Censorship in the Wild: Analyzing Internet Filtering in Syria

Abdelberi Chaabane  INRIA Rhône-Alpes  Montbonnot, France
Emiliano De Cristofaro  University College London  London, United Kingdom
Terence Chen  NICTA  Sydney, Australia
Arik Friedman  NICTA  Sydney, Australia
Mathieu Cunche  University of Lyon & INRIA  Lyon, France
Mohamed Ali Kaafar  NICTA & INRIA Rhône-Alpes  Sydney, Australia

ABSTRACT

Internet censorship is enforced by numerous governments worldwide, however, due to the lack of publicly available information, as well as the inherent risks of performing active measurements, it is often hard for the research community to investigate censorship practices in the wild. Thus, the leak of 600GB worth of logs from 7 Blue Coat SG-9000 proxies, deployed in Syria to filter Internet traffic at a country scale, represents a unique opportunity to provide a detailed snapshot of a real-world censorship ecosystem.

This paper presents the methodology and the results of a measurement analysis of the leaked Blue Coat logs, revealing a relatively stealthy, yet quite targeted, censorship. We find that traffic is filtered in several ways: using IP addresses and domain names to block subnets or websites, and keywords or categories to target specific content. We show that keyword-based censorship produces some collateral damage as many requests are blocked even if they do not relate to sensitive content. We also discover that 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 analytical look into Internet filtering in Syria.

Categories and Subject Descriptors

K.5.2 [Legal Aspects of Computing]: Governmental Issues—Censorship; K.4.1 [Computers and Society]: Public Policy Issues—Privacy

Keywords

Censorship; Internet Filtering; Measurements

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. Censors increasingly target access to, and dissemination of, electronic information, for instance, aiming to restrict freedom of speech, control knowledge available to the masses, or enforce religious/ethical principles.

Even though the research community dedicated a lot of attention to censorship and its circumvention, knowledge and understanding of filtering technologies is inherently limited, as it is challenging and risky to conduct measurements from countries operating censorship, while logs pertaining to filtered traffic are obviously hard to come by. Prior work has analyzed censorship practices in China [14, 15, 20, 29, 30], Iran [2, 4, 26], Pakistan [18], and a few Arab countries [10], however, mostly based on probing, i.e., inferring what information is being censored by generating requests and observing what content is blocked. While providing valuable insights, these methods suffer from two main limitations: (1) only a limited number of requests can be observed, thus providing a skewed representation of the censorship policies due to the inability to enumerate all censored keywords, and (2) it is hard to assess the actual extent of the censorship, e.g., what kind/proportion of the overall traffic is being censored.

Roadmap. In this paper, we present a measurement analysis of Internet filtering in Syria: we study a set of logs extracted from 7 Blue Coat SG-9000 proxies, which were deployed to monitor, filter and block traffic of Syrian users. The logs (600GB worth of data) were leaked by a “hacktivist” group, Telecomix, in October 2011, and relate to a period of 9 days between July and August 2011 [23]. By analyzing the logs, we provide a detailed snapshot of how censorship was operated in Syria. 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.

Main Findings. Our measurement-based analysis uncovers several interesting findings. First, we observe that a few different techniques are employed by Syrian authorities: IP-based filtering to block access to entire subnets (e.g., Israel), domain-based to block specific websites, keyword-based to target specific kinds of traffic (e.g., filtering-evading technologies, such as socks proxies), and category-based to target specific content and pages. As a side-effect of keyword-based censorship, many requests are blocked even if they do not relate to any sensitive content or anti-censorship technologies. Such collateral damage affects Google toolbar’s queries as well as a few ads delivery networks that are blocked as they generate requests containing the word proxy.

Also, the logs highlight that Instant Messaging software (like Skype) is heavily censored, while filtering of social media is limited to specific pages. Actually, most social networks (e.g., Facebook and Twitter) are not blocked, but only certain pages/groups are (e.g., the Syrian Revolution Facebook page). We find 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 (e.g., China and Iran), censorship in Syria seems to be less pervasive, yet quite targeted. Syrian censors particularly target Instant Messaging, information related to the political opposition (e.g., pages related to the “Syrian Revolution”), and Israeli subnets. 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, which seems to confirm reports about Syrian users engaging in self-censorship to avoid being arrested [3, 5, 19].

