Packets Lost in the Wild: An Analysis of Empirical Approaches to Measure Internet Censorship

Mohammad Taha Khan
taha@cs.uic.edu

Department of Computer Science
University of Illinois at Chicago

ABSTRACT

Over the past 20 years, the Internet has transformed into a global medium of communication via the publishing and access of content. Even though it is built on the very axioms of global connectivity, free speech, and net neutrality, there have been multiple instances of blocking by censoring regimes to restrict Internet traffic across their borders. On average, 25.3% of Internet users experience content blocking in one form or another. This form of censorship is implemented by exploiting Internet protocols, ranging from DNS injection and TCP RSTs to large scale AS level BGP hijacks. This phenomenon has been studied by many researchers in aims to understand the nature of blocking as well as to quantify such policing measures. This paper provides a background of the primary motivations behind Internet censorship and the various technical mechanisms used by censors to implement it. It then presents a detailed survey and evaluation of the systems developed by researchers to detect and quantify censorship. Finally, we provide summary insights as well as suggestive directions of focus to aid researchers in developing more accurate and robust measurement systems.

1. INTRODUCTION

The Internet today, is accessible to over 3 billion individuals around the world. It has become an essential commodity for a verity of reasons which include online communication, recreation, e-commerce, and the promotion and expression of opinions and ideas. Despite its pervasive use, access to the Internet is irregular due to certain aspects. One of the primary reasons behind this is Internet censorship. According to the reports collected by the Open Network Initiative [7], more than 60 countries experience some sort of content regulation and blocking due to political and religious reasons. This form of censorship is usually an attempt to block certain concepts and channels that promote freedom of speech or the expression of ideas that go against the ideal beliefs of the censoring regimes.

Over time, censors have continually devised sophisticated techniques to manipulate Internet traffic. On the other hand, activists and the proponents of free speech have come up with various mechanisms to evade these blocks. This, in turn, has led to an increased interest in the academic community to understand the concept of censorship. Although it is a very broad topic with several avenues of exploration, most of the technical research focuses on aspects of detecting and quantifying Internet censorship from empirical evidence [20, 39, 22, 36, 7, 21, 17, 3]. Other researchers have also aimed at developing circumvention tools [13, 33, 37]. In this paper, we focus on the developed censorship measurement systems and methodologies in the past 10 years. The following discussed approaches serve as a focus of this paper.

Concept Doppler [20]: Concept Doppler is an architecture developed specifically to measure censorship via keyword filtering in HTTP GET requests. The system also incorporates the basic language semantic techniques to create and maintain a list of potential list of blocked keywords.

URL Filtering Detection[22]: This discussed methodology focuses on detecting the URL filters developed by third party enterprises. The approach focuses on externally visible installations of these devices and uses complementary URL blacklist submissions to confirm their use for Internet censorship.

Censmon [36]: Censmon is a server based distributed system which leverages an overlay client network to collect censorship measurements. Censmon is a more generic development and can detect and differentiate specific instances of DNS, IP and HTTP level blocking.

Encore [18]: Encore is a measurement system that harnesses the primary capability of cross-origin requests within browsers to collect censorship measurements. The use of browsers as vantage points allows the system to capture measurements from globally diverse regions. The system also relies on a central server that distributes and collects measurements.

Apart from an in-depth description of these systems, we also evaluate their performance and state the limitations to their design. Based on our analysis, we then provide summary insights to help the community better understand the current state of Internet censorship. Furthermore, we suggest some future research directions in hopes that they will eventually aid researchers in developing more accurate and robust measurement systems.

2. BACKGROUND

The concept of censorship predates the creation of the Internet. Individuals in various circumstances have been regulated on the basis of their moral and political beliefs. Similarly, the struggle for freedom of expression has also existed equally in time throughout history [6]. While the inherent goal of these censoring regimes has always been to contain the ideas of religious and political freedom, with the Internet now being the primary medium for the propagation of thoughts and ideas, the methods in which the regimes decide to implement censorship have shifted from social to technical ones.

