Russian Collateral Censorship

Jesse Brizzi
jbrizzi@cs.stonybrook.edu

Konstantin Dmitriev
kdmitriev@cs.stonybrook.edu

Yingtao Tian
yittian@cs.stonybrook.edu

***Abstract*—In this project we are proposing a survey of neighboring sovereign nations to modern day Russia to investigate any possible collateral censorship. Given the socioeconomic state of some of these countries there may be limited infrastructure for Internet access, which requires their traffic to be routed through Russian territory. Russia actively censors various websites and Internet sources based on various reasons.**

Keywords*—*Collateral Censorship, Russia, Block, DPI, DNS, IP.

I. Motivation

In the summer of 2008, when Russia's mass media and telecom watchdog Roskomnadzor [1] was re-established, the Russian Internet, or RuNet, changed significantly. This Federal Service is regulated and put into motion by two laws, - "On Protecting Children from Information Harmful to Their Health and Development"[2] and "On Information, Information Technology and Information Protection"[3]. Both of them give judges a free hand in decision-making. As a result, a number of websites have been blocked quite chaotically, starting with opposition websites and articles, to Bitcoin communities and GitHub.

Sometimes such censorship systems can case collateral censorship, or damage. They block access to sites from users beyond those intended to protect[1]. This projects goal is to examine the effect of such possible collateral censorship to the requests that are originating from outside of Russia, with the possible extension to other countries that maintain the censoring services. The result of this project can be used to create a detailed analysis of collateral damage caused by different types of censoring techniques, and to potentially discover the paths at fault.

II. Related Work

Not a lot of research has been done in the area of the collateral censorship between networks in different countries. Partly because the impact of Internet censorship on global Internet service is usually unintended, and the probability of getting any results is fairly small. However, Chinas injection of forged DNS responses has been reported to cause large scale collateral damage by blocking outside traffic that traverses Chinese links [1]. The analysis shows that in the most extreme case, 70% of the open resolvers from Korea suffer collateral damage for queries to .de domains. Upstream filtering can also be behind traffic blockage outside of a censoring area due to ISP routing arrangements (for example, the Indian Internet filtering some users in Oman who are not able to access certain webpages [4]).

[1]Federal Service for Supervision of Communications, Information Technology and Mass Media (Russian:)

III. Censorship tools

The Russian Internet Service Providers (ISPs) and government use a number of different censorship techniques to block access to "unwanted" websites. Fig. 1 shows a chart of the most popular ones by the number of providers that maintain a particular method.

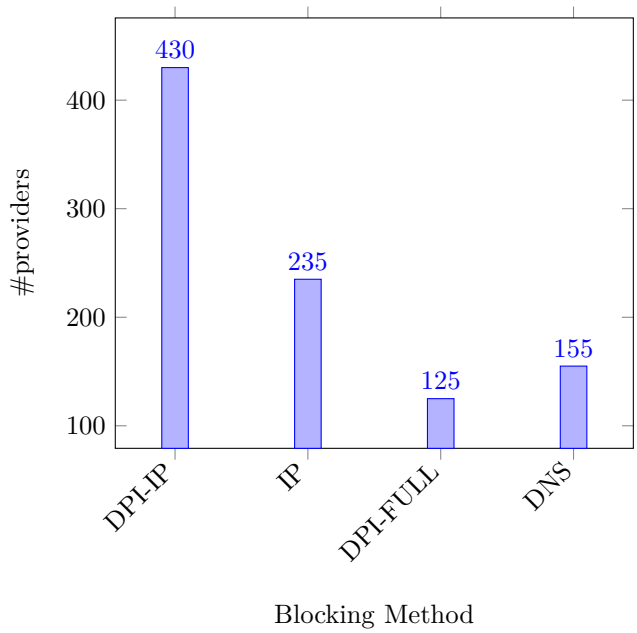


Figure 1: Number of providers that maintain a particular blocking method (DPI-IP - blocking using Deep Packet Inspection (DPI), that checks URL only at a specific IPs and port:80; DPI-FULL - blocking using DPI at every IP and ports; IP - blocking by IP address; DNS - DNS injection)