Contributions. To the best of our knowledge, we provide the first detailed snapshot of Internet filtering in Syria and the first set of large-scale measurements of actual filtering proxies’ logs. We show how censorship was operated via a statistical overview of censorship activities and an analysis of temporal patterns, proxy specializations, and filtering of social network sites. Finally, we provide some details on the usage of surveillance- and censorship-evading tools.

Remarks. 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 as censorship might have evolved in the last two years. According to Bloomberg [22], Syria has invested $500K in surveillance equipment in late 2011, thus an even more powerful filtering architecture might now be in place. Starting December 2012, Tor relays and bridges have reportedly been blocked [24]. Nonetheless, our analysis uncovers methods that might still be in place, e.g., based on Deep Packet Inspection. Also, Blue Coat proxies are reportedly still used in several countries [17].

Our work serves as a case study of censorship in practice: it provides a first-of-its-kind, data-driven analysis of a real-world censorship ecosystem, exposing its underlying techniques, as well as its strengths and weaknesses, which we hope will facilitate the design of 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

Due to the limited availability of publicly available data, there is little prior work analyzing logs from filtering devices. A fairly large body of work, which we overview in this section, has focused on understanding and characterizing censorship processes via probing. By contrast, our work is really the first to analyze traffic observed by actual filtering proxies and to provide a detailed measurement-based snapshot of Syria’s censorship infrastructure.

Ayran et al. [4] present measurements from an Iranian ISP analyzing HTTP host-based blocking, keyword filtering, DNS hijacking, and protocol-based throttling, and conclude that the censorship infrastructure heavily relies on centralized equipment. Winter and Lindskog [29] conduct some measurements on traffic routed through Tor bridges/relays to understand how China blocks Tor. Also, Dainotti et al. [9] analyze country-wide Internet outages, in Egypt and Libya, using publicly available data such as BGP inter-domain routing control plane data.

Another line of work deals with fingerprinting and inferring censorship methods and equipments. Researchers from the Citizen Lab [17] focus on censorship/performance served using Blue Coat devices and uncover 61 Blue Coat ProxySG devices and 316 Blue Coat PacketShaper appliances in 24 different countries. Dalek et al. [10] 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, UAE, Qatar, and Yemen.

Nabi [18] uses a publicly available list of blocked websites in Pakistan, checking 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. Furthermore, Verkamp and Gupta [26] detect censorship technologies in 11 countries, mostly using Planet Labs nodes, and discover DNS-based and router-based filtering. Crandall et al. [8] propose an architecture for maintaining a censorship “weather report” about what keywords are filtered over time, while Leberknight et al. [16] provide an overview of research on censorship resistant systems and a taxonomy of anti-censorship technologies.

Also, Knockel et al. [15] obtain a built-in list of censored keywords in China’s TOM-Skype and run experiments to understand how filtering is operated, while King et al. [14] devise a system to locate, download, and analyze the content of millions of Chinese social media posts, before the Chinese government censors them.

Finally, Park and Crandall [20] 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. [30] 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.

3. 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 “hacktivist” group called Telecomix announced the release of log files extracted from 7 Syrian Blue Coat SG-9000 proxies (aka ProxySG) [23]. The initial leak concerned 15 proxies but only data from 7 of them was publicly released. As reported by the Wall Street Journal [25] and CBS news [28], Blue Coat openly acknowledged that at least 13 of its proxies were used in Syria, but denied it authorized their sale to the Syrian government [6]. These devices have allegedly 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).

Given the nature of the dataset, one could question the authenticity of the logs. However, Blue Coat confirmed the use of its devices in Syria [25, 28] and a few findings emerging from the analysis of the logs actually correspond to events and facts that were independently reported before. Also, this leak is not the first censorship-related project carried out by Telecomix. Thus, we are confident that the datasets studied in this paper provide an accurate snapshot of Syrian censorship activities in Summer 2011.
3.2 Blue Coat SG-9000 Proxies

The Blue Coat SG-9000 proxies 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 one of the following three labels (as per 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 (might be redirected).

