have to continuously invest a huge amount of money to try and keep up with technological advancement as well as with the rapidly

growing number of Internet users in China. This paper investigated a number of techniques that can be used to effectively bypass the filtering if a number of prerequisites are fulfilled. Unfortunately most of the circumvention methods available today are far too complicated for the average user and thus are more likely to be used by tech-savvy users or geeks only. Consequently new and alternative solutions (e.g. browsers with built-in support for TOR or other anonymizers) must be developed to enable even the average (Chinese) user to easily circumvent Internet censorship and freely access any piece of information. Censorship is futile!

6. References

[1] Amnesty International. Irrepressible.info, an amnesty international campaign. Campaign published on website http://irrepressible.info, 2006.

[2] United Nations. Universal declaration of human rights. UN Resolution 217 A (III) of 10 December 1948, 1948.

[3] Maximillian Dornseif. Government mandated blocking of foreign web content. In *DFN-Arbeitstagung über Kommunikationsnetze*, pages 617–647, 2003.

[4] Richard Clayton. Failures in a hybrid content blocking system. In *Privacy Enhancing Technologies*, pages 78–92, 2005.

[5] Wes Noonan and Ido Dubrawsky. *Firewall Fundamentals (Fundamentals (Cisco Press))*. Cisco Press, 2006.

[6] Peter Szor. *The Art of Computer Virus Research and Defense*. Addison-Wesley Professional, 2005.

[7] Marcus Ranum. What is deep inspection? Document published online at http://www.ranum.com/security/computer_security/editorials/deepinspect/index.html, undated.

[8] China Internet Network Information Center. 18th statistical survey report on the internet development in china. Report published as PDF available at http://www.cnnic.net.cn/download/2006/18threport-en.pdf, July 2006.

[9] OpenNet Initiative. Internet filtering in china in 2004-2005: A country study. Report published online at http://www.opennetinitiative.net/studies/china/, April 2005.

[10] C. Hughes. *China and the Internet: Politics of the Digital Leap Forward*. RoutledgeCurzon, 2003.

[11] Reporters Without Borders. Cyberdissidents imprisoned. Report published online at http://www.rsf.org/rubrique.php3?id_rubrique=119, 2006.

[12] Reporters Without Borders. Judges urged to acquit cyber-dissident li jianping on subversion charge. News item published online at http://www.rsf.org/article.php3?id_article=17016, April 2006.

[13] Human Rights Watch. race to the bottom - corporate complicity in chinese internet censorship. Report published online at http://www.hrw.org/reports/2006/china0806/index.htm, August 2006.

[14] Richard Clayton et. al. Ignoring the great firewall of china. Article available online at http://www.cl.cam.ac.uk/~rnc1/ignoring.pdf, 2006.

[15] Reporters Without Borders. Handbook for bloggers and cyber-dissidents. Manual published online at http://www.rsf.org/IMG/pdf/handbook_bloggers_cyberdissidents-GB.pdf, September 2005.

[16] Freerk. howto bypass internet censorship. Howto available online at http://www.zensur.freerk.com/, March 2006.

[17] Jon Postel (editor). *RFC 793 - Transmission Control Protocol*. Defense Advanced Research Projects Agency, September 1981.

[18] Daniel Stødle. Ping tunnel. Software published on http://www.cs.uit.no/~daniels/PingTunnel/, 2005.

[19] Daniel J. Barrett et. al. *SSH, the Secure Shell: The Definitive Guide*. O'Reilly Media, 2005.

[20] James Yonan. Openvpn. Software published on http://openvpn.net/, 2006.

[21] Arne Vidstrom. Ack tunneling trojans. Paper and proof of concept released at http://ntsecurity.nu/papers/acktunneling/, 2002.

[22] Syn Ack Labs. Steg tunnel. Software released at http://www.synacklabs.net/projects/stegtunnel/, 2003.

[23] bigthistle. Google free proxy! Comment posted on http://www.oreillynet.com/pub/h/4807, 2005.

[24] Roger Dingledine et. al. Tor: An anonymous internet communication system. Published on http://tor.eff.org/index.html.en, 2006.