The Great DNS Wall of China

Graham Lowe, Patrick Winters, Michael L. Marcus

December 21, 2007

1 INTRODUCTION

Repressive governments censor websites that they deem socially controversial. Subjects that they consider taboo include anything from pornography to health care, independent news to human rights propaganda—anything that contradicts their doctrine and that might incite rebellion.

Internet freedom advocacy sites [1] have studied and documented these censorship practices, enumerating the techniques employed by the censoring bodies. For example, censors block the IP addresses of controversial websites, inspect TCP packet exchanges for keywords and tamper with DNS records. These advocacy sites have already catalogued the types of sites that are censored and the means by which censorship is employed; this paper focuses specifically on DNS tampering.

Our investigation explores the DNS tampering implementation in China. A key challenge in studying DNS tampering is discovering DNS servers located within the censoring country. This paper contributes a reliable technique derived from prior work [2].

The remainder of this paper is organized as follows. Section 2 provides a simple example of DNS tampering as well as the rationale for a censor to implement DNS tampering. Section 3 presents the basic questions that we attempted to answer in our study. Section 4 describes our process for finding foreign DNS servers and censored websites, and section 5 details the design of our study. Section 6 describes the findings, section 7 proposes future research topics, and section 8 concludes.

2 BACKGROUND

2.1 A Simple Example

When someone types a domain name (e.g, *www.example.com*) into her web browser, the browser asks a DNS server for the associated IP address. If the server doesn't know the answer, it will consult its DNS authority or recursively[1] perform a lookup through the DNS hierarchy for the information. At any time in this process, misinformation may be introduced and unknowingly cached by users and other DNS servers.

Once the computer has the IP address for the domain, it makes a connection to the corresponding web server and downloads a web page. DNS tampering involves falsifying the response that is returned by the DNS server, either through intentional configuration or DNS poisoning. The server may lie about the associated IP address, any CNAMEs related to the domain, the authoritative servers for the domain, or any combination of the three.

2.2 Rationale For DNS Tampering

Given that a censoring body has a variety of censorship techniques at its disposal (IP blocking, content filtering, etc.), we first ask why they would bother tampering with DNS in the first place. Previous work [3] proposes that the primary motivation is to reduce censorship administration costs.

Consider a scenario in which censors rely solely on blocking the IP address of a website. The blocked

[1] Servers that do not support recursive queries will advise the client to consult either the authoritative TLD server or a root server.

site could circumvent this technique by changing its IP address, but the censors would simply block this new address. This cat-and-mouse cycle could continue indefinitely.

Now consider a second censoring body that blocks content at the domain-name level. Changing an IP address is trivial—most end users are unaware when a host changes its IP address—but changing a domain name is not. Since the domain name is the primary means of locating the website, the end users would need to be notified of its new domain name. By blocking websites at the domain level, censors effectively cripple the ability of websites to circumvent IP blocking.

3 QUESTIONS

Besides providing DNS tampering statistics (e.g., the percentage of servers that are compromised), we answered the following questions:

- *Are there patterns in the data that point to a centralized system of censorship?*

  Prior work[4] demonstrates that Chinese censors filter TCP traffic. Do they use a similar method for UDP-based DNS queries?

- *Do censors employ keyword filtering?*

  Staying abreast of new websites to censor is an exhausting task. Keyword filtering on domain name may help to ease censorship administration, acting as an automatic censor for new domains whose names match sensitive topics.

- *Are the returned IP addresses random, or do they come from a pool?*

  Using IP addresses from a pool can reduce censorship administration costs. Certain tasks such as IP blocking, content impersonation, and tracking are easier if a censor only needs to consider a small set of IPs.

4 PREREQUISTES

We had two prerequisites for conducting our study: a list of DNS servers located within a censoring country, and a list of web sites that were likely to be censored.