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** that there is no need to contact the OCS, and **PROXIED** that the outcome is in the proxy’s cache. According to Blue Coat’s documentation [27], 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, in the “bandwidth gain profile” (see page 193 in [1]).

3.3 Datasets and Notation

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

1. **Full Logs** ($D_{full}$): The whole dataset (i.e., extracted from all logs) is composed of 751,295,830 requests.

2. **Sample Dataset** ($D_{sample}$): 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 sometimes consider a random sample covering 4% of the entire dataset. This dataset ($D_{sample}$) is only used to illustrate a few results, specifically, for a few summary statistics. According to standard theory about confidence intervals for proportions (see [13], 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 (α = 0.05).

3. **User Dataset** ($D_{user}$): Before the data release, Telecomix suppressed user identifiers (IP addresses) by replacing them with zeros. However, for a small fraction of the data (July 22-23), user identifiers were replaced with the hash of the IP addresses, thus making user-based analysis possible.

4. **Denied Dataset** ($D_{denied}$): This dataset contains all the requests that resulted in exceptions (x-exception-id ≠ ‘-‘), and hence are not served to the user.

In Table 1, we report, for each dataset, the number of requests in it, corresponding dates, and number of proxies. Then, in Table 2, we list a few fields from the logs that constitute the main focus of our analysis. The s-ip field logs the IP address of the proxy that processed each request, which is in the range 82.137.200.42 – 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-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-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-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-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-exception-id = ‘-‘): a request that is allowed and served to the client (no exception raised).
- **Denied** (x-exception-id ≠ ‘-‘): a request that is not served to the client, either because of a network error or due to censorship. Specifically:
  - **Censored** (x-exception-id ∈ {policy_denied, policy_redirect})): a denied request that is censored based on censorship policy.
  - **Error** (x-exception-id ∉ {‘-‘, policy_denied, policy_redirect}): a denied request not served to the client due to a network error.
- **Proxied** (sc-filter-result = 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

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Requests</th>
<th>Period</th>
<th># Proxies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>751,295,830</td>
<td>July 22-23, 31, 2011, August 1-6, 2011</td>
<td>7</td>
</tr>
<tr>
<td>Sample (4%)</td>
<td>32,310,958</td>
<td>July 22-23, 2011, August 1-6, 2011</td>
<td>7</td>
</tr>
<tr>
<td>User</td>
<td>6,374,333</td>
<td>July 22-23, 2011</td>
<td>1</td>
</tr>
<tr>
<td>Denied</td>
<td>47,452,194</td>
<td>July 22-23, 31, 2011, August 1-6, 2011</td>
<td>7</td>
</tr>
</tbody>
</table>

Table 1: Datasets description.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cs-host</td>
<td>Hostname or IP address (e.g., facebook.com)</td>
</tr>
<tr>
<td>cs-uri-scheme</td>
<td>Scheme used by the requested URL (mostly HTTP)</td>
</tr>
<tr>
<td>cs-uri-port</td>
<td>Port of the requested URL</td>
</tr>
<tr>
<td>cs-uri-query</td>
<td>Query of the requested URL (e.g., ?refid=7&amp;ref=inf_trk&amp;trk)</td>
</tr>
<tr>
<td>cs-uri-est</td>
<td>Extension of the requested URL (e.g., php, txt, gif, ...)</td>
</tr>
<tr>
<td>cs-user-agent</td>
<td>User agent from request header</td>
</tr>
<tr>
<td>cs-categories</td>
<td>Categories to which the requested URL has been classified (see Section 4 for details)</td>
</tr>
<tr>
<td>c-ip</td>
<td>Client’s IP address (removed or anonymized)</td>
</tr>
<tr>
<td>s-ip</td>
<td>The IP address of the proxy that processed the client’s request</td>
</tr>
<tr>
<td>sc-status</td>
<td>Protocol status code from the proxy to the client (e.g., ‘200’ for OK)</td>
</tr>
<tr>
<td>sc-filter-result</td>
<td>Content filtering result: DENIED, PROXIED, or OBSERVED</td>
</tr>
<tr>
<td>x-exception-id</td>
<td>Exception raised by the request (e.g., policy_denied, disc_error). Set to ‘-‘ if no exception was raised.</td>
</tr>
</tbody>
</table>

Table 2: Description of a few relevant fields from the logs.
be either allowed or denied, even if \textit{x-exception-id} does not indicate an exception.

