

# Censorship in the Wild: Analyzing Web Filtering in Syria

Abdelberi Chaabane<sup>1</sup>, Mathieu Cunche<sup>2</sup>, Terence Chen<sup>3</sup>, Arik Friedman<sup>3</sup>,  
Emiliano De Cristofaro<sup>4</sup>, Mohamed-Ali Kaafar<sup>1,3</sup>

<sup>1</sup>INRIA Rhone Alpes   <sup>2</sup>INSA-Lyon/INRIA   <sup>3</sup>NICTA   <sup>4</sup>University College London

## ABSTRACT

Over the past few years, the Internet has become a powerful means for the masses to interact, coordinate activities, and gather and disseminate information. As such, it is increasingly relevant for many governments worldwide to surveil and censor it, and many censorship programs have been put in place in the last years. Due to lack of publicly available information, as well as the inherent risks of performing active measurements, the research community is often limited in the analysis and understanding of censorship practices. The October 2011 leak by the Telecomix hacktivist group of 600GB worth of logs from 7 Blue Coat SG-9000 proxies (deployed by the Syrian authorities to monitor and filter traffic of Syrian users) represents a unique opportunity to provide a snapshot of a real-world censorship ecosystem and to understand the underlying technology. This paper presents the methodology and the results of a measurement-based analysis of these logs. Our study uncovers a relatively stealthy yet quite targeted filtering, compared to, e.g., that of China and Iran. We show that the proxies filter traffic, relying on IP addresses to block access to entire subnets, on domains to block specific websites, and on keywords and categories to target specific content. Instant messaging is heavily censored, while filtering of social media is limited to specific pages. Finally, we show that Syrian users try to evade censorship by using web/socks proxies, Tor, VPNs, and BitTorrent. To the best of our knowledge, our work provides the first look into Internet filtering in Syria.

## 1. INTRODUCTION

As the relation between society and technology evolves, so does censorship—the practice of suppressing ideas and information that certain individuals, groups or government officials may find objectionable, dangerous, or detrimental to their interests. Inevitably, with the rise of the Internet, censors have increasingly targeted access to, and dissemination of, electronic information.

Several countries worldwide have put in place Internet filtering programs, using a variety of techniques. The purpose of such programs often includes restricting freedom of speech, controlling knowledge available to the masses, and/or enforcing religious or ethical principles. Although the research

community has dedicated a lot of attention to censorship and its circumvention, the understanding of filtering processes and the underlying technologies is limited. Naturally, it is both challenging and risky to conduct active measurements from countries operating censorship; also, real-world datasets and logs pertaining to filtered traffic are hard to come by.

Prior work has analyzed censorship practices in China [13, 19, 28, 14, 29], Iran [2, 25, 1], Pakistan [17], and a few Arab countries [8]. Yet, these studies are mainly based on probing in order to *infer* what information is being censored, e.g., by generating web traffic/requests and observing what content is blocked. While providing valuable insights, such a probing-based method suffers from two main inherent limitations. First, only a limited number of requests can be observed/tested, thus providing a skewed representation of the censorship policies (e.g., due to the inability to enumerate all censored keywords). Second, it is hard to assess the extent of the censorship, e.g., what proportion of the overall traffic (and what kind) is being censored.

In this paper, we present a large-scale measurement analysis of Internet censorship in Syria: we study a set of logs extracted from 7 Blue Coat SG-9000 filtering proxies, which are deployed to monitor, filter and block traffic of Syrian users. The logs (600GB of data) were leaked by a “hacktivist” group called Telecomix in October 2011, and relate to a period of 9 full days (July/August 2011) [22]. By analyzing these logs, we provide a detailed snapshot of how censorship was operated in Syria, a country that has been classified for several years as “Enemy of the Internet” by Reporters Without Borders [20].

Naturally, dealing with such a huge amount of data (600GB worth of logs) is a non-trivial task. Thus, we devise a methodology that balances between accuracy and feasibility. We adopt a random sampling approach to extract global statistics about our dataset. This allows us to produce accurate results while minimizing the computation complexity. However, whenever needed, we look at the full dataset (e.g., to extract all censored requests). Also, given the sensitive nature of the logs, we put in place a few mechanisms to safeguard users’ privacy (see Section 3.4).

As opposed to probing-based methods, the analysis of actual logs allows us to extract information about processed

requests for *both* censored and allowed traffic and provide a detailed snapshot of Syrian censorship practices. In the process, we uncover several interesting findings:

We observed that a few different techniques are used to block traffic: IP-based filtering to block access to entire subnets (e.g., in Israel), domain-based to block specific websites, keyword-based to target specific kinds of traffic (e.g., censorship- and surveillance-evading technologies, such as web/socks proxies), and category-based to target specific content and pages. As a side effect of keyword-based censorship (i.e., blocking all requests containing the word *proxy*), many HTTP requests are blocked even if they do not relate to any sensitive content or anti-censorship technologies (e.g., Google toolbar’s queries including `/tbproxy/af/query`).

The logs highlight that Instant Messaging software (e.g. Skype) is heavily censored while filtering of social media is limited to specific pages. In fact, most social networks (e.g., Facebook and Twitter) are not blocked, yet certain targeted pages (e.g., the Syrian Revolution Facebook page) are. One of our salient findings is that proxies have specialized roles and/or slightly different configurations, as some of them tend to censor more traffic than others. For instance, one particular proxy blocks Tor traffic for several days, while other proxies do not.

Finally, we show that Syrian Internet users not only try to evade censorship and surveillance using well-known web/socks proxies, Tor, and VPN software, but also use P2P file sharing software (BitTorrent) to fetch censored content.

Our analysis shows that, compared to other countries (such as China and Iran), Internet filtering in Syria seems to be less invasive yet quite targeted. Syrian censors particularly target Instant Messaging, information related to the political opposition (e.g., pages referring to the “Syrian Revolution”), and geo-politically significant content (i.e., Israeli domains). Arguably, less evident censorship does not necessarily mean minor information control or less ubiquitous surveillance. In fact, Syrian users seem to be aware of this and do resort to censorship- and surveillance-evading software, as we show later in the paper. Also, as reported by the Arabic Network for Human Rights Information [12] and the Open Net Initiative [18], Syrian Internet users exercise self-censorship to avoid being arrested [4].

Logs studied in this paper date back to July-August 2011, thus, our work is not intended to provide insights to the *current* situation in Syria. Naturally, censorship might have evolved in the last two years. For instance, according to [21], \$500k have been invested in surveillance equipment in late 2011, hinting at an even more powerful filtering architecture. Also, since December 2012, both Tor relays and bridges have started to be blocked [23]. However, observe that our work studies methods that are actually still in use (e.g., DPI). More importantly, the BlueCoat proxy servers we analyse in this paper are still used for censoring in e.g. Egypt, Kuwait and Qatar. Nonetheless, we argue that our work – by studying

a real-world censorship instance – serves as a valuable case study of censorship in practice. It provides a first-of-its-kind analysis of a real-world censorship ecosystem, exposing its underlying techniques, policies, as well as its strengths and weaknesses. which we hope will facilitate the design of censorship-evading tools.

**Summary of contributions.** To the best of our knowledge, we provide the first, detailed analysis of a snapshot of Internet traffic in Syria. We show how censorship is operated in Syria by performing several large-scale measurements of real-world logs extracted from 7 filtering proxies in 2011. We provide a statistical overview of the censorship activities, and a detailed analysis to uncover temporal patterns, proxy specializations, as well as filtering of social network sites. Finally, we provide some details on the usage and the censorship of surveillance- and censorship-evading tools.

**Paper Organization.** The rest of this paper is organized as follows. The next section reviews related work. Then, Section 3 provides some background information and introduces the datasets studied throughout the paper. Section 4 presents a statistical overview of Internet censorship in Syria based on the Blue Coat logs, while Section 5 provides a thorough analysis to better understand censorship practices. After focusing on social network sites in Section 6 and anti-censorship technologies in Section 7, we discuss our findings in Section 8. The paper concludes with Section 9.

## 2. RELATED WORK

The limited availability of real-world datasets, as well as the intrinsic risks of studying censorship from within countries with oppressive governments, make our large-scale analysis of actual logs quite unique. Little work so far has measured and analyzed datasets of real-world traffic and, to the best of our knowledge, no systematic study exists that analyzes Syria’s censorship machinery.

We now review relevant related work, focusing on censorship characterization and fingerprinting, as well as reports of censorship in a few countries worldwide.

A recent paper by Ayran et al. [2] presents a few measurements conducted from a major Iranian ISP, during the lead up to the June 2013 presidential election. They investigate the technical mechanisms used for HTTP host-based blocking, keyword filtering, DNS hijacking, and protocol-based throttling, concluding that the censorship infrastructure heavily relies on centralized equipment.

A few projects have also attempted to characterize censorship worldwide. For instance, Winter and Lindskog [28] conduct some measurements on traffic routed through Tor bridges/relays to understand how China blocks Tor, while Winter [27] proposes an analyzer for Tor, to be run by volunteers. Dainotti et al. [7] analyze two country-wide Internet outages (Egypt and Libya) using publicly available data, such as BGP inter-domain routing control plane data.

Another line of work involves fingerprinting and infer-

ring censorship methods and equipments. Researchers from Citizen Lab [16] attempt to recognize censorship and surveillance performed using Blue Coat devices, and uncover 61 Blue Coat ProxySG devices and 316 Blue Coat PacketShaper appliances in 24 countries. Similarly, Dalek et al. [8] use a confirmation methodology to identify URL filtering using, e.g., McAfee SmartFilter and Netsweeper, and detect the use of these technologies in Saudi Arabia, United Arab Emirates, Qatar, and Yemen.

Nabi [17] focuses on Pakistan: using a publicly available list of blocked websites, he checks their accessibility from multiple networks within the country. Results indicate that censorship varies across websites: some are blocked at the DNS level, while others at the HTTP level. Also, Anderson [1] creates some hosts inside Iran and discovers what he believes is a “private network” within the country. Furthermore, Verkamp and Gupta [25] detect censorship technologies in 11 countries, mostly using Planet Labs nodes, and discover DNS-based and router-based filtering.

Crandall et al. [6] propose an architecture for maintaining a censorship “weather report” about what keywords are filtered over time, while Leberknight et al. [15] provide an overview of research on censorship resistant systems and a taxonomy of anti-censorship technologies.

Knockel et al. [14] obtain a built-in list of censored keywords in China’s TOM-Skype and run experiments to understand how filtering is carried out. Bamman et al. [3] infer how traffic is blocked in Chinese social media, based on message deletion patterns on Sina Weibo, differential popularity of terms on Twitter vs. Sina Weibo, and looking at terms that are blocked on Sina Weibo’s search interface. King et al. [13] devise a system to locate, download, and analyze the content of millions of Chinese social media posts, before the Chinese government is able to censor them. They compare the substantive content of posts censored to those not censored over time in each of 85 topic areas.