In this section, we provide a comprehensive overview of how the motivations of the censors are brought to technical realization. Taking a top-bottom approach, we primarily reorganize the censorship mechanisms stated in previous works [16, 31] and explain how each of them is implemented, either on a single, or a combination of layers of the network stack. The goal of this section is to provide the readers with the essential technical details of the blocking mechanisms before moving onto the in-depth analysis of the measurement methodologies in section 3.

DNS Tampering.

DNS is an application layer protocol that provides the primary service of hostname to IP mapping. A DNS resolution is usually the very first step in establishing a connection with a web server. As most censors have control over the recursive resolvers within the network, to implement censorship, they maintain a blacklist of domains and redirect an individual to a different block-page, or respond with an NXDomain. As this technique can sometimes be circumvented by contacting an open recursive resolver, some censors implement a more sophisticated version of DNS-based blocking which involves the injection of packets to a stub resolver. In the latter approach, ISP routers observe DNS packets and inject responses that imitate actual ones but with fake data. While this technique is through, it certainly suffers from collateral damage [30]. Recently, more advanced implementations [24] of stub resolvers have been suggested to circumvent censorship by DNS injection.

TLS Blocking.

This censorship method enables the blocking of domains that establish connections over HTTPS. Even though data communication is encrypted, in the initial key exchange, the client provides the hostname string referred as the server name indication (SNI). This capability was introduced as an extension to TLS 1.2 to allow secure connections with virtual servers that host multiple domains under the same IP address [11]. A censor can monitor the initial key exchange process and drop packets that contain a filtered domain. Nisar et al. show this method actively being used in Pakistan (in 2015) to block the access of YouTube over HTTPS [33].

HTTP Filters.

Censors implement HTTP filtering by installing proxy devices or URL filters (section 3.2). These devices are middleboxes which intercept all HTTP traffic. This technique allows more flexible and precise censorship than DNS or TLS blocking, as the filtering device can see the domains as well as the specific pages which are requested. In case the header contains any content that is supposed to be censored, the packets are dropped and the client is returned 404 or a block-page. Such devices, when implemented on a national level require an immense amount of processing power. This can sometimes lead to impaired blocking as a result of the proxy filters not being able to process complete realtime traffic. To deal with such scenarios, censors implement multi-stage blocking within the ASes as well as the country's border, which helps to distribute the traffic load.

Keyword Filters and TCP Disruption.

This blocking mechanism is implemented by incorporating elements from both the application, as well as the transport layer. Similar to HTTP filters, network based middleboxes are installed which look for specific keywords in request headers and the HTML responses. If blacklist keywords are detected, the ongoing connection is disrupted by sending out TCP RST packets to each connection endpoint. While the actual inspection of the traffic is performed over HTTP data, the devices leverage aspects of the TCP protocol to disrupt the ongoing communication process. Using the Great Firewall of China as a case study, previous works [39, 19] elaborate on the functioning of this blocking technique in the wild. This paper provides a more detailed evaluation of this mechanism in section 3.1.

IP Based Blocking.

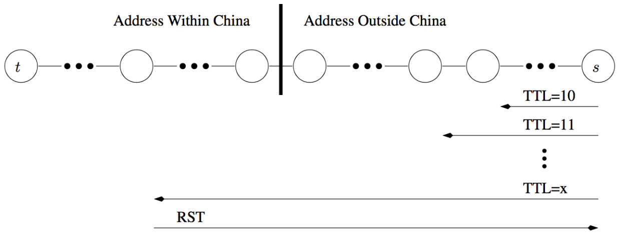
IP based blocking in implemented within routers which inspect the IP headers and block packets that match against a blacklisted destination address. This technique can be fast and efficient as simple IP hashing can be processed in constant time. However, it suffers from some limitations, the first one is keeping a track of an updated IP blacklist and its distribution across multiple routers. Secondly, most web services use name based virtual hosting which has multiple domains associated with a single IP address. Blocking based on IP can lead to collateral damage by hindering access to the other legitimate services being served under the same IP.

BGP Highjacking.