### 3.4 Ethical Considerations
Even though the dataset studied in this paper is publicly available, we are obviously aware of its sensitivity. Thus, we enforced a few mechanisms to safeguard privacy of Syrian Internet users. We encrypted all data (and backups) at rest and did not re-distribute the logs. We never obtained or extracted users’ personal information, and we only analyzed aggregated traffic statistics. While it is out of the scope of this paper to further discuss the ethics of using “leaked data” for research purposes (see [12] for a detailed discussion), we argue 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.1

We acknowledge that our work may be beneficial to entities on either side of censorship. However, our analysis helps understand the technical aspects of an actual censorship ecosystem. Our methodology exposes its underlying technologies, policies, as well as its strengths and weaknesses, and can thus 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 \( D_{\text{full}} \) (see Section 3.3).

### Traffic distribution

We start by observing the ratio of the different classes of traffic. For each of the datasets \( D_{\text{sample}}, D_{\text{ases}}, \) and \( D_{\text{denied}}, \) Table 3 reports how many requests are allowed, proxied, denied, or censored. In \( D_{\text{sample}}, \) 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 website 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 \( D_{\text{sample}} \)). The breakdown of \textit{x-exception-id} values within the proxied requests resembles that of the overall traffic.

### Denied traffic

Proxies also log requests that have been denied due to network errors: this happens for less than 6% of the requests in our sample. The inability of the proxy to handle the request (identified by the \textit{x-exception-id} field being set to \textit{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 \( D_{\text{full}} \)). 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 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.

---

1Note that we obtained the approval of INRIA’s and NICITA’s Institutional Review Board (IRB) to publish results of this work.
In Table 4, we report the top-10 allowed (resp., censored) domains in $D_{full}$. 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), google.com, gstatic.com, facebook.com, xvideos.com, google.com, gstatic.com, and skype.com 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 URL. As a result, we observe that a few well-targeted social media pages. Finally, networks 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 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 more details in Section 6.

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. However, according to Blue Coat’s representatives [25], the online services are not accessible to the Syrian proxy servers, and apparently the 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 more details in Section 6.

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

Table 4: Top-10 Domains (allowed and censored) in $D_{full}$.

Figure 3: Category distribution of censored traffic ($D_{sample}$). Categories obtained from McAfee’s TrustedSource, ‘Other’ is used for categories with less than 1K requests.

Due to the absence of URL categories, we use McAfee’s TrustedSource tool (www.trustedsources.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, 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 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.** 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 ($D_{sample}$). 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 method, and thus the proxy can censor requests based on the 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. While Blue Coat proxies indeed support interception of HTTPS traffic, 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-ext, 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, 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 client IP address and user agent designates a unique user. This means that a user connecting from several devices with different IP addresses (or a single device with different browsers) is not considered as a single user. Conversely, users behind NAT using browsers with identical user-agent are counted as one user. However, this combination of fields provides the best approximation of unique users within the limits of the available data [31].

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. Fig. 4(a) shows the distribution of the number of censored requests per user. We observe that 37.8% of these users have only a 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. We also look at 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. Censored users are more active than non-censored users: approximately 50% of the censored users generate more than 100 requests, while only 5% of non-censored users have the same level of activity. This might be explained due to the fact that, as discussed in Section 5.4, many requests are censored since they include a blacklisted keyword (e.g., proxy), and active users are more likely to encounter such keywords.

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 (<1%) of the traffic is actually censored. Most requests are 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

This section aims to understand the way the Internet is filtered in Syria. We analyze censorship’s temporal characteristics and compare the behavior of different proxies. Then, we study how the requests are filtered and infer 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 cen-
In conclusion, censorship peaks might be due to sudden higher volumes of traffic targeting Skype and MSN live messenger websites, which are being systematically censored by the proxies.