Finally, Park and Crandall [19] present results from measurements of the filtering of HTTP HTML responses in China, which is based on string matching and TCP reset injection by backbone-level routers. Xu et al. [29] explore the AS-level topology of China’s network infrastructure, and probe the firewall to find the locations of filtering devices, finding that even though most filtering occurs in border ASes, choke points also exist in many provincial networks.

In conclusion, while a fairly large body of work has focused on understanding and characterizing censorship processes (especially in China and Iran), our work is the first to analyze a large-scale dataset of traffic observed by actual filtering proxies. In addition, we provide the first detailed snapshot of Syria’s censorship machinery.

### 3. BACKGROUND AND DATASETS DESCRIPTION

This section overviews the dataset studied in this paper, and background information on the proxies used for censorship.

#### 3.1 Data Sources

On October 4, 2011, a “hactivist” group called Telecomix announced the release of log files extracted from 7 Syrian Blue Coat SG-9000 proxies (aka ProxySG) [22].<sup>1</sup> According to Telecomix, these devices have been used by the Syrian Telecommunications Establishment (STE backbone) to filter and monitor all connections at a country scale. The data is split by proxy (SG-42, SG-43, . . . , SG-48) and covers two periods: (i) July 22, 23, 31, 2011 (only SG-42), and (ii) August 1–6, 2011 (all proxies). The leaked log files are in csv format (comma separated-values) and include 26 fields, such as date, time, filter action, host and URI (more details are given in Section 3.3).

**Disclaimer.** Given the nature of the dataset (leaked by a hactivist group), we cannot ultimately guarantee the authenticity of the data. Nonetheless, Blue Coat’s acknowledgment of the usage of its devices in Syria following the release of the data<sup>2</sup> confirms the provenance of the data.

#### 3.2 Blue Coat SG-9000 Proxies

The data released by Telecomix consists of 600GB log files extracted from 7 Syrian Blue Coat SG-9000 proxies. These appliances are designed to perform filtering, monitoring, and caching of Internet traffic, and are typically placed between a monitored network and the Internet backbone. They can be set as *explicit* or *transparent* proxies: the former setting requires the configuration of the clients’ browsers, whereas transparent proxies seamlessly intercept traffic (i.e., without clients noticing it), which is the case in this dataset.

Monitoring and filtering of traffic is conducted at the application level. Each user request is intercepted and classified as per one of the following three labels (as indicated in the *sc-filter-result* field in the logs):

- **OBSERVED** – request is served to the client.
- **PROXIED** – request has been found in the cache and the outcome depends on the cached value.
- **DENIED** – request is not served to the client because an exception has been raised (the request might be redirected).

In other words, the classification reflects the action that the proxy needs to perform, rather than the outcome of a filtering process. **OBSERVED** means that content needs to be fetched from the Origin Content Server (OCS), **DENIED** means that there is no need to contact the OCS, while **PROXIED** means that the outcome can be found in the proxy’s cache.

<sup>1</sup>The initial leak concerned 15 proxies but only data from 7 of them was publicly released. As reported by the Wall Street Journal [24], Blue Coat acknowledged that at least 13 of its proxies were used in Syria.

<sup>2</sup>See <http://www.bluecoat.com/company/news/update-blue-coat-devices-syria>.

According to Blue Coat’s documentation [26], filtering is based on multiple criteria: website categories, keywords, content type, browser type and date/time of day. The proxies can also cache content, e.g., to save bandwidth – this corresponds to the so-called “bandwidth gain profile”, as detailed in [5] (page 193).

### 3.3 Datasets and Notation

Throughout the rest of this paper, our analysis will use the following four datasets:

1. **Full Logs:** The whole dataset is composed of 751,295,830 requests. We denote as  $D_{full}$  the dataset extracted from all the logs.
2. **Sample Dataset:** Most of the results shown in this paper rely on the full extraction of the relevant data from  $D_{full}$ , however, given the massive size of the log files (~600GB), we choose to also consider when relevant a *random* sample covering 4% of the entire dataset, which we denote  $D_{sample}$ . This dataset is only used to illustrate a few results. We observe that according to standard theory about confidence intervals for proportions (see [?], Equation 1, Chapter 13.9.2), for a sample size of  $n = 32M$ , the actual proportion in the full data set lies in an interval of 0,0001 around the proportion  $p$  observed in the sample with 95% probability ( $\alpha = 0.05$ ).
3. **User Dataset:** Before leaking the data, Telecomix suppressed user identifiers (i.e., user IP addresses) by replacing them with zeros. However, for a small fraction of the data (July 22-23), the user identifier was instead substituted by a hash of the IP address, thus making user-based analysis possible. We refer to this dataset as  $D_{user}$ .
4. **Denied Dataset:** This dataset contains all the requests that resulted in exceptions ( $x\text{-exception-id} \neq \text{'-'}$ ), and is denoted as  $D_{denied}$ .

For each dataset, we report in Table 1 its size, corresponding dates, and number of proxies.

Dataset	# Requests	Period	# Proxies
Full	751,295,830	July 22-23,31, 2011 August 1-6, 2011	7
Sample (4%)	32,310,958	July 22-23, 2011 August 1-6, 2011	7
User	6,374,333	July 22-23 2011	1
Denied	47,452,194	July 22-23,31, 2011 August 1-6, 2011	7

Table 1: Datasets description.

Table 2 lists a few fields from the logs that constitute the main focus of our analysis.

The  $s\text{-ip}$  field logs the IP address of the proxy that processed each request, which is in the range 82.137.200.42 –

Field name	Description
$cs\text{-host}$	Hostname or IP address (e.g., facebook.com)
$cs\text{-uri-scheme}$	Scheme used by the requested URL (mostly HTTP)
$cs\text{-uri-port}$	Port of the requested URL
$cs\text{-uri-path}$	Path of the requested URL (e.g., /home.php)
$cs\text{-uri-query}$	Query of the requested URL (e.g., ?refid=7&ref=nf_fr&rdr)
$cs\text{-uri-extension}$	Extension of the requested URL (e.g., php, flv, gif, ...)
$cs\text{-user-agent}$	User agent (from request header)
$cs\text{-categories}$	Categories to which the requested URL has been classified (see Section 4 for details)
$c\text{-ip}$	Client’s IP address (removed or anonymized)
$s\text{-ip}$	The IP address of the proxy that processed the client’s request
$sc\text{-status}$	Protocol status code from the proxy to the client (e.g., ‘200’ for OK)
$sc\text{-filter-result}$	Content filtering result: DENIED, PROXIED, or OBSERVED
$x\text{-exception-id}$	Exception raised by the request (e.g., <i>policy_denied</i> , <i>dns_error</i> ). Set to ‘-’ if no exception was raised.

Table 2: Description of a few relevant fields from the logs.

48. Throughout the paper we refer to the proxies as SG-42 to SG-48, according to the suffix of their IP address.

The  $sc\text{-filter-result}$  field indicates whether the request has been served to the client. In the rest of the paper, we consider as *denied* all requests that have not been successfully served to the client by the proxy, including requests generating network errors as well as requests censored based on policy. To further classify a denied request, we rely on the  $x\text{-exception-id}$  field: all denied requests which either raise *policy\_denied* or *policy\_redirect* flags are considered as *censored*.

Finally, we observe some inconsistencies in the requests that have a  $sc\text{-filter-result}$  value set to PROXIED with no exception. When looking at requests similar to those that are PROXIED (e.g., other requests from the same user accessing the same URL), some are consistently denied, while others are sometimes or always allowed. Since PROXIED requests only represent a small portion of the analyzed traffic (< 0.5%), we treat them like the rest of the traffic and classify them according to the  $x\text{-exception-id}$ . However, where relevant, we refer to them explicitly to distinguish them from the OBSERVED traffic.

In summary, throughout the rest of the paper, we use the following request classification:

- **Allowed** ( $x\text{-exception-id} = \text{'-'}$ ): a request that is allowed and served to the client (no exception raised).
- **Denied** ( $x\text{-exception-id} \neq \text{'-'}$ ): a request that is not served to the client, either because of a network error or due to censorship. Specifically:
  - **Censored** ( $x\text{-exception-id} \in \{policy\_denied, policy\_redirect\}$ ): a *denied* request that is censored based on censorship policy.
  - **Error** ( $x\text{-exception-id} \notin \{ \text{'-'}, policy\_denied, policy\_redirect \}$ ): a *denied* request not served to the client due to a network error.

- *Proxied* ( $sc-filter-result = \text{PROXIED}$ ): a request that does not need further processing, as the response is in the cache (i.e., the result depends on a prior computation). The request can be either allowed or denied, even if  $x-exception-id$  does not indicate an exception.

### 3.4 Ethical Considerations

Although the studied dataset is publicly available, we are obviously aware of its sensitivity. Thus, we put in place a few mechanisms to safeguard privacy of Syrian users. Specifically, we encrypted all data (and backups) at rest and did not re-distribute the logs. Also, special cautionary measures were taken during the analysis not to obtain or extract users’ personal information, and we only analyzed aggregated statistics of the traffic. While it is out of the scope of this paper to further discuss the ethics of using “leaked data” for research purposes (see [11] for a detailed discussion), we point out that analyzing logs of filtered traffic, as opposed to probing-based measurements, provides an accurate view for a large-scale and comprehensive analysis of censorship.

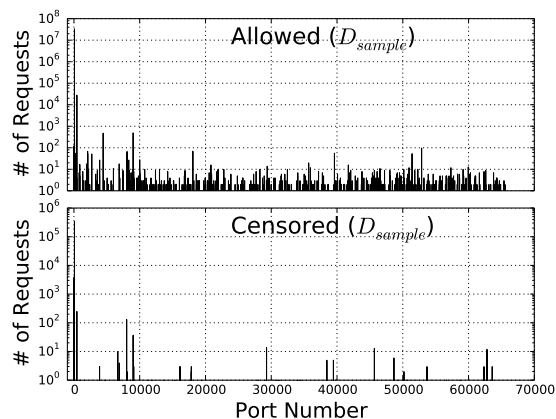
We acknowledge that our work can actually be beneficial to entities on either side of censorship. However, we believe that our analysis is crucial to better understand the technical aspects of a real-world censorship ecosystem, and that our methodology exposes its underlying technologies, policies, as well as its strengths and weaknesses (and thus can facilitate the design of censorship-evading tools).

## 4. A STATISTICAL OVERVIEW OF CENSORSHIP IN SYRIA

Aiming to provide an overview of Internet censorship in Syria, our first step is to compare the statistical distributions of the different classes of traffic (as defined in Section 3.3), and also look at domains, TCP/UDP ports, website categories, and HTTPS traffic.

Unless explicitly stated otherwise, the results presented in this section are based on the full dataset denoted as  $D_{full}$  (see Section 3.3).