The Border Gateway Protocol (BGP) is responsible for propagating correct route information on the AS level. To enforce censorship with BGP path highjacking, a censor broadcasts a false shortest path prefix internally within an AS to route the traffic into a black hole. Censorship of this scale can only be performed by ISPs that have control over BGP traffic. Unlike other censorship methods, which require additional infrastructure and complex reconfiguration of devices, prefix highjacking is easy to implement and provides fast results. However, as BGP relies on a tentative trust model, if this technique is not implemented in a proper manner, it can have serious adverse effects. An instance of which occurred when in 2009, a Pakistani ISP accidentally announced shortest prefixes for YouTube to external, instead of internal ASes, causing a global downtime of the website that lasted several hours [8].

3. MEASUREMENT METHODOLOGIES

In this section, we cover the prominent detection and measurement methodologies used by researchers to evaluate the various types of censorship strategies summarized in section 2. The discussed approaches use active measurements and collect empirical evidence for their analysis. While the main focus is on the system description, we provide a summary of the actual results obtained by each of the following approaches. We also give an evaluation of these methodologies and outline some of the limitations of each approach.



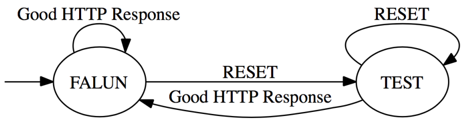


Figure 1: Keyword probing state machine to test for keywords [20]

3.1 Measuring Keyword Filtering

Keyword filtering is an application layer blocking mechanism which is performed by the inspection of HTTP GET requests and responses for specific keywords. In this section, we refer to the devices performing keyword filtering as firewall routers. On detection, these devices inject RST packets to disrupt ongoing TCP and sometimes also temporarily block the IP address from connecting. As censorship is an economic activity [23], blocking certain content can cause collateral damage. One of the major advantages of using keyword filtering as a blocking technique is that it allows more fine-grained filtering of content. The ability to partially block domains minimizes the collateral harm. An anecdotal example of this kind of blocking is as follows, keyword filtering at the Great Firewall of China (GFC) would allow a GET request to the URL `www.cnn.com/search/?text=olympics` while blocking a request to `www.cnn.com/search/?text=falun`[1].

System Description.

Concept Doppler [20], evaluates the amount of keyword filtering occurring in China. While the system methodology is tested on the GFC, the approach is generic enough to evaluate any keyword based filtering mechanism implemented for censorship purposes. Blocking based on individual keywords can be a computationally intensive task as it involves the inspection of every packet in real-time. To deploy this in an efficient manner China has its filtering devices implemented in distributed fashion on the AS level [39]. These on-path devices monitor connections and perform blocking by injecting RST packets to end connections.

Measuring keyword filtering involves two distinct dimensions. The first is the localization of the specific devices that perform blocking and understand where filtering occurs. The second is the development of an efficient probing technique and maintain a develop a comprehensive list of the filtered keywords.

To probe and discover the actual placement of the firewall routers, the developed an algorithm leverages the decrementing TTL field in a packet. Initially, a representative sample of target servers is curated using the top 100 URLs ending in `.cn` as a result of a Google search. The URLs are then mapped to the IP addresses, which correspond to the servers present within China. A TCP connection is first initiated with the destination. Next, specific GET request packets containing a filtered keyword are crafted and sent with increasing TTL values. Once the packet reaches a firewall router, a RST packet to the source is returned. This is also accompanied by a `time exceeded` ICMP message from the router as a result of the TTL value reaching zero. The information about the source IP of the RST packet along with the ICMP response can be used to identify the location of the firewall router within China. The approach has been visualized in Figure 3.1. The technique finds that approximately 28% of the probes sent are not blocked, and 29% of the filtering is done past the first hop past the border of China and is distributed different ISPs. A similar technique has also been used by researchers in [39] to understand how the GFC blocking mechanism has been deployed.