5.2 Comparing different proxies

We now compare the behavior of the 7 different proxies whose logs are included in our datasets. In Fig. 7(a), we plot the traffic distribution across proxies, restricted to two days (August 3 and 4) to ease presentation. Note that the load is fairly distributed among the proxies, however, if one only considers censored traffic (Fig. 7(b)), different behaviors become evident. 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.

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.

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

For more details on the comparison between proxies, and an evaluation of the similarity between censored requests handled by

\[ \text{Relative Censored traffic Volume (RCV)} \]

\[ \text{time of day} \]

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

We now compare the behavior of the 7 different proxies whose logs are included in our datasets. In Fig. 7(a), we plot the traffic distribution across proxies, restricted to two days (August 3 and 4) to ease presentation. Note that the load is fairly distributed among the proxies, however, if one only considers censored traffic (Fig. 7(b)), different behaviors become evident. 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.

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.

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

For more details on the comparison between proxies, and an evaluation of the similarity between censored requests handled by
As reported in Table 6, the most common URLs are policy_redirect to 11 in total. 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.

5.3 Denied vs. Redirected Traffic

Requests are censored in one of two ways: they are either denied or redirected. If a request triggers a policy_redirect 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 tcp_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 6, 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 Filtering

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 [1], 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 C be the set of censored URLs and A the set of allowed URLs; (2) Manually identify a string w appearing frequently in C; (3) Let C and A be the number of occurrences of w in C and A, respectively; (4) If Nc > 1 and Na = 0 then remove from C all requests containing w, add w to the list of censored strings, and go to step 2.

To mitigate selection of strings that are unrelated to the censorship decision during the manual string identification (step 2), we took a conservative approach by only 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 7 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 8 the top-10 categories according to the number of censored requests. Clearly, there is a heavy censorship 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 9. 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 /tpproxy/af/query, which can be found on the google.com domain, and is unrelated to anti-censorship software. Nevertheless, this element accounts for
Table 7: Top-10 domains suspected to be censored (number of requests and percentage for each class of traffic in $D_{\text{full}}$).

<table>
<thead>
<tr>
<th>Domain</th>
<th>Censored</th>
<th>Allowed</th>
<th>Proxied</th>
</tr>
</thead>
<tbody>
<tr>
<td>metacafe.com</td>
<td>1,278,583</td>
<td>0 (0.00%)</td>
<td>1,164 (0.03%)</td>
</tr>
<tr>
<td>skype.com</td>
<td>503,932 (6.83%)</td>
<td>0 (0.00%)</td>
<td>1,313 (0.04%)</td>
</tr>
<tr>
<td>wikipedia.org</td>
<td>306,994 (4.16%)</td>
<td>0 (0.00%)</td>
<td>3,379 (0.10%)</td>
</tr>
<tr>
<td>.il</td>
<td>112,369 (1.52%)</td>
<td>0 (0.00%)</td>
<td>8,785 (0.25%)</td>
</tr>
<tr>
<td>amazon.com</td>
<td>62,759 (0.85%)</td>
<td>0 (0.00%)</td>
<td>345 (0.01%)</td>
</tr>
<tr>
<td>aawsat.com</td>
<td>51,518 (0.70%)</td>
<td>0 (0.00%)</td>
<td>5,670 (0.16%)</td>
</tr>
<tr>
<td>jumbilo.com</td>
<td>23,214 (0.31%)</td>
<td>0 (0.00%)</td>
<td>0 (0.00%)</td>
</tr>
<tr>
<td>jeddabhiakers.com</td>
<td>21,274 (0.29%)</td>
<td>0 (0.00%)</td>
<td>130 (0.00%)</td>
</tr>
<tr>
<td>badoo.com</td>
<td>14,502 (0.20%)</td>
<td>0 (0.00%)</td>
<td>358 (0.01%)</td>
</tr>
<tr>
<td>islamway.com</td>
<td>14,408 (0.20%)</td>
<td>0 (0.00%)</td>
<td>329 (0.01%)</td>
</tr>
</tbody>
</table>

Table 8: Top-10 domain categories censored by URL (number of censored requests and percentage of censored traffic in $D_{\text{sample}}$).