**Traffic distribution.** We start by observing the ratio of the different classes of traffic. For each of the datasets  $D_{sample}$ ,  $D_{user}$  and  $D_{denied}$ , Table 3 reports how many requests are allowed, proxied, denied, or censored. In  $D_{sample}$ , more than 93% of the requests are allowed, and less than 1% of them are censored due to policy-based decisions. The number of censored requests seems relatively low compared to the number of allowed requests. Note, however, that these numbers are skewed because of the request-based logging mechanism, which “inflates” the volume of allowed traffic; a single access to a web page may trigger a large number of requests (e.g., for the html content, accompanying images, scripts, tracking websites and so on) that will be logged, whereas a *denied* request (either because it has been censored or due to a network error) only generates one log entry. Finally, note that only a small fraction of requests are proxied (0.47% in  $D_{sample}$ ).



**Figure 1:** Destination port distributions of allowed and censored traffic ( $D_{sample}$ ).

The breakdown of  $x-exception-id$  values within the proxied requests resembles that of the overall traffic.

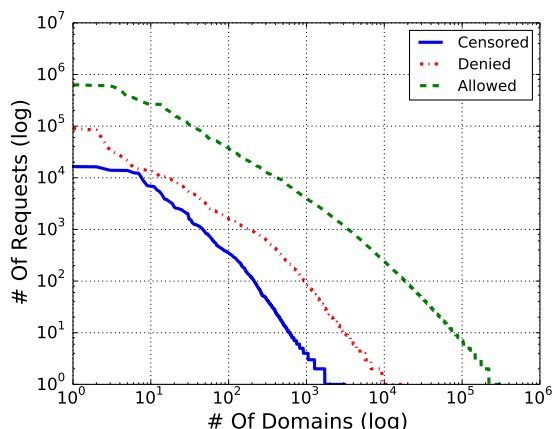
**Denied traffic.** As mentioned earlier, proxies also log requests that have been denied due to network errors. In our sample, this happens for less than 6% of the requests. The inability of the proxy to handle the request (identified by the  $x-exception-id$  field being set to *internal\_error*) accounts for 31.15% of the overall denied traffic. Although this could be considered censorship (no data is received by the user), these requests do not actually trigger any policy exception and are not the result of policy-based censorship. TCP errors, typically occurring during the connection establishment between the proxy and the target destination, represent more than 45% of the denied traffic. Other errors include DNS resolving issues (0.41%), invalid HTTP request or response formatting (5.65%), and unsupported protocols (1.46%). The remaining 15.33% of denied traffic represent the actual censored requests, which the proxy flags as denied due to policy enforcement.

**Ports.** We also look at the traffic distribution by port number for both allowed and censored traffic (in  $D_{sample}$ ). We report it in Fig. 1. Ports 80 and 443 (HTTPS) represent the majority of censored content. Port 9001 (usually associated with Tor servers) is ranked third in terms of blocked connections. We discuss Tor traffic in more detail in Section 7.1.

**Domains.** Next, we analyze the distribution of the number of requests per unique domain. Fig. 2 presents our findings. The y-axis (log-scale) represents the number of (allowed/denied/censored) requests, while each point in the x-axis (also log-scale) represents the number of domains receiving such a number of requests. Unsurprisingly, the curves indicate a power law distribution. We observe that a very small fraction of hosts ( $10^{-5}$  for the allowed requests) are the target of between few thousands to few millions requests, while the vast majority are the destination of only few requests. Allowed traffic is at some point one order of magnitude bigger, this

sc-filter-result	x-exception-id	Classification	Sample ( $D_{sample}$ )		User ( $D_{user}$ )		Denied ( $D_{denied}$ )	
OBSERVED	–	<i>Allowed</i>	30,140,158	(93.28%)	6,038,461	(94.73%)	–	–
PROXIED	(total)	<i>Proxied</i>	151,554	(0.47%)	26,541	(0.42%)	267,354	(0.56%)
DENIED	(total)	<i>Denied</i>	2,019,246	(6.25%)	309,331	(4.85%)	47,184,840	(99.44%)
	tcp_error	<i>Error</i>	947,083	(2.93%)	54,073	(0.85%)	21,499,871	(45.30%)
	internal_error		636,335	(1.97%)	198,058	(3.11%)	14,720,952	(31.02%)
	invalid_request		115,297	(0.36%)	36,292	(0.57%)	2,668,217	(5.62%)
	unsupported_protocol		28,769	(0.09%)	1,348	(0.02%)	719,189	(1.51%)
	dns_unresolved_hostname		6,247	(0.02%)	3,856	(0.06%)	141,558	(0.30%)
	dns_server_failure		2,235	(0.01%)	396	(0.01%)	58,401	(0.12%)
	unsupported_encoding		6	(0.00%)	0	(0.00%)	269	(0.00%)
	invalid_response		1	(0.00%)	2	(0.00%)	8	(0.00%)
	policy_denied		283,197	(0.88%)	15,306	(0.24%)	7,374,500	(15.54%)
policy_redirect	76	(0.00%)	0	(0.00%)	1,875	(0.04%)		

**Table 3:** Statistics of different decisions and exceptions in the three datasets in use.



**Figure 2:** The distribution of the number of requests per unique domain.

happens for at least two reasons: (i) allowed requests target highly popular websites (e.g., Google and Facebook), and (ii) an allowed request is potentially followed up by additional requests to the same domain, whereas a denied request is not.

In Tables 4 and 5, respectively, we report the top-10 allowed (resp., censored) domains in  $D_{sample}$ . Unsurprisingly, google.com and its associated static/tracking/advertisement components represent nearly 15% of the total allowed requests. Other well-ranked domains include facebook.com (and its associated CDN service, fbcdn.net) and xvideos.com (a pornography-associated website). The top-10 censored domains exhibit a very different distribution: facebook.com (and fbcdn.net), skype.com and metacafe.com (a popular user-contributed video sharing service) account for more than 43% of the overall censored requests. Websites like Facebook and Google are present both in the censored and the allowed traffic, since the policy-based filtering may depend on the actual content the user is fetching rather than the website, as we will explain in Section 6. Finally, observe that mediafire.com is ranked at #9 in the top non-censored domains: according to the Electronic Frontier Foundation (EFF), mediafire.com was actually used to deliver malware targeting Syrian activists.<sup>3</sup>

<sup>3</sup><https://www.eff.org/deeplinks/2012/12/internet-back-in-syria-so-is-malware>

Domain	# Of Requests	Percentage
google.com	2.26M	7.51
gstatic.com	1.03M	3.44
xvideos.com	876,933	2.9
facebook.com	769,558	2.55
microsoft.com	740,323	2.45
fbcdn.net	654,873	2.17
windowsupdate.com	652,357	2.16
google-analytics.com	553,910	1.83
doubleclick.net	518,152	1.71
msn.com	498,523	1.65
yting.com	470,255	1.56
mediafire.com	392,056	1.30
yahoo.com	320,517	1.06

**Table 4:** Top-10 allowed Domains ( $D_{sample}$ ).

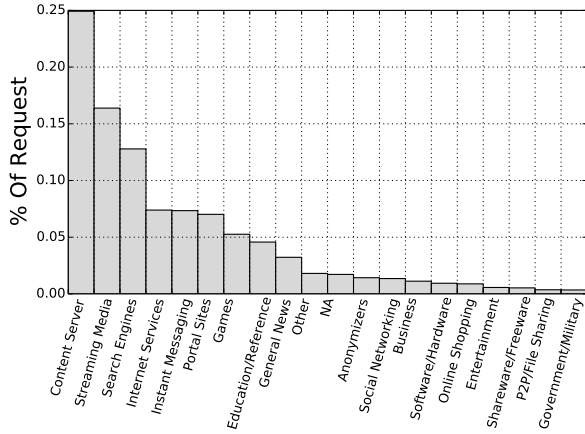
Domain	# Of Requests	Percentage
facebook.com	68,782	24.28
skype.com	23,558	8.31
metacafe.com	19,257	6.79
live.com	18,861	6.65
google.com	18,154	6.40
zynga.com	16,775	5.92
yahoo.com	16,368	5.77
wikimedia.org	13,506	4.76
fbcdn.net	12,531	4.42
ceipmsn.com	6,146	2.16
conduitapps.com	5,092	1.79
msn.com	3,758	1.32
conduit.com	3,310	1.16

**Table 5:** Top-10 censored Domains ( $D_{sample}$ ).

**Categories.** The Blue Coat proxies support filtering according to URL categories. This categorization can be done using a local database, or using Blue Coat’s online filtering tool.<sup>4</sup> However, according to Blue Coat’s representatives [24], the online services are not accessible to the Syrian proxy servers, and apparently the Syrian proxy servers are not using a local copy of this categorization database. Indeed, the *cs-categories* field in the logs, which records the URL categories, contains only one of two values: one value associated with a default category (named “unavailable” in five of the proxies, and “none” in the other two), and another value associated with a custom category targeted at Facebook pages (named “Blocked sites; unavailable” in five of the proxies, and “Blocked sites” in the other two), which is discussed in

<sup>4</sup><http://sitereview.bluecoat.com/categories.jsp>



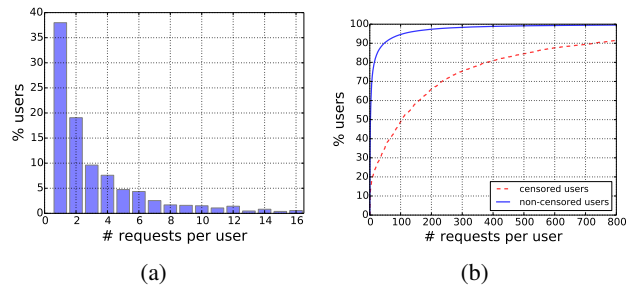


**Figure 3:** Category distribution of censored traffic ( $D_{sample}$ ), for categories obtained from McAfee’s TrustedSource. ‘NA’ denotes not available, and ‘Other’ is used for categories with less than 1K requests.

more details in Section 6.2.

Due to the absence of URL categories, we rely on McAfee’s TrustedSource tool, available at [www.trustedsource.org](http://www.trustedsource.org), to characterize the censored websites. Fig. 3 shows the distribution of the censored requests across the different categories. The “Content Server” category ranks first, with more than 25% of the blocked requests (this category mostly includes CDNs that host a variety of websites, such as [cloudfront.net](http://cloudfront.net), [googleusercontent.com](http://googleusercontent.com)). “Streaming Media” are next, hinting at the intention of the censors to block video sharing. “Instant Messaging” (IM) websites, as well as “Portals Sites”, are also highly blocked, possibly due to their role in coordination of social activities and protests. Note that both Skype and [live.com](http://live.com) IM services are always censored and belong to the top-10 censored domains. However, surprisingly, both “News Portals” and “Social Networks” rank relatively low: as we explain in Section 6, censorship only blocks a few well-targeted social media pages. Finally, categories like “Games” and “Education/Reference” are also occasionally blocked.