IV. Resources

As a main reference for blacklisted URLs, domain names, and IPs in Russia, we use lists provided by the ICLab, they are maintained specifically to test URL censoring. These lists are divided by country codes. Some websites are in English, some are in the local language

and picked individually by the regional expert. They have content representing a wide range of categories:

- Politics
- Social (sexuality, gambling, and illegal drugs and alcohol)
- Conflict/Security (armed conflicts, border disputes, separatist movements, and militant groups)
- Internet Tools (web sites that provide e-mail, Internet hosting, search, translation, Voice-over Internet Protocol (VoIP) telephone service)

A. Challenges

Manually analyzing lists of the blocked URLs for some countries we realized that some of them have the same censored websites, which, if not taking this into account, would give us wrong results - we might mistakenly think that some country is affected by the Russian collateral censorship, when in fact it has its own censorship. In order to overcome this problem, we have refiltered the list of the blocked URLs for Russia, our main list, in a way that have only unique URLs, that are blocked only in Russia. This procedure reduced the size of the Russian list down to 141 URLs.

As a reference of the potential victims of the collateral censorship, we use telegraphy maps (Fig. 2), and the map of the supported countries for one of the largest Russian backbone service provider (Fig. 3), concentrating on the bordering countries.

Also we are using VPNs provided by IP Vanish [2]

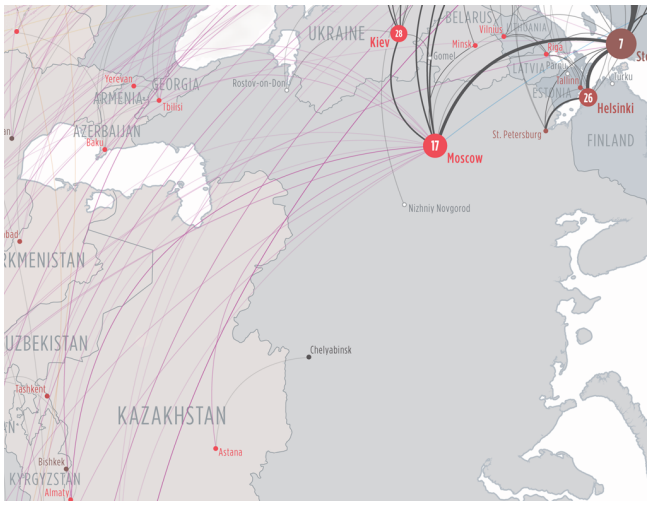


Figure 2: Telegeography Map of the Region

[2]https://www.ipvanish.com/

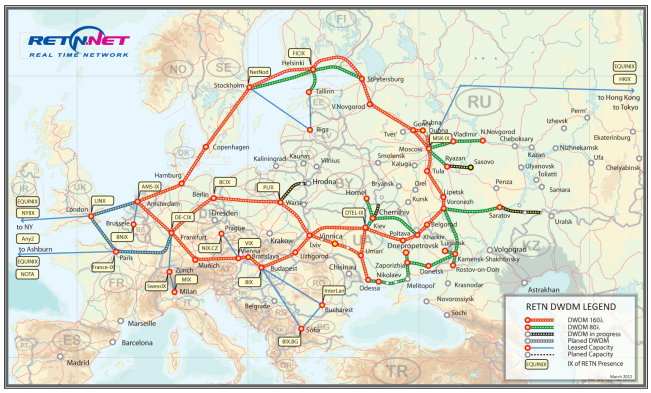


Figure 3: RETN's backbone map

V. Research Outline

A. Ultimate Aim

Our ultimate aim is to write an experiment in the form of Python scripts that will probe websites censored in Russia from various points in neighboring countries. This experiment will not only be able to compare the received web-pages with the blocked ones, but also return the score of the similarity between the received and the blocked web-pages. To collect the source code of blocked web-pages we use IPVanish and ExpressVPN servers.