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

Table 9: The list of 5 keywords identified as censored (fraction and number of requests for each class of traffic in $D_{\text{full}}$).

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Censored</th>
<th>Allowed</th>
<th>Proxied</th>
</tr>
</thead>
<tbody>
<tr>
<td>proxy</td>
<td>3,954,795 (33.61%)</td>
<td>0 (0.00%)</td>
<td>14,846 (0.42%)</td>
</tr>
<tr>
<td>hotpotshield</td>
<td>126,127 (1.71%)</td>
<td>0 (0.00%)</td>
<td>816 (0.02%)</td>
</tr>
<tr>
<td>ultrareach</td>
<td>50,769 (0.69%)</td>
<td>0 (0.00%)</td>
<td>68 (0.00%)</td>
</tr>
<tr>
<td>israel</td>
<td>48,119 (0.65%)</td>
<td>0 (0.00%)</td>
<td>477 (0.01%)</td>
</tr>
<tr>
<td>ultrasurf</td>
<td>31,483 (0.43%)</td>
<td>0 (0.00%)</td>
<td>541 (0.02%)</td>
</tr>
</tbody>
</table>

4.85% of the censored requests in the $D_{\text{sample}}$ 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 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 $D_{\text{full}}$) for which the $cs$-host field is an IPv4 address. We geolocalize each IP address in $D_{IPv4}$ using the Maxmind GeoIP database.\(^9\) 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 10 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.\(^{10}\) Table 11 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 targets a few geographical areas, i.e., Israeli subnets.

5.5 Summary

The analysis presented in this section has shown evidence 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. 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

\(^9\)http://www.maxmind.com/en/country

\(^{10}\)The list of IP4 subnets corresponding to Israel is available from http://www.ip2location.com/free/visitor-blocker.
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.

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 $D_{full}$, and report the top-10 censored social networks in Table 12.

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. 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.

**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 [21] stated that “the 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 12: Top-10 censored social networks in $D_{full}$ (number and percentage of requests for each class of traffic).

<table>
<thead>
<tr>
<th>OSN</th>
<th>Censored</th>
<th>Allowed</th>
<th>Proxied</th>
</tr>
</thead>
<tbody>
<tr>
<td>facebook.com</td>
<td>1,616,174 (21.91%)</td>
<td>17,703M (2.53%)</td>
<td>125,876 (3.60%)</td>
</tr>
<tr>
<td>badoo.com</td>
<td>14,502 (0.20%)</td>
<td>0 (0.00%)</td>
<td>358 (0.01%)</td>
</tr>
<tr>
<td>netlog.com</td>
<td>9,252 (0.13%)</td>
<td>2,011 (0.03%)</td>
<td>2,227 (0.06%)</td>
</tr>
<tr>
<td>linkedin.com</td>
<td>7,194 (0.10%)</td>
<td>186,047 (0.03%)</td>
<td>1,723 (0.05%)</td>
</tr>
<tr>
<td>skyrock.com</td>
<td>3,307 (0.04%)</td>
<td>7,395 (0.01%)</td>
<td>11 (0.00%)</td>
</tr>
<tr>
<td>hi5.com</td>
<td>2,995 (0.04%)</td>
<td>210,411 (0.03%)</td>
<td>463 (0.01%)</td>
</tr>
<tr>
<td>twitter.com</td>
<td>1,630 (0.03%)</td>
<td>2,384 (0.00%)</td>
<td>1,654 (0.42%)</td>
</tr>
<tr>
<td>ning.com</td>
<td>6 (0.00%)</td>
<td>41,993 (0.01%)</td>
<td>69 (0.00%)</td>
</tr>
<tr>
<td>meetup.com</td>
<td>3 (0.00%)</td>
<td>108 (0.00%)</td>
<td>0 (0.00%)</td>
</tr>
<tr>
<td>flickr.com</td>
<td>2 (0.00%)</td>
<td>383,321 (0.05%)</td>
<td>3179 (0.09%)</td>
</tr>
</tbody>
</table>

Table 13: Top blocked Facebook pages in $D_{full}$.