**HTTPS traffic.** In our logs, the number of HTTPS requests is a few orders of magnitude lower than that of HTTP requests. HTTPS accounts for 0.08% of the overall traffic and only a small fraction (0.82%) is censored ( $D_{sample}$  dataset). It is interesting to observe that, in 82% of the censored traffic, the destination field indicates an IP address rather than a domain, and such an IP-based blocking occurs at least for two main reasons: (1) the IP address belongs to an Israeli AS, or (2) the IP address is associated with an Anonymizer service. The remaining part of the censored HTTPS traffic actually contains a hostname: this is possible due to the use of the HTTP CONNECT method, which allows the proxy to identify both the destination host and the user agent (for instance, all connections to Skype servers are using the CONNECT



**Figure 4:** (a) Number of censored requests per user in  $D_{user}$ ; (b) The distribution of the overall number of requests per user (both allowed and denied), for censored and non-censored users.

method, and thus the proxy can censor requests based on the [skype.com](http://skype.com) domain).

According to the Electronic Frontier Foundation, the Syrian Telecom Ministry has launched man in the middle (MITM) attacks against the HTTPS version of Facebook.<sup>5</sup> While Blue Coat proxies indeed support interception of HTTPS traffic,<sup>6</sup> we do not identify any clear sign of such an activity. For instance, the values of fields such as *cs-uri-path*, *cs-uri-query* and *cs-uri-extension*, which would have been available to the proxies in a MITM attack, are not present in HTTPS requests. However, also note that, by default, the Blue Coat proxies use a separate log facility to record SSL traffic,<sup>7</sup> so it is possible that this traffic has been recorded in logs that were not obtained by Telecomix.

**User-based analysis.** Based on the  $D_{user}$  dataset, which comprises the logs of proxy SG-42 from July 22-23, we analyze user behavior with respect to censorship. We assume that each unique combination of *c-ip* (client IP address) and *cs-user-agent* designates a unique user. This assumption does not always hold – for example, a single user may use several devices with different IP addresses (or a single device with different browsers), and users who use similar browsers (with identical user agent strings) may share the same IP address through NAT. However, this combination of fields provides the best approximation of unique users within the limits of the available data [30].

We identify 147,802 total users in  $D_{user}$ , 2,319 (1.57%) of them generate at least one request that is denied due to censorship. Focusing on this subset of users, Fig. 4(a) shows the distribution of the number of censored requests per user. 37.8% of those users only have one single request censored during the observed period. Typically, users do not attempt to access a URL again once it is blocked, but, in some cases, we do observe a few more requests to the same URL. Overall, for 93.87% of the users, all the censored requests (one or more per user) are to the same domain.

<sup>5</sup><https://www.eff.org/deeplinks/2011/05/syrian-man-middle-against-facebook>

<sup>6</sup><https://kb.bluecoat.com/index?page=content&id=KB5500>

<sup>7</sup>See <https://bto.bluecoat.com/doc/8672>, page 22.

Fig. 4(b) shows the distribution of the number of overall requests per user, for both non-censored and censored users, where a censored user is defined as a user for whom at least one request was censored. We found that the censored users are more active than non-censored users, observing approximately 50% of the censored users have sent more than 100 requests, while only 5% of non-censored users show the same level of activity. As we discuss in Section 5.4, many requests are censored since they happen to contain a blacklisted keyword (e.g., *proxy*), even though they may not be actually accessing content that is the target of censorship. Since active users are more likely to encounter URLs that contain such keywords, this may explain the correlation between the user level of activity and being censored. We also observe that in some cases the user agent field refers to a software repeatedly trying to access a censored page (e.g., *skype.com*), which augments the user’s activity.

**Summary.** Our measurements have shown that only a small fraction (less than 1%) of the traffic is actually censored. The vast majority of requests is either allowed (93.28%) or denied due to network errors (5.37%). Censorship targets mostly HTTP content, but several other services are also blocked. Unsurprisingly, most of the censorship activity targets websites that support user interaction (e.g., Instant Messaging and social networks).

A closer look at the top allowed and censored domains shows that some hosts are in both categories, thus hinting at a more sophisticated censoring mechanism, which we explore in the next sections.

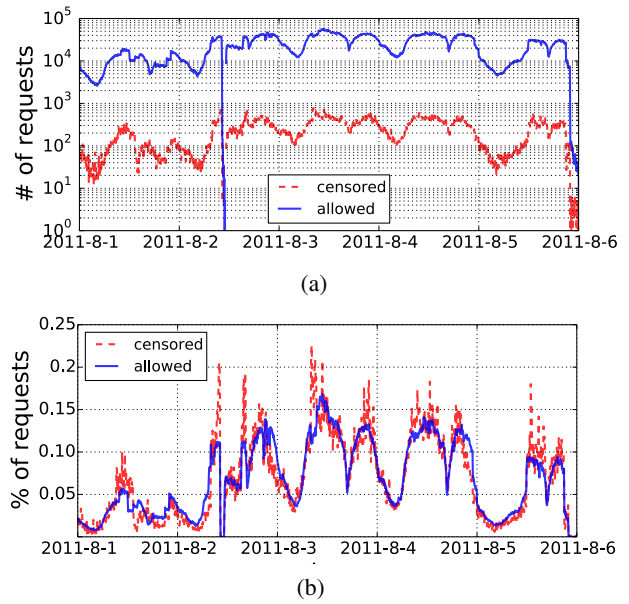
Finally, our user-based analysis has shown that only a small fraction of users are directly affected by censorship.

## 5. UNDERSTANDING THE CENSORSHIP POLICY

This section aims to understand the way the Internet is filtered in Syria. First, we analyze censorship’s temporal characteristics and compare the behavior of different proxies. Then, we study *how* the requests are filtered and infer the characteristics on which censorship policies are based.

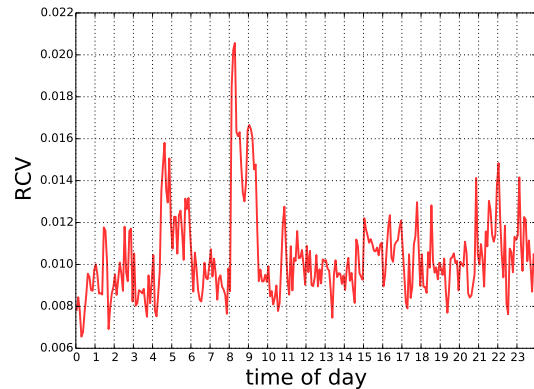
### 5.1 Temporal Analysis

We start by looking at how the traffic volume of both censored and allowed traffic changes over time (5 days), with 5-minute granularity. The corresponding time-series are reported in Fig. 5(a): as expected, they roughly follow the same patterns, with an increasing volume of traffic early mornings, followed by a smooth lull during afternoons and nights. To evaluate the overall variation of the censorship activity, we show in Fig. 5(b) the temporal evolution of the number of censored (resp., allowed) requests at specific times of the day, *normalized* by the total number of censored (resp., allowed) requests. Note that the two curves are not comparable, but illustrate the relative activity when considering the overall nature of the traffic over the observation period. The relative



**Figure 5:** Censored and allowed traffic over 5 days (absolute and normalized).

censorship activity exhibits a few peaks, with a higher volume of censored content on particular periods of time. There are also two sudden “drops” in both allowed and censored requests, which might be correlated to some protests that day.<sup>8</sup> There is a visible reduction in traffic from Thursday afternoon (August 4) to Friday (August 5), consistent with press reports of Internet connections being slowed almost every Friday “when the big weekly protests are staged” [20].



**Figure 6:** Relative Censored traffic Volume (RCV) for August 3 (in  $D_{sample}$ ) as a function of time.

**Heavy censoring activities.** To further study the activity peaks, we zoom in on one specific day (August 3) that has a particularly high volume of censored content. Let RCV (*Relative Censored traffic Volume*) be the ratio between the number of censored requests at a time frame (with a 5-minute granularity) and the total number of requests received on

<sup>8</sup>See <http://www.enduringamerica.com/home/2011/8/3/syria-and-beyond-liveblog-the-sights-and-sounds-of-protest.html>.



the same time frame; in Fig. 6, we plot RCV as a function of the time of day. There are a few sharp increases in the censorship activity, with the fraction of censored content increasing from 1% to 2% of the total traffic around 8am, while, around 9.30am, the RCV variation exhibits a sudden decay. A few other peaks are also observed early morning (5am) and evening (10pm).

We further investigate the main factors triggering the heavier censorship activities by analyzing the distribution of censored content between 8am and 9.30am on August 3. Table 6 shows the top-10 censored domains during this period and the adjacent ones, as well as the corresponding percentage of censored volume each domain represents.

6am - 8am		8am - 10am		10am -12pm	
Domain	%	Domain	%	Domain	%
metacafe.com	20.4%	skype.com	29.24%	facebook.com	22.47%
trafficholder.com	16.87%	facebook.com	19.45%	metacafe.com	18.56%
facebook.com	15.08%	live.com	9.59%	live.com	11.93%
google.com	8.15%	metacafe.com	7.59%	skype.com	11.79%
yahoo.com	6.43%	google.com	6.76%	google.com	6.81%
zynga.com	5.14%	yahoo.com	3.57%	zynga.com	3.43%
live.com	3.04%	wikimedia.org	2.47%	ceipmsn.com	2.38%
conduitapps.com	1.45%	trafficholder.com	2.06%	mtn.com.sy	2.13%
all4syria.info	1.44%	dailymotion.com	1.58%	panet.co.il	1.02%
hotsptshld.com	1.18%	conduitapps.com	1.11%	bbc.co.uk	0.91%

Table 6: Top censored domains, August 3, 6am-12pm.

It is evident that skype.com is being heavily blocked (up to 29% of the censored traffic), probably due to the protests that happened in Syria on August 3, 2011. However, we observe that 9% of the requests to Skype servers are related to update attempts (for Windows clients) and all of them are denied. There is also an unusually higher number of requests to MSN live messenger service (through msn.com), thus suggesting that the censorship activity peaks are correlated to high demand targeting Instant Messaging software websites.<sup>9</sup> In conclusion, we observe that the censorship peaks are mainly due to a sudden higher volume of traffic targeting Skype and MSN live messenger websites, which are being systematically censored by the proxies.

## 5.2 Comparing different proxies

Our datasets include data from seven proxy servers, thus, we decided to compare the behavior of the different proxies. Fig. 7(a) shows the traffic distribution across proxies, restricted to two days (August 3 and 4) for ease of presentation. The load is fairly distributed among the proxies. However, when only considering censored traffic (Fig. 7(b)), we observe different behaviors. In particular, Proxy SG-48 is responsible for a large proportion of the censored traffic, especially at certain times. One possible explanation is that different proxies follow different policies, or there could be a high proportion of censored (or likely to be censored) traffic being redirected to proxy SG-48 during one specific period of time.