To find foreign DNS servers, we began with the assumption that hosts are located within the country associated with their TLD (e.g., www.google.cn is located in China). Also, based on prior work [2], we further assumed that DNS servers are closely located to the hosts for which they are authoritative. Armed with these assumptions, we ran search queries against Google using a combination of open-ended, top-level domain queries (e.g., ”site:.cn”) and keyword feedback to generate a list of 50,000 web sites. Then, by consulting a local DNS server, we found the authoritative DNS servers for each of the websites in the result set. We removed redundant servers, using IP address and hostname as discriminants, from our list. Using a widely available geolocation database for IP addresses [5], we filtered the list to include only those servers that were located within the censoring country and would therefore be more susceptible to tampering. This process yielded 2,896 servers. To distinguish which servers answered recursive queries, we queried each about a domain[2] for which none were authoritative. This filtering left us with 1,607 recursive servers.

To obtain a list of probable censored domains, we consulted two web sites. The first, *www.dit-inc.us*, provided us with a list of the top 10 censored domains as of 2002. The second, *cyber.law.harvard.edu*, provided 18,931 websites which were blocked as of 2002. From the latter list we added the subset that indicated DNS tampering (1014 websites), totaling 1,024 domains to test. We discarded 73 of the domains that indicated load balancing, leaving us with 951.

5 DESIGN

We conducted four experiments designed to answer our questions. Our first experiment involved query-

[2] *www.citizenterminal.net*

ing DNS servers about sensitive domains so that we could observe IP trends over a large set of responses. Our second experiment involved querying DNS servers about *fake* domains; the domains were designed to reveal whether DNS tampering was triggered by keyword. Our third experiment involved querying a single DNS server many times to observe whether its response was consistent. Finally, our last experiment involved generating IP packets and manipulating the TTL field in the IP header so that we could isolate where in the network path DNS tampering was implemented.

5.1 IP Trends

To determine the veracity of foreign DNS server responses, we established a canonical DNS response for each website by consulting five DNS servers in the U.S.[3] The reason for using multiple servers was to detect DNS-based load balancing; since this technique would complicate the analysis, we chose to exclude such websites. To further simplify our analysis, we also eliminated any responses for which the U.S. servers disagreed about the IP address.

For each Chinese DNS server we performed UDP-based DNS queries for each website, comparing the foreign responses against the canonical versions. When the answers differed, we assumed that the cause was DNS tampering. We repeated this test to confirm results.

5.2 Keyword Domains

To discover whether censors sent tampered DNS responses based on keywords in the domain name, we queried servers about a set of nonsense domains (e.g., pSyfA6srAZ0qCxU63.com, pSyfA6srAZ0qCxU63.biz, and pSyfA6srAZ0qCxU63.net).

The nonsense domains consisted of 3 domains that had a keyword as the subdomain (e.g., *falungong*, *voanews*, and *minghui*), 3 control domains that had *www* in the subdomain, and 366 domains with censored domains[4] embedded as the subdomain (e.g., 168net.cjb.net.pSyfA6srAZ0qCxU63.com, www.epochtimes.com.pSyfA6srAZ0qCxU63.com, etc.).

As in our previous test, we first queried the U.S. servers and then the Chinese servers. We also repeated this test to confirm results.

5.3 Consistency of Repeated Queries

We queried a single bad DNS server, as determined by our first experiment, 600 times about the same censored domain to observe whether its responses were consistent.

5.4 IP TTL Manipulation

To determine if DNS tampering is implemented at the DNS server or router level, we sent custom DNS message packets with a shortened IP TTL using hping [7] and used wireshark [6] to collect and analyze the corresponding response packets.

We chose an arbitrary bad server, as determined by our first experiment, and sent it several queries about a non-censored domain (*www.google.com*). We observed the ID and TTL fields in the response header and used these values as our control response. We then repeated the query about a censored domain (*168net.cjb.net*) and compared the IP ID and TTL fields of the responses. Finally, we repeated this process decreasing the TTL value to a point where it could no longer reach the destination DNS server and observed if there were any responses.