In order to create this experiment, we use the following modules:

1) **urllib** - for network resource access;
2) **socket** - to get an access to the BSD socket interface;
3) **ssl** - to get an access to Transport Layer Security encryption and peer authentication facilities for network socket;
4) **dnspython** - to get an access to high and low levels of DNS.

and, of course, the OpenVPN [3] client to connect to VPN servers. Also we utilize the scripts from Tunnelblick [4] for connection setup.

B. Challenges

Given the possibility where no censorship leakage is found, we will restructure the experiment to try and prove our results are accurate, i.e. that there is little to no collateral censorship resulting from the Russian government. This is done by repeating the experiment multiple times at different times of the day, along with expanding our list of candidates for possible collateral censorship to other countries in close proximity geographically and in terms of Internet topology.

[3]http://openvpn.net/
[4]https://code.google.com/p/tunnelblick/

VI. Methodology

In order to measure possible collateral damage caused by Russian censorship, we conduct an experiment that can be coarsely divided into 3 parts:

1) Set a connection to a VPN server from a list of examined countries.
2) Access and collect HTML code of web-pages that are blocked in Russia.
3) Compare the percent of similarity between web-pages access from Russia and outside of it.

The overall process is illustrated in Figure 4.

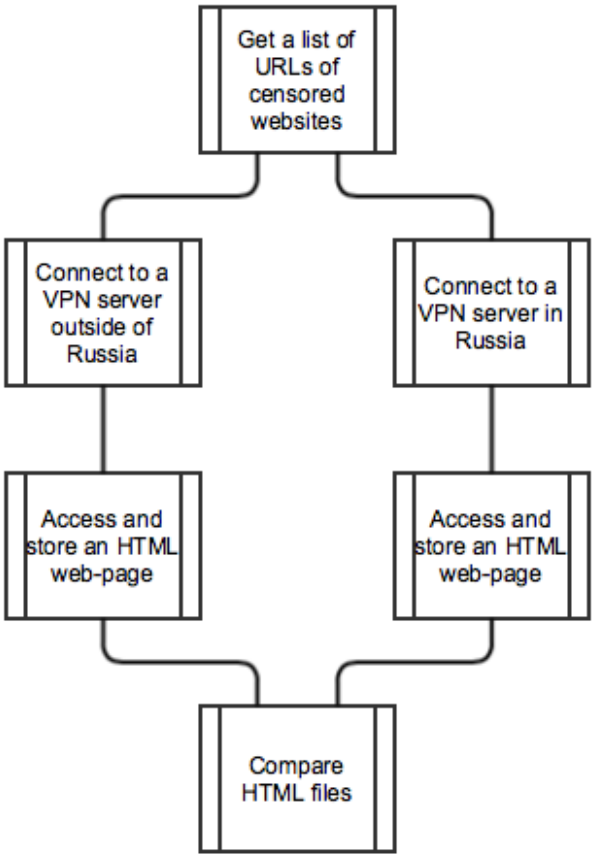


Figure 4: Scheme of the experiment

A. VPN

For the connections, we are facing the following challenges:

1) Access to web pages from different place, the basic need for our experiments, is required.
2) Renting machines, either physical or virtual, is beyond our consideration, because the renting is either too expensive, or just unavailable in several countries.
3) All communication, including data retrieval and other things like DNS request, should be done in the destination countries, because Collateral Censorship is sensitive to the path, which in turn is sensitive to DNS result.

We are using servers provided by IPVanish and ExpressVPN because they provide access from many countries. According to it's website [5], its servers "span 25,000+ IPs on 165+ servers in 60+ countries". This is a huge advantage as countries near Russia, like Finland and Estonia, are among countries supported by IP Vanish, thus we can investigate the influence of Russian censorship on its geographical neighbors. Also, we are using a computer directly located in Russia to get more accurate result for Russia censorship.

For our VPN client we are using OpenVPN. Basically, we conntect to a OpenVPN server hosted by IP Vanish using configuration files provided by IP Vanish and associated credits. Also we are using scripts from Tunnelblick to handle setting-up / tearing-down the connection on Mac OS X systems. We are running OpenVPN in daemon mode, communicating with it via a telnet server open locally for management purpose.