<sup>9</sup>Very similar results are present also for other periods of censorship activity peaks.

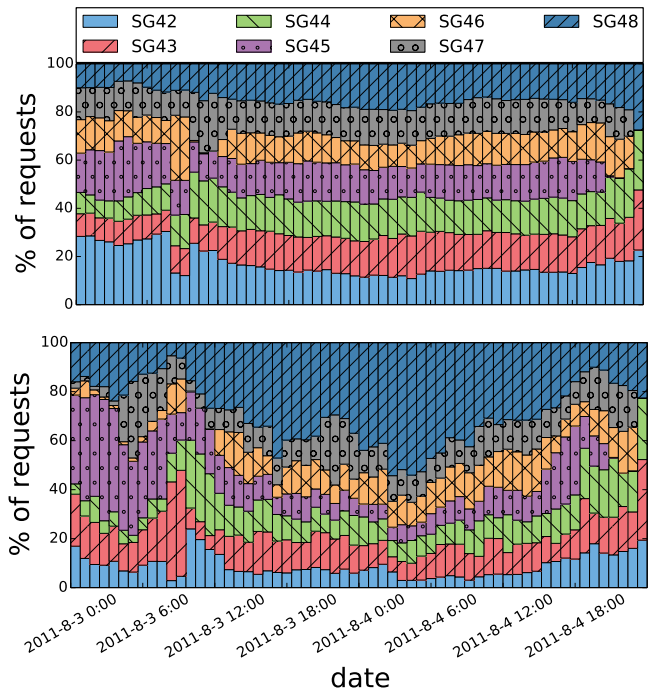


Figure 7: The distribution of traffic load through each proxy and censored traffic over time.

We also consider the top-10 censored domain names in the period of time August 3 (12am)–August 4 (12am) and observe that the domain metacafe.com is *always* censored and that almost all related requests (more than 95%) are processed *only* by proxy SG-48. This might be due to a domain-based traffic redirection process: in fact, we observed a very similar behavior for skype.com during the censorship peaks analysis presented earlier in Section 5.1.

In order to verify our hypotheses, we evaluate the similarity between censored requests handled by each proxy. We do so by relying on the Cosine Similarity, defined as

$$\text{cosine\_similarity}(A, B) = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}}$$

where  $A_i$  and  $B_i$  denote the number of requests for domain  $i$  censored by proxies  $A$  and  $B$ , respectively. Note that *cosine\_similarity* lies in the range  $[-1, 1]$ , with  $-1$  indicating patterns that are not at all similar, and  $1$  indicating a perfect match. We report the values of *cosine\_similarity* between the different proxies in Table 7. A few proxies exhibit high similarity, while others very low. This suggests that a few proxies are “specialized” in censoring specific types of content.

We also look at the categories distribution of all requests across the different proxies and concentrate on two categories, “Unavailable” and “None”, which show a peculiar distribution across the proxies (recall that categories have been discussed in Section 4). We note that the “None” category is *only* observed on two different proxies (SG-43 and SG-48), while “Unavailable” is less frequently observed on these two. This

	SG-42	SG-43	SG-44	SG-45	SG-46	SG-47	SG-48
SG-42	1.0	0.5944	0.5424	0.3905	0.6134	0.2921	0.0896
SG-43	0.5944	1.0	0.8226	0.4769	0.821	0.3138	0.0696
SG-44	0.5424	0.8226	1.0	0.6177	0.8757	0.3003	0.0721
SG-45	0.3905	0.4769	0.6177	1.0	0.4752	0.2316	0.6701
SG-46	0.6134	0.821	0.8757	0.4752	1.0	0.3294	0.066
SG-47	0.2921	0.3138	0.3003	0.2316	0.3294	1.0	0.0455
SG-48	0.0896	0.0696	0.0721	0.6701	0.066	0.0455	1.0

**Table 7:** Cross correlation of censored domains: Cosine similarity between different proxy servers (day: 2011-08-03).

<i>cs_host</i>	# requests
upload.youtube.com	86.97%
www.facebook.com	10.69%
ar-ar.facebook.com	1.76%
competition.mbc.net	0.33%
sharek.aljazeera.net	0.29%

**Table 8:** Top-5 hosts for *policy\_redirect* requests in  $D_{full}$ .

suggests either different configuration of the proxies or a content specialization of the proxies.

### 5.3 Denied vs. Redirected Traffic

According to our study, requests are censored in one of two ways: the request is either *denied* or *redirected*. Whenever a request triggers a *policy\_denied* exception, the requested page is not served to the client. Upon triggering *policy\_redirect*, the request is redirected to another URL. For these requests, we only have information from the *x-exception-id* field (set to *policy\_redirect*) and the *s-action* field (set to *tcp\_policy\_redirect*). The *policy\_redirect* exception is raised for a small number of hosts – 11 in total. As reported in Table 8, the most common URLs are upload.youtube.com and Facebook-owned domains.

Note that the redirection should trigger an additional request from the client to the redirected URL immediately after *policy\_redirect* is raised. However, we found no trace of a secondary request coming right after (within a 2-second window). Thus, we conclude that the secondary URL is either hosted on a website that does not require to go through the filtering proxies (most likely, this site is hosted in Syria) or that the request is processed by proxies other than those in the dataset. Since the destination of the redirection remains unknown, we do not know whether or not redirections point to different pages, depending on the censored request.

### 5.4 Category, String, and IP based Censorship

We now study the three main triggers of censorship decisions: URL categories, strings, and IP addresses.

**Category-based Filtering.** According to Blue Coat’s documentation [5], proxies can associate a category to each request (in the *cs-categories* field), based on the corresponding URL, and this category can be used in the filtering rules. In the set of censored requests, we identify only two categories: a default category (named “unavailable” or “none”, depending

on the proxy server), and a custom category (named “Blocked sites; unavailable” or “Blocked sites”). The custom category targets specific Facebook pages with a *policy\_redirect* policy, accounts for 1,924 requests, and is discussed in detail in Section 6. All the other URLs (allowed or denied) are categorized to the default category, which is subject to a more general censorship policy, and captures the vast majority of the censored requests. The censored requests in the default category consist mostly of *policy\_denied* with a small portion (0.21% either PROXIED or DENIED) of *policy\_redirect* exceptions. We next investigate the policy applied within the default category.

**String-based Filtering.** The filtering process is also based on particular strings included in the requested URL. In fact, the string-based filtering only relies on URL-related fields, specifically *cs-host*, *cs-uri-path* and *cs-uri-query*, which fully characterize the request. The proxies’ filtering process is performed using a simple string-matching engine that detects any blacklisted substring in the URL.

We now aim to recover the list of strings that have been used to filter requests in our dataset. We expect that a string used for censorship should only be found in the set of censored requests and never in the set of allowed ones (for this purpose, we consider PROXIED requests separately from OBSERVED requests, since they do not necessarily indicate an allowed request, even when no exception is logged). In order to identify these strings, we use the following iterative approach:

1. Let  $\mathcal{C}$  be the set of censored URLs and  $\mathcal{A}$  the set of allowed URLs.
2. Manually identify a string  $w$  appearing frequently in  $\mathcal{C}$ ;
3. Let  $N_{\mathcal{C}}$  and  $N_{\mathcal{A}}$  be the number of occurrences of  $w$  in  $\mathcal{C}$  and  $\mathcal{A}$ , respectively.
4. If  $N_{\mathcal{C}} \gg 1$  and  $N_{\mathcal{A}} = 0$  then remove from  $\mathcal{C}$  all requests containing  $w$ , add  $w$  to the list of censored strings, and go to step 2.

The manual string identification (in step 2) poses some non-trivial challenges: to mitigate selection of strings that are unrelated to the censorship decision, we took a conservative approach by considering non-ambiguous requests. For instance, we select simple requests, e.g., HTTP GET `new-syria.com/`, which only contains a domain name and has an empty path and an empty query field. Thus, we are sure that the string `new-syria.com` is the source of the censorship.

**URL-based Filtering.** Using the iterative process described above, we identify a list of 105 “suspected” domains, for which no request is allowed. Table 9 presents the top-10 domains in the list, according to the number of censored requests. We further categorized each domain in the list and show in Table 10 the top-10 categories according to the number of censored requests. Clearly, there is a heavy

Domain	Censored		Allowed		Proxied	
skype.com	23,558	(8.32%)	0	(0.00%)	39	(0.03%)
metacafe.com	19,257	(6.80%)	0	(0.00%)	49	(0.03%)
wikimedia.org	13,506	(4.77%)	0	(0.00%)	143	(0.09%)
.il	2,609	(0.92%)	0	(0.00%)	370	(0.24%)
amazon.com	2,356	(0.83%)	0	(0.00%)	13	0.01%
aawsat.com	2,180	(0.77%)	0	(0.00%)	230	(0.15%)
jumblo.com	1,158	(0.41%)	0	(0.00%)	0	(0.00%)
jeddahbikers.com	907	(0.32%)	0	(0.00%)	5	(0.00%)
islamway.com	702	(0.25%)	0	(0.00%)	16	(0.01%)
badoo.com	614	(0.22%)	0	(0.00%)	25	(0.02%)

**Table 9:** Top-10 domains suspected to be censored (number of requests and fraction for each class of traffic in  $D_{sample}$ ).

Category (#domains)	Censored requests	
Instant Messaging (2)	47,116	(16.63%)
Streaming Media (6)	39,282	(13.87%)
Education/Reference (4)	27,106	(9.57%)
General News (62)	8,700	(3.07%)
NA (42)	6,776	(2.39%)
Online Shopping (2)	4,712	(1.66%)
Internet Services (6)	2,964	(1.05%)
Social Networking (6)	2,114	(0.75%)
Entertainment (4)	1,828	(0.65%)
Forum/Bulletin Boards (8)	1,606	(0.57%)

**Table 10:** Top-10 domain categories censored by URL (number of censored requests and fraction of censored traffic in  $D_{sample}$ ).

Keyword	Censored		Allowed		Proxied	
proxy	194,539	(68.68%)	0	(0.00%)	1,106	(0.73%)
hotspotshield	5,846	(2.06%)	0	(0.00%)	24	(0.02%)
ultrareach	2,290	(0.81%)	0	(0.00%)	436	(0.29%)
israel	2,267	(0.80%)	0	(0.00%)	25	(0.02%)
ultrasurf	2,073	(0.73%)	0	(0.00%)	468	(0.31%)

**Table 11:** The list of 5 keywords identified as censored (fraction and number of requests for each class of traffic in  $D_{sample}$ ).

copyright of Instant Messaging software, as well as news, public forums, and user-contributed streaming media sites.