We conducted this test on the standard DNS port (i.e., 53) and the HTTP port (i.e., 80).

6 FINDINGS

This section describes the findings for the experiments we enumerated in section 5.

[3] The set of servers was arbitrary and assumed to be reliable.

	Round 1	Round 2
# Domains	854	841
Tampered Domains	393	383
# Distinct IPs For U.S. Server Responses	454	441
# Distinct IPs For Tampered Responses	21	18
% Tampered Servers	99.88%	99.88%

Table 1: IP Trend Statistics

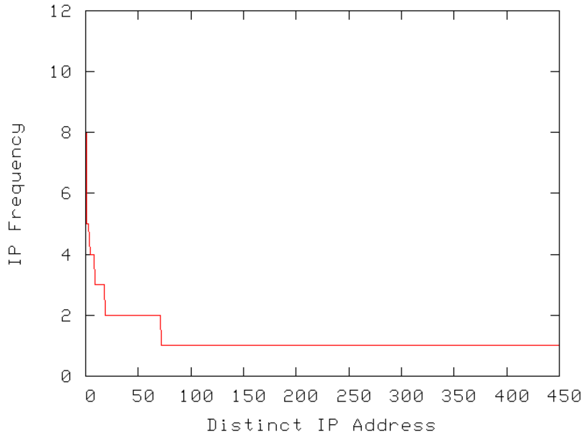

Figure 1: CDF Distinct IP Frequency Over Canonical Responses.

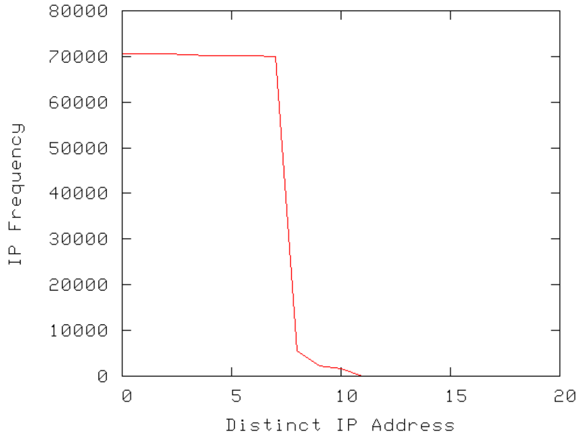

Figure 2: CDF Distinct IP Frequency Over Bad Responses.

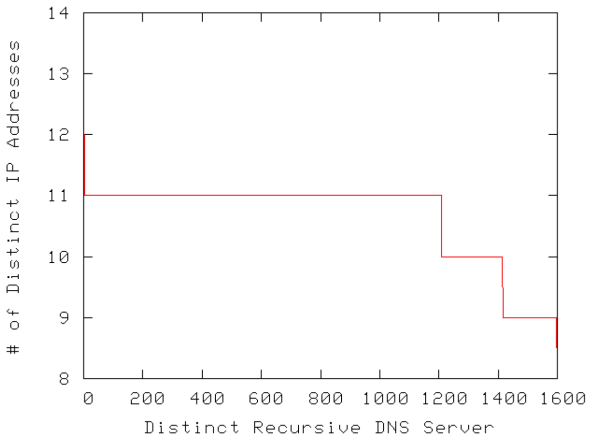

Figure 3: Number of Distinct IP Addresses from Censoring Servers.

6.1 IP Trends

We conducted two rounds of testing as described in section 5.1. To recap, we compiled responses from the five U.S. servers for the 951 domains. In the first round of this test, the U.S. servers were able to agree on 854 of the domains, while in the second there was agreement on only 841. Recall that in order for us to consider a response consistent, we required that all five U.S. DNS servers return the same IP address for a domain; if there was any disagreement, or even a time out, we discarded the domain from consideration for the round of testing.