B. Data Collection

The high-level description of the data collection algorithm is presented in pseudocode on Algorithm 1.

Algorithm 1: Data Collection

```
Input: ListOfCountries, ListOfURLs;
for country in ListOfCountries do
    Establish a connection;
    for url in ListOfURLs do
        Access a webpage (timeout = 2 sec);
        Store its HTML code;
    end
end
```

The algorithm goes through every country in the list of selected countries, then it establishes a VPN connection, and then requests all 141 urls from the list of blocked URLs in Russia, it accesses the webpage within 2 seconds timeout and stores it's HTML code for the offline comparison.

For the purpose of data collection we implemented a special class - `CollectData`. Its instance connects to a specified VPN server and performs either collection of a single wep-page, specified by a URL from multiple countries, or collects a list of web-pages, using single VPN connection (Country major vs Website major order). This gives us the option of targeting a specific website in a small time windows to minimise differences in time sensitive content.

We chose to collect the webpage is country major order rather than website major order because to time restrictions. Collecting the website in website major order would be better as it would shrink the time window that

[5] https://www.ipvanish.com/why-vpn.php

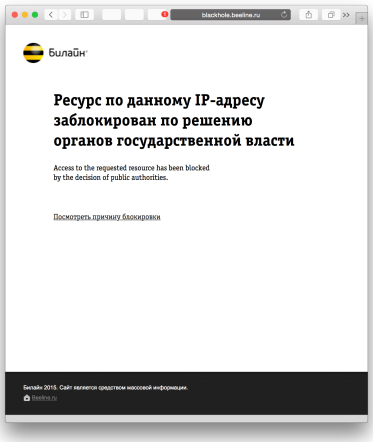


Figure 5: Example of provider's brochure page

we are accessing the webpages to minimize time based differences in the webpages. This is inefficient in times as the vpn takes multiple seconds to connect and disconnect from the server, and if we had to do this for every webpage rather than every country the data collection portion of our project would take days rather than hours.

After the web-page access attempt, there could be four possible outcomes:

1) An original web-page.
2) A web-page with a removed part (partial censorship, Figure 6).
3) A censorship's system brochure web-page (Figure 5).
4) A failure to load a web-page.

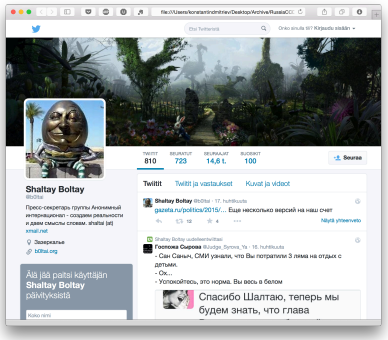
In order to deal with the final case, we stop waiting for a response after 2 seconds (Timeout parameter).

C. Web Page Comparison

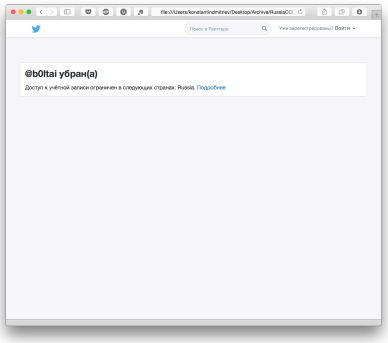
Web Page comparison is an essential part of our project. The aim is to compare the similarity between different copies of the same page accessed from difference countries. Censorship usually alters the content of web pages, so if a pages is effected by censorship in one country, the content varies between a copy accessed from this country, and one from a reference uncensored country. In this scenario, comparing the similarity between these different copies provides us an efficient way to detect and quantify the content changes.

However, the content changes could also happen due to reasons besides censorship. Some of the reasons could be temporal, for example a change of time stamp in these copies are totally reasonable, while another example may be recent news blocks in some news sites. Also valid reasons include geographical customization, such as local news that varies from one place to another one.