**Keyword-based Filtering.** We also identify five keywords that trigger censorship when found in the URL (*cs-host*, *cs-path* and *cs-query* fields): *proxy*, *hotspotshield*, *ultrareach*, *israel*, and *ultrasurf*. We report the corresponding number of censored, allowed, and proxied requests in Table 11. Four of them are related to anti-censorship technologies and one refers to Israel. Note that a large number of requests containing the keyword *proxy* are actually related to seemingly “non sensitive” content, e.g., online ads content, tracking components or online APIs, but are nonetheless blocked. For instance, the Google toolbar API invokes a call to `/tbproxy/af/query`, which can be found on the `google.com` domain, and is unrelated to anti-censorship software. Nevertheless, this element accounts for 4.85% of the *censored* requests in the  $D_{sample}$  dataset. Likewise, the keyword *proxy* is also included in some online social networks’ advertising components (see Section 6).

**IP-based censorship.** We now focus on understanding whether some requests are censored based on IP address. To this end, we look at the requests for which the *cs-host* field is an IPv4 address and notice that some of the URLs of censored requests do not contain any meaningful information except for the IP address. As previously noted, censorship can

Country	Censorship Ratio (%)	# Censored	# Allowed
Israel	6.69	5,191	72,416
Kuwait	2.02	16	776
Russian Federation	0.64	959	149,161
United Kingdom	0.26	2,490	942,387
Netherlands	0.17	12,206	7,077,371
Singapore	0.13	19	14,768
Bulgaria	0.09	14	14,786

**Table 12:** Censorship ratio for top censored countries in  $D_{IPv4}$ .

Subnet	Censored		Allowed		Proxied	
	# req.	# IPs	# req.	# IPs	# req.	# IPs
84.229.0.0/16	574	198	0	0	4	4
46.120.0.0/15	571	11	5	1	0	0
89.138.0.0/15	487	148	1	1	3	3
212.235.64.0/19	474	5	325	1	0	0
212.150.0.0/16	471	3	6,366	12	1	1

**Table 13:** Top censored Israeli subnets.

be done at a country level, e.g., for Israel, as all `.il` domains are blocked. Thus, we consider the possibility of filtering traffic with destination in some specific geographical regions, based on the IP address of the destination host.

We construct  $D_{IPv4}$ , which includes the set of requests (from  $D_{full}$ ) for which the *cs-host* field is an IPv4 address. We geo-localize each IP address in  $D_{IPv4}$  using the Maxmind GeoIP database.<sup>10</sup> We then introduce, for each identified country, the corresponding censorship ratio, i.e., the number of censored requests over the total number of requests to this country. Table 12 presents the censorship ratio for each country in  $D_{IPv4}$ . Israel is by far the country with the highest censorship ratio, suggesting that it might be subject to an IP-based censorship.

Next, we focus on Israel and zoom in to the subnet level.<sup>11</sup> Table 13 presents, for each of the top censored Israeli subnets, the number of requests and IP addresses that are censored and allowed. We identify two distinct groups: subnets that are almost always censored (except for a few exceptions of allowed requests), e.g., 84.229.0.0/16, and those that are either censored or allowed but for which the number of allowed requests is significantly larger than that of the censored ones, e.g. 212.150.0.0/16. One possible reason for a systematic subnet censorship could be related to blacklisted keywords. However, this is not the case in our analysis since the requested URL is often limited to a single IP address (*cs-uri-path* and *cs-uri-query* fields are empty). We further check, using McAfee smart filter, that none but one (out of 1155 IP addresses) of the censored Israeli IP addresses are categorized as Anonymizer hosts. These results show then that IP filtering is targeting a few geographical areas, and in particular Israeli subnets.

## 5.5 Summary

The analysis presented in this section has shown evidence

<sup>10</sup><http://www.maxmind.com/en/country>

<sup>11</sup>The list of IPv4 subnets corresponding to Israel is available from <http://www.ip2location.com/free/visitor-blocker>.

Social network	# Censored	# Allowed	# Proxied
facebook.com	68,782 (24.28%)	769,555 (2.55%)	3,942 (2.60%)
badoo.com	614 (0.22%)	0 (0.00%)	25 (0.02%)
netlog.com	438 (0.15%)	0 (0.00%)	100 (0.07%)
linkedin.com	308 (0.11%)	7,019 (0.02%)	75 (0.05%)
hi5.com	124 (0.04%)	9,301 (0.03%)	20 (0.01%)
skyrock.com	117 (0.04%)	270 (0.00%)	3 (0.00%)
twitter.com	7 (0.00%)	115,502 (0.38%)	585 (0.39%)
livejournal.com	1 (0.00%)	818 (0.00%)	0 (0.00%)
ning.com	1 (0.00%)	1,886 (0.01%)	5 (0.00%)
last.fm	0 (0.00%)	1,777 (0.01%)	1 (0.00%)

**Table 14:** Top-10 censored social networks in  $D_{sample}$  (fraction and number of requests for each class of traffic).

of domain-based traffic redirection between proxies. A few proxies seem to be specialized in censoring specific domains and type of content. Also, our findings suggest that the censorship activity reaches peaks mainly because of unusually high demand for Instant Messaging Software websites (e.g., Skype), which are blocked in Syria. Moreover, we found that censorship is based on four main criteria: URL-based filtering, keyword-based filtering, destination IP address, and a custom category-based censorship (further discussed in the next section). The list of blocked keywords and domains demonstrates the intent of Syrian censors to block political and news content, video sharing, and proxy-based censorship-circumvention technologies. Finally, Israeli-related content is heavily censored as the keyword *Israel*, the .il domain, and some Israeli subnets are blocked.

## 6. CENSORSHIP OF SOCIAL MEDIA

In this section, we analyze the filtering and censorship of Online Social Networks (OSNs) in Syria. Social media have often been targeted by censors, e.g., during the recent uprisings in the Middle East and North Africa. In Syria, according to our logs, popular OSNs like Facebook and Twitter are not *entirely* censored and most traffic is allowed. However, we observe that a few specific keywords (e.g., *proxy*) and a few pages (e.g., the ‘Syrian Revolution’ Facebook page) are blocked, thus suggesting a targeted censorship.

### 6.1 Overview

We select a representative set of social networks containing the top 25 social networks according to alexa.com as of November 2013, and add 3 social networks popular in Arabic-speaking countries: netlog.com, salamworld.com, and muslimup.com. For each of these sites, we extract the number of allowed, censored and proxied requests in  $D_{sample}$ , and report the top-10 censored social networks in Table 14.

We find no evidence of systematic censorship for most sites (including last.fm, MySpace, Google+, Instagram, and Tumblr), as all requests are allowed. However, for a few social networks (including Facebook, LinkedIn, Twitter, and Flickr) many requests are blocked. We observe that several requests are censored based on blacklisted keywords (e.g., *proxy*, *Israel*), thus suggesting that the destination domain is not the actual reason of censorship. However, requests to Netlog and Badoo are never allowed and there is only a

minority of requests containing blacklisted keywords, which suggests that these domains are always censored. In fact, both netlog.com and badoo.com were identified in the list of domains suspected for URL-based filtering, described in Section 5.4.

### 6.2 Facebook

Recall that the majority of requests to Facebook are allowed, yet facebook.com is one of the most censored domains. As we explain below, censored requests can be classified into two groups: (i) requests to Facebook pages with sensitive (political) content, and (ii) requests to the social platform with the blacklisted keyword *proxy*.

**Censored Facebook pages.** Several Facebook pages are censored for political reasons and are identified by the proxies using the custom category “Blocked Sites.” Requests to those pages trigger a *policy\_redirect* exception, thus redirecting the user to a page unknown to us. Interestingly, Reporters Without Borders [20] stated that “[t]he government’s cyber-army, which tracks dissidents on online social networks, seems to have stepped up its activities since June 2011. Web pages that support the demonstrations were flooded with pro-Assad messages.” While we cannot infer the destination of redirection, we argue that this mechanism could technically serve as a way to show specific content addressing users who access targeted Facebook pages.

Table 15 lists the Facebook pages we identify in the logs that fall into the custom category. All the requests identified as belonging to the custom category are censored. However, we find that not all requests to the facebook.com/(censored\_page) pages are correctly categorized as “Blocked Site.” For instance, www.facebook.com/Syrian.Revolution?ref=ts is, but http://www.facebook.com/Syrian.Revolution?ref=ts&...a=11&ajaxpipe=1&quickling[version]=414343%3B0 is not, thus suggesting that the categorization rules targeted a very narrow range of specific *cs-uri-path* and *cs-uri-query* combinations. As can be seen in Table 15, many requests to the targeted Facebook pages are allowed; none of the allowed requests is categorized as “Blocked Site.” We also identify successful requests sent to Facebook pages such as Syrian.Revolution.Army, Syrian.Revolution.Assad, Syrian.Revolution.Caricature and ShaamNewsNetwork, which are not categorized as “Blocked Site” and are allowed. Finally, we note that the proxied requests are sometimes categorized as “Blocked Site” (e.g., all the requests for the Syrian.revolution.page) and sometimes not.

**Social plugins.** Facebook provides so-called social plugins (one common example is the *Like* button), which can be loaded into web pages to enable interaction with the social platform. Some of the URLs in which these social plugins are placed include the keyword *proxy* in the *cs-uri-path* field or in the *cs-uri-query* field, and this automatically raises the *policy\_denied* exception whenever the page is loaded.

Table 16 reports, for each of the top-16 social plugin elements, the fraction of the Facebook traffic and the num-

Facebook page	# Censored	# Allowed	# Proxied
Syrian.Revolution	1461	891	16
Syrian.revolution	0	0	25
syria.news.F.N.N	191	165	1
ShaamNews	114	3944	7
fffm14	42	18	0
barada.channel	25	9	0
DaysOfRage	19	2	0
Syrian.R.V	10	6	0
YouthFreeSyria	6	0	0
sooryoon	3	0	0
Freedom.Of.Syria	3	0	0
SyrianDayOfRage	1	0	0

Table 15: Top Facebook pages of the “Blocked Site” category in  $D_{full}$ .