Enforcing censorship with keyword filtering also requires maintaining a blacklist of words at the firewall routers. Measurement via probing all possible keywords can be arduous and invasive. This essentially necessitates the need for an efficient method to replicate a comprehensive blacklist so that it may be monitored over the varying temporal and geographic scales. Latent Semantic Analysis (LSA) [29] is a technique that allows the extraction of a set of common concepts from a set of documents. While the technical details of LSA are beyond the scope of this paper, we cover them in the Appendix for interested readers. To first accumulate a linguistic corpus of documents, the system collects all the articles from the Chinese-Language Wikipedia. It then associates terms within the wiki link with the document. To extract further keywords, they perform LSA correlation analysis with 12 well-known concepts filtered in China. Generating blocked keywords by seeding from concepts is an organic approach and is likely to produce more precise results as it replicates the initial intention of the censors, which aim to block certain concepts, however they have to revert back to keyword level granularity due to technical limitations.

The LSA analysis provides a list of terms from the documents that are highly correlated with the seed concepts. Using that list, a total of 30,000 terms and 122 previously unknown blocked keywords are discovered. Figure 1 shows the approach of how each term in the list is tested. An HTTP GET request containing a keyword from the list is launched to `ww.yahoo.cn`. If it generates a RST packet from the GFC, it is tagged as blocked, the system then waits for 30 seconds and switches to a non-filtered keyword eg. `TEST`, until a valid HTTP response is received. The valid response serves as an indicator of the system being in a stable state, ready to be used to test the next keyword. The wait between making requests ensures that RST packets are not confused among keywords. It also allows the temporary IP block timeout at the GFC router to expire.

Evaluation.

Next, we discuss some of the assumptions and limitations of this measurement approach. The firewall router discovery

[1] *Falun* is a commonly censored keyword in China

method is based on the assumption that every incremental TTL packet is sent via the same route. Though the experiments are through, there is no validation of path consistency when discovering the routers. This described approach is able to detect the scope of censorship specifically within the current implementation of the GFC. If in case the GFC implementation changed, the methodology would require revisions. For instance, the current state of the firewall routers sends RST packets to both ends to terminate the connection. However, if the GFC policy just decided to drop the packets the measuring technique would yield inaccurate results and would need to be extended to evaluate dropped packets.

The keyword detection mechanism only represents the state of the blocking of external GET requests. The developed system does not test for how the firewall routers would treat HTML response packets which are also inspected [19]. Additionally, the system does not test for the symmetry of the state of blocking of requests originated internally vs the ones that are from outside of China. As the firewall routers are able to inspect the source IP address, it is possible that firewalls have different rules based on the origin IP address in means to implement more fine grained blocking.

For keyword extraction, the authors make use of a basic version LSA. This technique, when performing rank reduction, assumes the word distribution in a text corpus being of Gaussian nature. However, in reality, terms in languages tend to follow a Poisson distribution [28]. A technique that would provide more accurate correlated keywords for probing would be to use probabilistic LSA [27].

3.2 Detecting URL Filtering Products

Third party URL filters are products that allow traffic management in certain controlled environments. These devices usually come with a preloaded database of common blacklist URLs pertinent to specific categories. Operational administrators also have the capability of creating custom block categories. In some cases, URL filters regularly fetch blacklist updates from servers. They are commonly produced in Europe and North America by organizations like McAfee, Blue Coat, and Netsweeper [1, 5, 14]. While primarily intended for very specific corporate and educational environments, the methodology discussed is able to identify substantial evidence of their use for censoring traffic in several censoring regimes of the world.

Measurement Methodology.

Dalek et.al [22] develop a systematic approach to detect the presence of URL filtering products in specific networks and confirm their use for Internet censorship. As censorship is a regional phenomenon, access to appropriate vantage points is one of the key challenges of detecting URL filters, this in some cases, can be too risky under oppressive governments. Furthermore, the identification also requires an understanding of the specifics of the filtering products in order to yield accurate results. The suggested methodology is scalable and does not require the involvement of user reports. The technique leverages on certain Internet measurement and scanning resources. It initially makes use of the Internet indexing tool, Shodan [12], to collect all globally visible IP addresses and their corresponding metadata (e.g. HTTP headers). To shortlist the IP addresses associated as potential installations of URL filters, specific identifiers associated with URL proxy filters are searched within the metadata fields of the Shanon dataset. These identifiers are keywords extracted from the datasets from previous manual analysis [7]. As a further validation step, the methodology incorporates the use of WhatWeb [15] (a signature profiling tool) to confirm the product installed at a give IP host. IP to ASN mappings are also evaluated to ensure that the location of the URL filters resides within any one of countries under observation.