These two kinds of content changes are different: The changes due to censorship tend to replace the whole page,



(a) Latvia



(b) Russia

Figure 6: Example of the partial censorship

while the normal changes are more block-wise, and thus preserving general architecture of web pages. Therefore, Our algorithm should be sensitive to a fully change page, but meanwhile insensitive to block changes, especially small ones.

Algorithm 2: Longest Common Subsequence

Data: Two strings $S_{1...n}$ and $T_{1...m}$
Result: Length of longest common subsequence of S and T

$F_{0...n,0...m} \leftarrow 0$;
for $i \leftarrow 1$ **to** n **do**
  for $j \leftarrow 1$ **to** m **do**
    $F_{i,j} \leftarrow \max(F_{i-1,j}, F_{i,j-1})$;
    if $S_i = T_j$ **then**
      $F_{i,j} \leftarrow \max(F_{i,j}, F_{i-1,j-1} + 1)$
    end
  end
end
return $F_{n,m}$

We hereby propose the use of Longest Common Subsequence, or LCS, for scoring. LCS is a good metric for HTML file comparison, because it is relatively robust to block-wise content change (I.E a DOM tree node is replaced by another one, while preserving other parts of the

tree). The algorithm for LCS is detailed on Algorithm 2. More precisely, we use the following scoring formula:

$$Score(s_1, s_2) = \frac{LCS(s_1, s_2)}{Len(s_1) + Len(s_2)}$$

to ensure that our similarity metric would not prefer longer/shorter web pages. This metric means that a larger score means larger degree of similarity, also the score is constrainted in a given range ($0 \leq score \leq 0.5$), so the results are comparable for all page length.

However, a plain implementation of LCS is time-consuming: It requires $O(nm)$ time to compute where n and m are lengths of two strings, respectively. This is insufficient for HTML comparison because the average size of contemporary pages has already excessed 1.6 million bits[6], or 200 kilo-bytes.

Algorithm 3: Sift3b

```
Data: Two strings S_{1...n} and T_{1...m}, and O, the
      maximum offset
Result: Approximate length of longest common
        subsequence of S and T, with respect to
        maximum offset O
if n = 0 or m = 0 then
    return 0;
end
i ← 1;
j ← 1;
lcs ← 0;
while i ≤ n and j ≤ m do
    if S_i = T_j then
        lcs ← lcs + 1;
    else
        if i < j then
            j ← i;
        else
            i ← j;
        end
        for k ← 0 to O − 1 do
            if i + k ≤ n and S_{i+k} = T_j then
                i ← i + k;
                break;
            end
            if j + k ≤ m and S_i = T_{j+k} then
                j ← j + k;
                break;
            end
        end
    end
    i ← i + 1;
    j ← j + 1;
end
return lcs
```

This means that we need to use some algorithm that is more efficient. One such algorithm we are using is Sift3b [5], which is an approximate algorithm with speed optimizations, as shown in Algorithm 3.

Sift3b approximates the longest common subsequence by setting a parameter O, or max offset, and ignoring attempts to match characters whose offset is larger than the max offset. This means the running time of Sift3b is $O((n + m)O)$ where n and m are lengths of two strings, respectively. This gives us some kind of trade-off in parameters, because a larger max offset means a better approximation and slower running time. For our problem, we choose $O = 130$ for a balance between accuracy and running time.

VII. Results

After running the comparison algorithm and assigning every page a score, we measure the possibility of some country being affected by the collateral censorship by analyzing the data, same 141 webpages, but accessed from the USA. We ran the comparing algorithm once again, calculated the score for every page accessed from the USA, estimated the mean μ^{USA} and the standard deviation σ^{USA}. Then, we assume, that if some country has a mean value which is more than the mean value for the USA, then there is a high chance of the censorship leakage from Russia:

$$\mu^{country} > \mu^{USA} + k * \sigma^{USA} - country\ is\ affected$$

$$\mu^{country} \leq \mu^{USA} + k * \sigma^{USA} - country\ is\ not\ affected$$

For the final step, we've collected more than 25000 webpages, using IPVanish and ExpressVPN servers, the results are presented in Figure 7. This bar chart shows that 4 countries (Belarus, Estonia, Lithuania and Ukraine) show some signs of the possible censorship leakage.