Social plug-in	Censored	Allowed	Proxied
/plugins/like.php	29,456 (42.83%)	35,011 (4.55%)	351 (8.90%)
/extern/login_status.php	26,865 (39.06%)	2,402 (0.31%)	142 (3.60%)
/plugins/likebox.php	3,223 (4.69%)	13,011 (1.69%)	121 (3.07%)
/plugins/send.php	2,994 (4.35%)	85 (0.01%)	9 (0.23%)
/plugins/comments.php	2,317 (3.37%)	197 (0.03%)	14 (0.36%)
/connect/canvas_proxy.php	1,866 (2.71%)	0 (0.00%)	3 (0.08%)
/fbml/fbjs_ajax_proxy.php	1,760 (2.56%)	0 (0.00%)	5 (0.13%)
/platform/page_proxy.php	80 (0.12%)	0 (0.00%)	0 (0.00%)
/ajax/proxy.php	60 (0.09%)	0 (0.00%)	2 (0.05%)
/plugins/facepile.php	30 (0.04%)	34 (0.00%)	0 (0.00%)
/common/scribe_endpoint.php	19 (0.03%)	679 (0.09%)	0 (0.00%)
/dialog/oauth	9 (0.01%)	28 (0.00%)	0 (0.00%)
/plugins/registration.php	6 (0.01%)	2 (0.00%)	0 (0.00%)
/plugins/login.php	3 (0.00%)	0 (0.00%)	0 (0.00%)
/WorkingProxy	3 (0.00%)	0 (0.00%)	0 (0.00%)
/plugins/serverfbml.php	2 (0.00%)	19 (0.00%)	0 (0.00%)

Table 16: Top-16 Facebook social plugin elements in  $D_{sample}$  (fraction of Facebook traffic and number of requests).

ber of requests for each class of traffic. The top two censored social plugin elements (*/plugins/like.php* and */extern/login\_status.php*) account for more than 80% of the censored traffic on the facebook.com domain, while the 16 social plugin elements we consider account for 99.87% (68,693) of the censored requests on the facebook.com domain.

To conclude, the large number of censored requests on the facebook.com domain is in fact mainly caused by social plugins elements that are not related with censorship circumvention tools or any political content.

### 6.3 Summary

We have studied the censorship of 28 major online social networks and found that most of them are not censored, unless requests contain blacklisted keywords (such as *proxy*) in the URL. This is particularly evident looking at the large amount of Facebook requests that are censored due to the presence of *proxy* in the query. Using a custom category, the censors also target a selected number of Facebook pages, without blocking all traffic to the site, thus making censorship and surveillance harder to detect (as independently reported in [18]).

## 7. ANTI-CENSORSHIP TECHNOLOGY IN USE

In this section, we investigate the usage and the effectiveness of censorship-circumvention technologies in Syria based on our dataset.

### 7.1 Tor

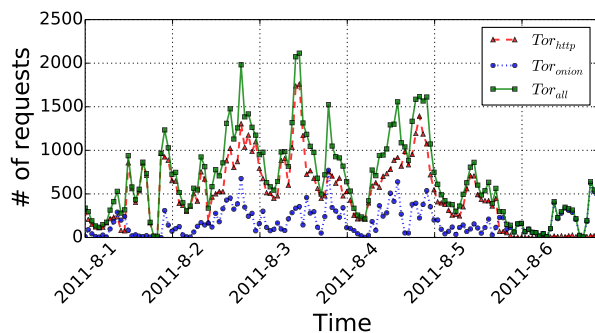


Figure 8: Number of Tor related requests per hour from August 1-6 in  $D_{full}$ .

Tor [10] is a relay network based on onion-routing that allows users to circumvent Internet censorship while providing strong anonymity. According to our logs, access to the Tor project website<sup>12</sup> and the majority of Tor traffic were allowed in July/August 2011. In fact, access to the Tor network was first reported to be blocked on December 16, 2012 [23].

Tor traffic can be classified into two main classes: (1) HTTP signaling traffic, aiming to fetch and establish connections with Tor directories (which we denote as  $Tor_{http}$ ), and (2) traffic related to establishing Tor circuits and data transfer (which we denote as  $Tor_{onion}$ ). To identify Tor traffic, we extract Tor relays’ IP addresses and port numbers from the Tor server descriptors and network status files, publicly available from <https://metrics.torproject.org/formats.html>. Next, we match the extracted  $\langle \text{node IP, port, date} \rangle$  triplets to the requests in  $D_{full}$  to identify Tor traffic. We further isolate Tor HTTP signaling messages by identifying all HTTP requests to Tor directories, e.g., */tor/server/authority.z or /tor/keys*.<sup>13</sup>

Observe that this method does not take into account the connections via Tor bridges, since there is no public list of them (in fact, bridges were introduced to overcome filtering of connections to known Tor relays, so they are not listed in the main Tor directory). However, as discussed later in this section, as of 2011, Tor relays were not filtered in Syria, thus suggesting that users did not actually need to use bridges at the time.

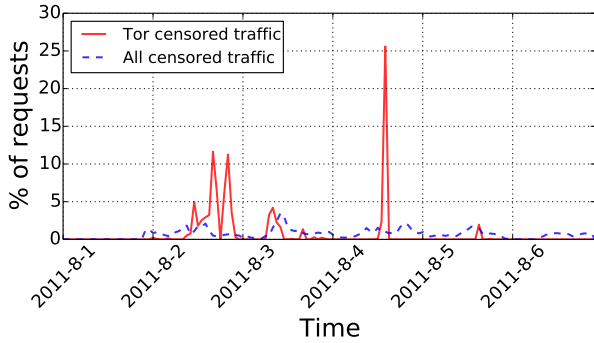
We identify about 95K requests to 1,111 different Tor relays, 73% of which belong to  $Tor_{http}$ . Only a small fraction (1.38%) of the requests are censored and 16.2% of them generate TCP errors. Figure 8 shows the number of requests for a period of six days (August 1-6). The traffic has several peaks, in particular on August 3, when several protests were taking place.

We take a closer look at censored Tor traffic, and observe that 99.9% of it is blocked by a single proxy (SG-44), even though the overall traffic is uniformly distributed across all

<sup>12</sup><http://www.torproject.org>

<sup>13</sup>See [https://gitweb.torproject.org/torspec.git?a=blob\\_plain;hb=HEAD;f=dir-spec-v2.txt](https://gitweb.torproject.org/torspec.git?a=blob_plain;hb=HEAD;f=dir-spec-v2.txt) for a full description.





**Figure 9:** Percentage of all censored traffic and Tor censored traffic by Proxy SG-44.

the 7 proxies. (the other 0.01% of the traffic is censored by SG-48). This finding is in line with our earlier discussion on the specialization of some proxies. We also analyze the temporal pattern of Tor censored traffic and compare it to the overall censored requests of SG-44. As shown in Figure 9, Tor censoring exhibits much more variance compared to the overall censored traffic.

While censoring  $Tor_{http}$  is technically simple, as it only requires matching of regular expressions against the HTTP requests, identifying and censoring  $Tor_{onion}$  is more challenging, as it involves encrypted traffic. However, only  $Tor_{onion}$  traffic is censored according to the logs, while  $Tor_{http}$  is always allowed.

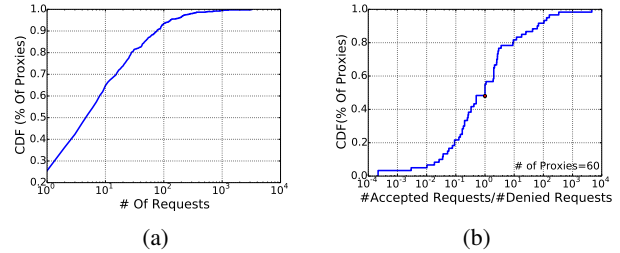
**Note:** We also observed some inconsistencies between proxies concerning the blocking of a few Tor relays. Due to space limitation, we omit further analysis from this version of the paper.

## 7.2 Web Proxies and Virtual Private Networks (VPNs)

As we have already observed, access to web/socks proxies is censored, as demonstrated by an aggressive filtering of requests including the keyword *proxy*. In the following, we use “proxies” to refer to the Blue Coat appliances whose logs we study in this paper, and “web proxies” to refer to services used to circumvent censorship.

To use web proxies, end-users need to configure their browsers or their network interfaces, or rely on tools (e.g., Ultrasurf) that automatically redirect all HTTP traffic to the web proxy. Some web proxies support encryption, and create a SSL-based encrypted HTTP tunnel between the user and the web proxy. Similarly, VPN tools (e.g., Hotspot Shield) are often used to circumvent censorship, again, by relaying traffic through a VPN server.

We analyze the usage of VPN and web proxy tools and find that a few of them are very popular among Internet users in Syria. However, as discussed in Section 5.4, keywords like *ultrasurf* and *hotspotshield* are heavily monitored and censored. Nonetheless, some web proxies and VPN software (such as Freegate, GTunnel and GPass) do not include the



**Figure 10:** (a) CDF of the number of requests for identified “anonymizer” hosts; (b) Ratio of Allowed versus Censored number of requests for identified “anonymizer” hosts.

keyword *proxy* in request URLs and are therefore not censored. Similarly, we do not observe any censorship activity triggered by the keyword *VPN*.

Next, we focus on the domains that are categorized as “Anonymizers” by McAfee’s TrustedSource tool, including both web proxies and VPN-related hosts. In the  $D_{sample}$  dataset, there are 821 “Anonymizer” domains, which are the target of 122K requests (representing 0.4% of the total number of requests). 92.7% of these hosts (accounting for 25% of the requests) are never filtered. Figure 10(a) shows the CDF of the number of requests sent to each of those allowed hosts. Less than 10% of these hosts receive more than 100 requests, suggesting that only a few popular services attract a high number of the requests.

Finally, we look at the 7.3% of the identified “Anonymizer” hosts, for which some of the requests are censored. We calculate the ratio between the number of allowed requests in  $D_{full}$  and the number of censored requests in  $D_{denied}$ . Figure 10(b) shows the CDF of this censorship ratio. We observe a non-consistent policy for whether a request is allowed or censored, with more than 50% of the proxies showing a higher number of allowed requests than censored requests. This suggests that these requests are not censored based on their IP or hostname, but rather on other criteria, e.g., the inclusion of a blacklisted keyword in the request.

In conclusion, while some services (such as Hotspot Shield) are heavily censored, other, less known, services are not, unless related requests contain blacklisted keywords. This somehow introduces a trade-off between the ability to bypass censorship and promoting censorship- and surveillance-evading tools (e.g., in web searches) by including keywords such as *proxy* in the URL.

## 7.3 Peer-to-Peer networks

The distributed architecture of peer-to-peer networks makes them, by nature, more resilient to censorship: users obtain content from peers and not from a central server, which makes it harder to locate and block content. Shared data is usually identified by a unique identifier (e.g., info hash in BitTorrent), and these identifiers are useless to censors unless mapped back to, e.g., the description of the corre-

sponding files. However, resolving these identifiers is not trivial: content can be created by anyone at any time, and the real description can be distributed in many different ways, publicly and privately.

To investigate the use of peer-to-peer networks as a way to access censored content, we look for signs of BitTorrent traffic in the logs. We find a total of 338,168 BitTorrent announce requests from 38,575 users (identified by peerID) for 35,331 unique contents in the  $D_{full}$  dataset.<sup>14</sup> Most of these requests (99.97%) are allowed. Censored requests can be explained, once again, with the occurrence of blacklisted keywords, e.g., *proxy*, in the request URL. For instance, all announce requests sent to the tracker on *tracker-proxy.furk.net* are censored.