<table>
<thead>
<tr>
<th>Social plugin</th>
<th>Censored</th>
<th>Allowed</th>
<th>Proxied</th>
</tr>
</thead>
<tbody>
<tr>
<td>/plugins/like.php</td>
<td>694,788 (43.04%)</td>
<td>0 (0.00%)</td>
<td>8,919 (0.07%)</td>
</tr>
<tr>
<td>/extren/login_status.php</td>
<td>629,495 (38.99%)</td>
<td>0 (0.00%)</td>
<td>3,307 (0.03%)</td>
</tr>
<tr>
<td>/plugins/likebox.php</td>
<td>77,244 (4.78%)</td>
<td>0 (0.00%)</td>
<td>3555 (0.22%)</td>
</tr>
<tr>
<td>/plugins/send.php</td>
<td>70,146 (4.35%)</td>
<td>0 (0.00%)</td>
<td>272 (0.02%)</td>
</tr>
<tr>
<td>/plugins/comments.php</td>
<td>54,265 (3.36%)</td>
<td>0 (0.00%)</td>
<td>331 (0.02%)</td>
</tr>
<tr>
<td>/fbml/fbjs_proxy.php</td>
<td>42,649 (3.64%)</td>
<td>0 (0.00%)</td>
<td>43 (0.03%)</td>
</tr>
<tr>
<td>/connect/canvas_proxy.php</td>
<td>40,516 (2.61%)</td>
<td>0 (0.00%)</td>
<td>37 (0.03%)</td>
</tr>
<tr>
<td>/ajax/proxy.php</td>
<td>1,544 (0.10%)</td>
<td>0 (0.00%)</td>
<td>6 (0.00%)</td>
</tr>
<tr>
<td>/platform/page_proxy.php</td>
<td>1,519 (0.09%)</td>
<td>0 (0.00%)</td>
<td>4 (0.00%)</td>
</tr>
<tr>
<td>/plugins/facpile.php</td>
<td>669 (0.04%)</td>
<td>0 (0.00%)</td>
<td>4 (0.00%)</td>
</tr>
</tbody>
</table>

Table 14: Top-10 Facebook social plugin elements in $D_{full}$ (fraction of Facebook traffic and number of requests).

As shown in Table 13, many requests to targeted Facebook pages are allowed and no allowed request is categorized as “Blocked Site.” We also identify successful requests sent to Facebook pages such as Syrian.Revolution.Amy, Syrian.Revolution.Assad, Syrian.Revolution.Caricature and ShamaNewsNetwork, which are not categorized as “Blocked Site” and are allowed. Finally, proxied requests are sometimes, but not always, categorized as “Blocked Site” (e.g., all the requests for the Syrian.revolution page).

**Social plugins.** Facebook provides so-called social plugins (one common example is the Like button), which is loaded by web pages to enable interaction with the social platform. Some of the URLs used by these social plugins 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 14 reports, for each of the top-10 social plugin elements, the fraction of the Facebook traffic and the number 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 10 social plugin elements we consider account for 99.9% (1,612,835) 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.

**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 specific content addressing users who access targeted Facebook pages, thus making censorship and surveillance harder to detect.
We further isolate HTTP signaling messages by identifying all HTTP requests to the Tor project website and the Tor project directory, e.g., /tor/server/authority.z or /tor/keys.11 This does not take into account the connections via Tor bridges: there is no public list of them (bridges are used to overcome filtering of connections to known Tor relays), however, Tor relays were not filtered in Syria as of 2011, thus users did not actually need to use bridges.

We identify 95K requests to 1,111 different Tor relays, 73% of which belong to Torhttp. Only a small fraction (1.38%) of the requests are censored and 16.2% of them generate TCP errors. Figure 8(a) shows the number of requests for August 1-6. The traffic has several peaks, in particular on August 3, when several protests were taking place. We observe that 99.9% of censored Tor traffic is blocked by a single proxy (SG-44), even though the overall traffic is uniformly distributed across all the 7 proxies. The other 0.01% of the traffic is censored by SG-48. This finding echoes 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 8(b), Tor censoring exhibits a higher variance. While censoring Torhttp is technically simple, as it only requires matching of regular expressions against the HTTP requests, identifying and censoring Toronion is more challenging, as it involves encrypted traffic. However, only Toronion traffic is censored according to the logs, while Torhttp is always allowed.