Table 6.1 highlights some interesting figures for both rounds of tests. We observed that almost all of the Chinese DNS servers returned tampered responses for 383–393 domains and that the number of distinct IPs returned for these responses was extremely low when compared to the uniqueness of the correct responses. In fact, 366 bad domains shared eight IP addresses; figure 6.1 depicts the number of distinct IP addresses that each foreign server returned over the set of censored domains.

Furthermore, we have plotted for each distinct IP address, the frequency of its appearance across the re-

[4] We used the domains that we found to be reliably censored in our first experiment.

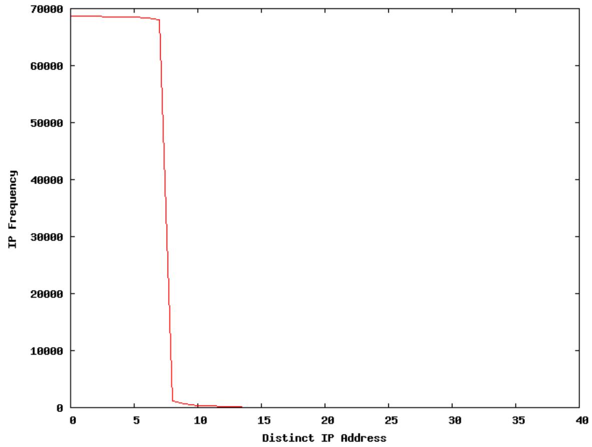

Figure 4: CDF Distinct IP Frequency Over Keyword Responses.

sult sets in figures 6.1 and 6.1. The canonical servers exhibit a high number of distinct IPs which appear infrequently amongst the responses[5]. On the other hand, the Chinese servers exhibit a handful of IP addresses that appear in most of the tampered responses[6].

6.2 Keyword Domains

We conducted two rounds of testing as described in section 5.2.

We found that, as expected, the canonical U.S. servers returned a "domain does not exist" for all of the nonsense domains. The Chinese servers exhibited different behavior. For both the domains with a control subdomain (i.e., www) and domains that had a keyword embedded as a subdomain, the majority of the servers returned a "domain does not exist"[7]. The nonsense domains with the censored domains embedded as the subdomain triggered the same type result as the previous experiment, returning the set of eight IP addresses.

hostname:port	ident	ttl	sequence #
www.google.com:53	50095	37	0
	50102	37	1
	50135	37	2
	50150	37	3
	50157	38	4
168net.cjb.net:53	64	41	0
	21595	77	0
	*53029	37	0
	64	41	1
	21491	85	1
	*53041	37	1
	64	41	2
	21452	88	2
	*53051	37	2
	64	41	3
	21348	32	3
	*53059	37	3
	64	41	4
	21218	42	4
	*53101	37	4

Table 3: IP Packet Information on Standard Port DNS Requests.

The CDF in 6.2 shows the number of distinct IP addresses appearing in the response set and each of their relative frequencies. As expected there were fewer than twenty distinct IP addresses returned, and the overwhelming majority trends to eight.

6.3 Repeated Question

We conducted two rounds of testing as described in section 5.3.

Our findings corroborated the results of the other two; for each of the 600 responses about the same domain, the server returned a random IP address from the pool of eight. Table 6.3 lists the eight IPs.

[5] The frequency sometimes deviates above 1, because some domains in the censored set refer to the same second-level domain (e.g., cjb.net)

[6] Eight, in particular, repeat as a factor of the size of the query set

[7] A few servers appear to be configured to return the IP address of the servers's ISP for all unknown domains, perhaps to deliver a customized error page to the end user.

IP Address	**Whois Lookup**	**Origin**
202.106.1.2	CNCGROUP	Beijing, CN
202.181.7.85	First Link Internet	North Rocks, AU
203.161.230.171	POWERBASE-HK	Hong Kong, HK
209.145.54.50	World Internet Services	San Marcos, CA, U.S.
211.94.66.147	China United Telecom	Beijing, CN
216.234.179.13	Tera-byte Dot Com	Edmonton, CA
4.36.66.178	Level 3 Communications	Broomfield, CO, U.S.
64.33.88.161	OLM,LLC	Lisle, IL, U.S.