Using the hashes of torrent files provided in the announce messages, we crawl *torrentz.eu* and *torrentproject.com* to extract the titles of these torrent files, achieving a success rate of 77.4%. The five blacklisted keywords, reported in Table 11, are actually present in the titles of some of the BitTorrent files, yet the associated announce requests are allowed. While we do not find any content that can be directly associated with sensitive topics like “Syrian revolution” or “Arab spring” (these files may still be shared via BitTorrent without publicly announcing the content), we do identify content titles which relate to anti-censorship software, such as UltraSurf (2,703 requests for all versions), HideMyAss (176 requests), Auto Hide IP (532 requests) and anonymous browsers (393 requests). Our findings suggest that peer-to-peer networks are indeed used by users inside Syria to circumvent censorship to a certain extent. Also note that BitTorrent is used to download Instant Messaging software, such as Skype, MSN messenger, and Yahoo! Messenger, which cannot be downloaded directly from the official download pages due to censorship.

## 7.4 Google Cache

When searching for terms in Google’s search engine, the result pages allow access to cached versions of suggested pages. While Google Cache is not intended as an anti-censorship tool, a simple analysis of the logs shows that it provides a way to access content that is otherwise censored.

We identify a total of 4,860 requests accessing Google’s cache on *webcache.googleusercontent.com* in the  $D_{full}$  dataset. Only 12 of them are censored due to an occurrence of a blacklisted keyword in the URL, and a single request to retrieve a cached version of *http://ar-ar.facebook.com/SYRIANREVOLUTION.K.N.N* has *policy\_denied*, although it is not categorized as a “Blocked Site.” However, the rest of the requests are allowed. Interestingly, some of the allowed requests, although small in number, relate to cached versions of webpages that are otherwise censored, such as *www.panet.co.il*, *aawsat.com*, *www.facebook.com/Syrian.Revolution*, and *www.free-syria.com*.

While the use of Google cache to access censored content

<sup>14</sup>BitTorrent clients send announce requests to BitTorrent servers (aka trackers) to retrieve a list of IP addresses from which the requested content can be downloaded.

is obviously limited in scope, the logs actually suggest that it is very effective. Thus, when properly secured with HTTPS, Google cache could serve as a way to access censored content.

## 7.5 Summary

The logs highlighted that Syrian users do resort to censorship circumvention tools, with a relatively high effectiveness. While some tools and websites are monitored and blocked (e.g., Hotspot Shield), many others are successful in bypassing censorship. Our study also shows that some tools that were not necessarily designed as circumvention tools, such as BitTorrent and Google cache, could provide additional ways to access censored content if proper precautions are taken, especially considering that Syrian ISPs started to block Tor relays and bridges in December 2012.

## 8. DISCUSSION

**Economics of Censorship.** Our analysis shows that Syrian authorities deploy several techniques to filter Internet traffic, ranging from blocking entire subnets to filtering based on specific keywords. This range of techniques can be explained by the cost/benefit tradeoff of censorship, as described by Danezis and Anderson [9]. In particular, while censoring the vast majority of the Israeli network – regardless of the actual content – can be explained on geo-political grounds, completely denying the access to social networks, such as Facebook, could generate unrest. For instance, facing the “Arab spring” uprisings, the Syrian authorities decided to allow access to Facebook, Twitter, and Youtube in February 2011. Nonetheless, these websites are monitored and selectively censored. Our analysis shows that censorship aims at a more subtle control of the Internet, by only denying access to a predefined set of websites, as well as a set of keywords. This shift is achievable as the proxy appliances seamlessly support Deep Packet Inspection (DPI), thus allowing fine-grained censorship in real-time.

**Censorship’s target.** Censored traffic encompasses a large variety of content, mostly aiming to prevent users from using Instant Messaging software (e.g., Skype), video sharing websites (e.g., *metacafe.com*, *upload.youtube.com*), Wikipedia, as well as sites related to news and opposition parties (e.g., *islammemo.cc*, *alquds.co.uk*). Censors also deliberately block any requests related to a set of predefined anti-censorship tools (e.g., ‘proxy’). This mechanism, however, has several side effects as it denies the access to any page containing these keywords, including those that have nothing to do with censorship circumvention.

**Censorship Circumvention.** Our study has also shown that users do take actions to circumvent censorship. One interesting way is using BitTorrent to download anti-censorship tools such as UltraSurf as well as Instant Messaging software. Users also rely on well-know techniques, such as web and socks proxies, and Tor.

## 9. CONCLUSION

This paper presented a large-scale measurement analysis of Internet censorship in Syria. We analyzed 600GB worth of logs extracted from 7 Blue Coat SG-9000 filtering proxies, which are deployed to monitor, filter, and block traffic of Syrian users. By analyzing these logs, we provided a detailed analysis of how censorship was operated in Syria in 2011, a country that has been classified for several years as “Enemy of Internet” by Reporters Without Borders [20].

Our large-scale analysis of real-world logs allowed us to extract information about processed requests for *both* censored and allowed traffic and provide a detailed, first-of-a-kind snapshot of Syrian censorship practices. We uncovered the presence of a relatively stealthy yet quite targeted filtering, which operates, at the same time, relying on IP addresses to block access to entire subnets, on domains to block specific websites, and on keywords to target specific content. Instant Messaging software is heavily censored while filtering of social media is limited to specific pages. Finally, we showed that Syrian users try to evade censorship by using several tools, including web/socks proxies, Tor, VPNs, and BitTorrent.

## 10. REFERENCES

- [1] C. Anderson. The Hidden Internet of Iran: Private Address Allocations on a National Network. *arXiv preprint arXiv:1209.6398*, 2012.
- [2] S. Aryan, H. Aryan, and J. A. Halderman. Internet Censorship in Iran: A First Look. In *FOCI*, 2013.
- [3] D. Bamman, B. O’Connor, and N. Smith. Censorship and deletion practices in Chinese social media. *First Monday*, 17(3), 2012.
- [4] BBC News. Syrian jailed for internet usage. [http://news.bbc.co.uk/1/hi/world/middle\\_east/3824595.stm](http://news.bbc.co.uk/1/hi/world/middle_east/3824595.stm), 2004.
- [5] BlueCoat. Blue Coat Systems ProxySG – Configuration and Management Guide. [http://wikileaks.org/spyfiles/files/0/226.BLUECOAT-SGOS\\_CMG\\_4.1.4.pdf](http://wikileaks.org/spyfiles/files/0/226.BLUECOAT-SGOS_CMG_4.1.4.pdf), 2013.
- [6] J. R. Crandall, D. Zinn, M. Byrd, E. T. Barr, and R. East. ConceptDoppler: A Weather Tracker for Internet Censorship. In *CCS*, 2007.
- [7] A. Dainotti, C. Squarcella, E. Aben, K. C. Claffy, M. Chiesa, M. Russo, and A. Pescapé. Analysis of country-wide internet outages caused by censorship. In *IMC*, 2011.
- [8] J. Dalek, B. Haselton, H. Noman, A. Senft, M. Crete-Nishihata, P. Gill, and R. J. Deibert. A Method for Identifying and Confirming the Use of URL Filtering Products for Censorship. In *IMC*, 2013.
- [9] G. Danezis and R. Anderson. The Economics of Censorship Resistance. In *WEIS04*, 2004.
- [10] R. Dingleline, N. Mathewson, and P. Syverson. TOR: The second-generation onion router. In *USENIX Security*, 2004.
- [11] S. Egelman, J. Bonneau, S. Chiasson, D. Dittrich, and S. E. Schechter. It’s Not Stealing If You Need It: A Panel on the Ethics of Performing Research Using Public Data of Illicit Origin. In *Financial Cryptography*, 2012.
- [12] T. A. N. for Human Rights Information. Online Syria, Offline Syrians. <http://old.openarab.net/en/node/1625>, 2013.
- [13] G. King, J. Pan, and M. Roberts. How censorship in China allows government criticism but silences collective expression. In *APSA Annual Meeting*, 2012.
- [14] J. Knockel, J. R. Crandall, and J. Saia. Three Researchers, Five Conjectures: An Empirical Analysis of TOM-Skype Censorship and Surveillance. In *FOCI*, 2011.
- [15] C. S. Leberknight, M. Chiang, H. V. Poor, and F. Wong. A taxonomy of Internet censorship and anti-censorship. <http://www.princeton.edu/~chiangm/anticensorship.pdf>, 2012.
- [16] M. Marquis-Boire et al. Planet Blue Coat: Mapping Global Censorship and Surveillance Tools. <https://citizenlab.org/wp-content/uploads/2013/01/Planet-Blue-Coat.pdf>, 2013.
- [17] Z. Nabi. The Anatomy of Web Censorship in Pakistan. In *FOCI*, 2013.
- [18] Open Net Initiative. Internet Filtering in Syria. <https://opennet.net/research/profiles/syria>, 2009.
- [19] J. C. Park and J. R. Crandall. Empirical Study of a National-Scale Distributed Intrusion Detection System: Backbone-level Filtering of HTML Responses in China. In *ICDCS*, pages 315–326, 2010.
- [20] Reporters Without Borders. Syria. <http://en.rsf.org/syria-syria-12-03-2012,42053.html>, 2012.
- [21] V. Silver. H-P Computers Underpin Syria Surveillance. <http://www.bloomberg.com/news/2011-11-18/hewlett-packard-computers-underpin-syria-electronic-surveillance-project.html>, 2011.
- [22] Telecomix. #OpSyria: Syrian Censorship Logs (Season 3). <http://reflets.info/opsyria-syrian-censoship-log/>, 2011.
- [23] The Tor Project. Tor’s censorshipwiki – Syria. <https://trac.torproject.org/projects/tor/wiki/doc/OONI/censorshipwiki/CensorshipByCountry/Syria>, 2013.
- [24] J. Valentino-Devries, P. Sonne, and N. Malas. U.S. Firm Acknowledges Syria Uses Its Gear to Block Web. <http://online.wsj.com/news/articles/SB10001424052970203687504577001911398596328>, 2011.
- [25] J.-P. Verkamp and M. Gupta. Inferring Mechanics of Web Censorship Around the World. In *FOCI*, 2012.
- [26] VFM Systems & Services Ltd. Blue coat ProxySG Solution with Web Filter and Reporter. <http://www.slideshare.net/vfmindia/vfm-bluecoat-proxy-sg-solution-with-web-filter-and-reporter>, 2013.
- [27] P. Winter. Towards a Censorship Analyser for Tor. In *FOCI*, 2013.
- [28] P. Winter and S. Lindskog. How the Great Firewall of China is Blocking Tor. In *FOCI*, 2012.
- [29] X. Xu, Z. M. Mao, and J. A. Halderman. Internet censorship in China: Where does the filtering occur? In *PAM*, pages 133–142, 2011.
- [30] T.-F. Yen, Y. Xie, F. Yu, R. P. Yu, and M. Abadi. Host Fingerprinting and Tracking on the Web: Privacy and Security Implications. In *NDSS*, 2012.