There is another interesting finding. We were able to detect sings of the censorship leakage only from IP-Vanish vantage points. The possible explanation is that ExpressVPN circumvents the censorship on its own, since that's what most of their users would want.

VIII. Current Issues

A major obstacle that we plan to overcome in the future is the inconsistency of the censorship from our different viewpoints from *within* Russian borders. Using our various VPN subscriptions and a physical computer that we have access to in Russia we are observing different levels of access to certain websites. Our PC in Russia is censored in more cases then it is not when testing against the list of target websites. Where our VPN service through IP Vanish seems to get through to most of these websites. This may be due to the VPN company purposely fetching these locked websites as this would be a feature that most of its users would want or their servers are falsely labeled as being located in Moscow. We just happen to be in the fringe case of actually wanting to be censored.

[6] http://www.websiteoptimization.com/speed/tweak/average-web-page/

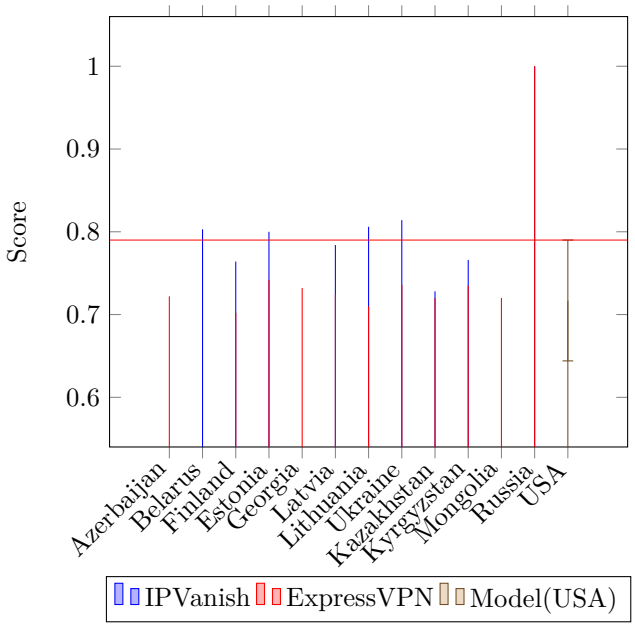


Figure 7: Results. Red bars are the mean values for the respective country estimated using ExpressVPN, blue bars are the mean values estimated using IPVanish. The most right bar is the mean and std values for the USA. The horizontal red line is the maximum mean value that indicates the possible collateral censorship for countries with higher scores.

We plan on either only collecting Russian samples from our dedicated PC or trying a difference VPN service. An issue may be similar de-censorship in the neighboring countries too, but we do not have dedicated machines in these locations to check. This may lead to a lot of invalid results in the end.

IX. Future Work

Currently we plan on focusing on expanding our HTML comparison methods to try and find what type of things are being changed if only parts of webpages are being censored. To account for different dynamic content that may change from instance to instance (ads, time dependant information).

We have references to methods of figure printing the type of censorship being used from the Russian ISPs, we would like to integrate this into our project as well.

References

[1] Anonymous, *The Collateral Damage of Internet Censorship by DNS Injection*. SIGCOMM Comput. Commun. Rev., July 2012.

[2] *"Law on Protecting Children from Negative and Harmful Information."* President of Russia. N.p., n.d. Web. 14 Feb. 2015.

[3] *Russian Federation: Federal Law No. 149-FZ* of July 24, 2006, on Information, Information Technology and Information Protection (as Amended up to Federal Law No. 398-FZ of December 28, 2013). N.p., n.d. Web. 14 Feb. 2015.

[4] C. Lab., *Routing Gone Wild: Documenting upstream filtering in Oman via India.* Technical report, Citizen Lab, 2012.

[5] S. Zackwehdex, *Super Fast and Accurate string distance algorithm: Sift3*, Web post, accessed at May 1, 2015, available at http://siderite.blogspot.com/2007/04/super-fast-and-accurate-string-distance.html.