Next, to actually confirm that these products are being used for censorship, and not for legitimate use, two sets of experiments are devised. First, a measurement client is setup inside the ISP where censorship is suspected. The client generates requests to certain URLs which are censored . As a control set, a control client outside the ISP also generates requests to the identical URL list. The difference in response provides evidence if the URL filters in the specific ISP is being used for censorship. The second set of experiments leverages submitting potential URLs for blocking to the vendors that such as McAfee, Netsweeper and Blue Coat, which provide this service. A set of websites containing potentially objectionable content e.g. adult images, or proxy anonymizers, is created. To ensure that they are accessible, the created websites are then reached from within the suspect ISPs using these products. Next, the website URLs are submitted to the blocking service, and after a certain amount of time (3-5 days) it is observed that the URLs are no longer accessible.

This methodology is tested for ISPs in Saudi Arabia, Qatar, and the UAE. A list of URLs associated with commonly censored categories such as human rights, freedom of speech, political/religious freedom and LGBT rights is curated by regional experts from the countries. These lists are then tested by the measurement client present within the country. The results indicate that URL filtering products are pervasively being used for the blocking of such content which is against the fundamentals of human rights and free speech.

Evaluation.

Although the technique presented provides insight on how certain URL filtering products are used for censorship, it undergoes certain limitations. The first one being that the discovery method is not robust enough to identify devices that are not visible on the global Internet. While the approach can detect some instances of URL filters for censorship, it will most likely miss out more sophisticated installations which are not publicly visible. The approach also heavily depends on certain external web services and previous datasets to generate their URL blacklists. There is no obvious validation of the accuracy of the complementary datasets incorporated in the current methodology.

Scalability is also a considerable limitation of the current methodology. The current set of results are generated from specific ASes within the ISPs and there is no way to encapsulate how the methodology would differ across different regions within the country. This would necessarily require more measurement clients within the regions to be tested.

The methodology also depends on very specific implementations of the URL filters. As the use of these devices is a source of economic growth for the vendors, they can possibly collude with the censors and use trivial mechanisms to evade the current detection approach. The vendors can remove any identifiable metadata in the application layer data to camouflage their device installations. Furthermore, they can also preclude the option of external URL submission, or adopt a more scrutinized submission policy to remove dummy submissions used for the confirmation of censorship in the methodology.

Overall, the technique provides a way to detect and confirm the use of URL proxy filters for censorship in specific countries. While there is evidence of the misuse of such products, due to scalability reasons, it can only be used to test a subset of regions with countries. Also, as it depends on external web scanners and previous data-dumps, it is not adaptable to the changing dynamics of censor and vendor implementations.

3.3 Distributed Censorship Measurement

While keyword and URL filtering mechanisms are very specific and targeted in nature, researchers have also developed more generic and distributed censorship monitoring systems that perform measurement from within specific vantage points, and report to a central server for analysis. Censmon [36] is one such development. The system provides real-time monitoring of censorship data. It takes in a feed of target URLs as inputs at a central server. The server then distributes these target URL to PlanetLab [9] nodes in certain global locations which perform thorough testing of the URLs and report the results back to the server. The system design is robust and efficient in nature; it has the capability to differentiate network failures from censorship events as well as to identify different blocking types. The system is automated and does not require any user interaction for censorship evaluation. Figure 2 shows an overview of the Censmon design architecture.

System Description.