Table 2: Whois Lookup for the "Bad Eight"

6.4 IP TTL Manipulation

We conducted two rounds of testing as described in section 5.4.

We chose an arbitrary bad DNS server and sent it several requests about a non-censored domain (*www.google.com*) and examined the responses. The IP ID and TTL fields were normal; the ID field was increasing and the TTL field was consistent. When we repeated our queries with an arbitrary censored domain (*168net.cjb.net*), we received strange results. We received several duplicate UDP packets. The IP ident field was inconsistent, as would be expected if the responses were sent from a stateless router instead of the DNS server. The IP TTL field was also wildly inconsistent. The payload—the actual DNS message containing the IP address—contained an IP from the set of eight.

To confirm that the tampered responses were originating from a router and not the server, we sent several requests about the non-censored domain, reducing the TTL until we received an ICMP response indicating that the packet had expired before it had reached its destination. Using this same TTL with requests about a censored domain resulted in tampered responses, indicating that a router is responsible for the tampering.

Table 6.4 displays the packet information and sequence numbers. Some duplicates were entirely identical (ignoring round trip time) and have been omitted for brevity. We have asterisked packets had a plausible TTL and a monotonically increasing ident field implying that these responses may be from the DNS server we queried. We infer that packets are not being dropped, and that bad responses are being sent in duplicate similar to the TCP reset method described in [4]. Unfortunately, the DNS payload of the packets from the DNS server are also inaccurate—they contain one of the bad eight IP addresses. This is expected, however, since the DNS server itself is caching tampered DNS responses it receives from within the Chinese network.

We repeated this test, sending the packet on port 80, but received no responses. This indicates that the filtering is occurring only on the standard DNS port.

7 FURTHER RESEARCH

There are several avenues for further research. We have started preliminary efforts to map the set of routers responsible for tampering; producing a visual topology of the *infected* route superimposed on a geographic map would produce a compelling image depicting the extent of DNS censorship. We have also started preliminary efforts to circumvent tampering through packet fragmentation, but our userspace program was not powerful enough to do proper fragmentation; this could prove whether or not filtering routers are kept stateless. We would also like to verify whether it would be possible to initiate queries about censored domains from within China to a U.S. DNS server and inspect the stream of packets for the true response. In addition, it may be enlightening to perform this experiment from other parts of the globe; Planetlab (*http://www.planet-lab.org/*)

has been suggested. Finally, we have found no connection between the set of eight IP addresses; contacting the owners and analyzing their access logs may shed light on the matter. Port scanning results on the eight IP addresses suggest that web servers are not listening at the majority of these destinations. By setting up a simple port listener at one of the U.S. addresses, we could easily log requests and reveal the amount of traffic redirected and the domains that are being censored.

8 CONCLUSION

Our investigation has revealed that Chinese censors employ DNS tampering at the router level, effectively poisoning all DNS servers on the route. We have also inferred that during filtering, the UDP packets are not being dropped, so it may be possible to obtain a true DNS response by issuing a request to a non-censoring country on a different port, or by issuing the request on the standard port and selectively listening for the correct response packet.

References

[1] Advocacy sites consulted were: www.opennet.net, www.dit-inc.us, and cyber.law.harvard.edu.

[2] Gummadi et al. King: Estimating Latency between Arbitrary Internet End Hosts. In *Proceedings of the SIGCOMM Internet Measurement Workshop* (IMW 2002).

[3] *http://www.dit-inc.us/report/hj.htm*

[4] Clayton, Murdoch, Watson: Ignoring the Great Firewall of China. *http://www.cl.cam.ac.uk/ rnc1/ignoring.pdf*

[5] *http://www.maxmind.com/app/ip-location*

[6] *http://www.wireshark.org/*

[7] *http://www.hping.org/*