Next, we verify whether the Tor censorship is consistent (i.e., whether the blocked Tor relay is re-allowed later). To do so, we proceed as follows: First, we create a set of all censored Tor relay addresses denoted Censored-IPs. Then, for each one hour time window considered as a time bin k, we create the set of all Tor relays addresses that were allowed, we denote Allowed-IPs (k). For each time bin k, we calculate the relative overlap between the censored Tor relay nodes and the Tor relays allowed at bin k as:

\[ R_{\text{filter}}(k) = 1 - \frac{|\text{Censored-IPs} \cap \text{Allowed-IPs}(k)|}{|\text{Censored-IPs}|} \]

where |·| denotes the set size. Figure 9 depicts the variation of \( R_{\text{filter}} \) as a function of time. Note that \( R_{\text{filter}}(k) \) equals 0 if, at a specific time bin k, all connections to Censored-IPs are allowed (line curve) or none of the IPs in Allowed are in Censored (circle curve). \( R_{\text{filter}}(k) \) equals 1 if none of the connections to Censored-IPs have been allowed at a specific time bin k.

The high variance of \( R_{\text{filter}} \) shows that the censorship is inconsistent. In fact, the beginning of this plot is either dominated by red circles (all traffic is allowed) or few small peaks showing a very mild censorship. Then few successive peaks with a high variance of blockage appear for several hours, followed by a lull at the night of 03 of August. The same scenario is repeated several times, alternating between aggressive censorship and more mild period. This behavior is hard to explain as some IPs are alternating between blocked and allowed status. One explanation might be a testing phase that is carried in a single router and for a short period of time to test a new censorship approach (in this scenario the newly deployed Tor censorship). Or, this censorship is based on fields that are not logged by the appliance and hence inaccessible to us.

7.2 Web Proxies and VPNs

As already mentioned, 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 hotspotsield are heavily monitored and censored. Nonetheless, some web proxies and VPN software (such as Freegate, GTunnel and GPass) do not include the keyword proxy in request URLs and are therefore not censored. Similarly, we do not observe any censorship- and surveillance-evading tools (e.g., in web searches) for the dataset.

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 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 corresponding 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. We find a total of 338,168 BitTorrent announce requests from 38,575 users (BitTorrent uses a 20-byte peer_id field to identify peers, which we use to count unique users) for 35,331 unique contents in the $D_{full}$ dataset. Most of these requests (99.97%) are allowed. Censored requests can be explained 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 9, are actually present in the titles of some of the BitTorrent files, yet the associated announce requests are allowed. 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), however, we identify content relating 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 key-

---

12 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.
word 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 is obviously limited in scope, the logs actually suggest that it is very effective. Thus, when properly secured with HTTPS, Google cache could actually 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, e.g., by Danézis and Anderson [11]. 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. Censors might aim 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. We also highlighted that users attempt 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 known censorship-evading technologies, such as web/socks proxies and Tor.

9. CONCLUSION
This paper presented a measurement analysis of Internet filtering in Syria. We analyzed 600GB worth of logs produced by 7 Blue Coat SG-9000 proxies in Summer 2011 and, by extracting information about processed requests for both censored and allowed traffic, we provided a detailed, first-of-a-kind snapshot of a real-world censorship ecosystem. We uncovered the presence of a relatively stealthy yet quite targeted filtering, which relies on IP addresses to block access to entire subnets, on domains to block specific websites, and on keywords to target specific content. Keyword-based censorship (e.g., denying all requests containing the word ‘proxy’) also produces collateral damage, as many requests are blocked even if they do not relate to sensitive content. Finally, we showed that Instant Messaging software is heavily censored, that filtering of social media appears to be limited to specific pages, and that Syrian users try to circumvent censorship using web/socks proxies, Tor, VPNs, and BitTorrent.

10. REFERENCES