The design of Censmon can be divided into three distinct phases. The first is to be able to collect a representative sample set of URLs for testing. To enable this, the system provides a front-end submission form to crowdsource URLs from individuals. It also uses the Google Alerts and Trends services as well as Twitter Trends to get updates about real-time content generated on the Internet pertinent to the censorship topics. As the trend services of Google and Twitter do not provide direct URLs, the system collects top 10 URLs returned by Google search for each relevant trend. Furthermore, it also incorporates previous web URLs collected by ONI's monitoring system [7] along with URLs associated with their global list of censored categories.

Having a comprehensive list of URLs to test, the next phase is the distribution and collection of URL test results. The distributed network comprises of 174 PlanetLab nodes in 33 countries. To evaluate the censorship status of a URL, each node takes the following systemic approach:

1. A DNS resolution request is made to the URL under consideration. After attempting resolution, the IP results of the resolution along with any errors such as NXDOMAIN are recorded. This initial step allows the detection any DNS level censorship.

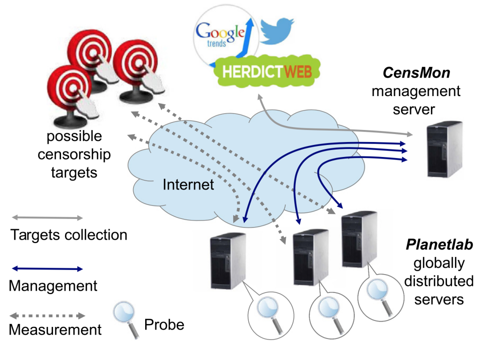

Google trends
HERDICTWEB
CensMon management server
possible censorship targets
Internet
Targets collection
Management
Measurement
Probe
Planetlab globally distributed servers

Figure 2: The Censmon architecture model diagram explained by Aceto et al. [16]

2. On successful resolution, the system attempts to establish a TCP connection on port 80 of the IP address. This specific step establishes a stateful connection with the server and also identifies if there is any IP-level blocking in place. During this process, each node also makes note of any network reachability issues.

3. The system then generates 2 specific HTTP requests to evaluate URL or keyword filtering. This step also differentiates between filtering at the URL or HTML response level. To identify URL request filtering, an initial request is made to a dummy server by appending the actual test URL as a query parameter. If the response is different from the intended dummy response, the system becomes aware of a URL filtering event. The second request is made to the actual test URL, the HTML response is stored and reported back to the central server along with other previously collected meta information for further analysis.

Once the central server receives information from the nodes regarding a specific test URL, it performs the following analysis steps to confirm and categorize the censorship event.

1. In the case of any reported reachability issues, the system issues a repeated task on the node to differentiate censorship from network issues. While the technique is simple is serves an efficient purpose for removing any false positives.

2. The central server next looks at the Whios records for the actual test URL domain and compares them with the information provided by the nodes to characterize censorship instances of DNS blocking.

3. For URLs with successful responses, the central server pareses the HTML DOM of the URL response to extract content that is likely to be textual in nature. This is essentially performed by applying a readability function [10]. It then uses fuzzy hashing to compare the readable HTML text to specifically identify instances of partial content filtering.

The developed System is tested on approximately 4950 unique URLs from 2500 domains and is able to detect 95

URLs from 193 domains. More than 90% of the filtered domains are found in China. The system also categorizes the types of detection, with HTTP filtering accounting for 48.5% of the blocking, and DNS and IP-based blocking, which are 18.2% and 33.3% respectively.

Evaluation.

In comparison to the previously discussed systems, Censmon is generic in design and can perform diverse evaluations of censorship instances. However, it suffers a similar limitation to the approach discussed in section 3.2 in regards to scalability. The current set of the experimental network only consists of PlanetLab nodes. Using these nodes, the system is able to detect censorship in 7 of the 33 evaluated countries, while most of the blocking instances being from within China. Censorship is a time variant phenomenon, currently taking place in over 60 countries on different severity levels [38, 7]. This requires more spread out vantage points to get a more clear insight into the state of blocking. Also, the evaluations derived from the PlanetLab nodes, which belong to academic networks are not clearly representative of the residential and broadband and residential networks [32].

The current detection mechanisms used by Censmon for DNS blocking is based on the assumption that each domain resolves to a unique set of IP addresses. In case a web service is distributed servers to serve different regions, DNS records will be localized and hence the current system will not be able to differentiate between the false positives as a result of distributed global servers. Furthermore, To detect HTML response and partial filtering, the system relies on a set readability heuristics. While most of the censorship results are from China, research indicates that HTML response filtering was previously discontinued [34]. Hence the current evaluation is not a valid measure of the response filtering method, and there is still need for validation or control testing to verify that the heuristics yield accurate results.

3.4 Measuring via Cross Origin Requests

As seen in previous mechanisms, one of the primary issued faced while performing continuous censorship measurement is the access to globally distributed vantage points. This is usually done by setting up setting up measurement clients and running install scripts. In some scenarios, researchers have to interact with locals, which entails language and cultural barriers. Such deployments can be resource consuming, have a high maintenance cost and are short-lived. As an alternative, cross origin requests within browsers can be harnessed to measure censorship. The inherent structure of the Internet allows cross origins requests across web domains [2]. This allows resources on one web page to be requested from a different domain, outside the scope of that web page. While web browsers enforce limitations on cross-origin embedded resources such as images, videos, iframes, and scripts are permitted. Leveraging this technique, the existing, unmodified web browser instances can become vantage points for collecting censorship measurements.

System Description.

Enocre[18] is a system developed by Brunett et al. that leverages side channels that exist due to cross-origin requests to infer the state of URL filtering in a specific region. The developed system requires webmasters to install a measurement script on their website homepage. The installed script is a self-contained and autonomous, this ensures that modern browsers will not block it due to security reasons. As an efficient design choice, instead of using simple AJAX requests, measurement tasks are performed by the script by embedding image, scripts, stylesheets and iframe tags in the page source.

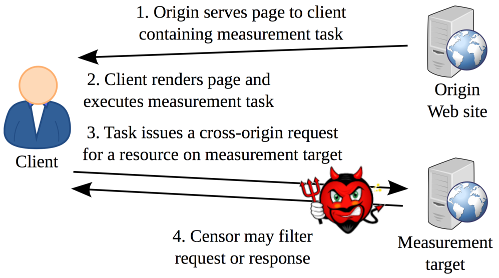

1. Origin serves page to client
containing measurement task
2. Client renders page and
executes measurement task
3. Task issues a cross-origin request
for a resource on measurement target
4. Censor may filter
request or response
Client
Origin
Web site
Measurement
target

Figure 3: The Encore architecture. [18]

Even before initiating the measurement process, Encore requires a list of URLs that need to be tested on the measurement clients. The system acquires a list of 700 potentially high-value URLs (and their corresponding patterns) which are likely to be censored. The next step is the conversion of the URL patterns into a specific measurement task by selecting the resources on the censored page that can be embedded b the measurement script. This is done as a multistep process; first, all the URL patterns are transformed into URLs by searching them on the web and collecting the absolute URLs. Next, a sub-system placed in a non-censored network makes requests to all the URLs and stores the complete response in an HTTP archive (HAR). Finally, a selector module analyzes the timing and type information of the specific elements in the HAR. The ones that have a minimal overhead and match the requirements are converted into embedded measurement tasks. These tasks are then distributed to each client from a coordination server.

Figure 3 shows how the measurement process works from a client's perspective. The web browser serving as a vantage point makes a request to one of the websites that have the Encore measurement system. On receiving the embedded Encore script, the client browser establishes a parallel connection with a coordination server, which intelligently assigns the client a measurement task from the pre-created pool. These tasks are the embedded cross-origin requests to image, stylesheets, scripts and iframe elements associated with the censored URLs. The client then performs a request to that embedded resource and informs the collection server about the result of the assigned task.

To test the blocking of complete domains, Encore embeds image and stylesheet elements from the censored website. To detect if an image was loaded, the `onload` or the `onerror` JS functions are invoked, while stylesheets loads are verified by ensuring the browser applied the style to the page. If multiple resources on a domain are blocked, Encore infers that the complete domain being blocked.

As censors sometimes block specific web pages as opposed to complete domains, an associated challenge with cross-origin based measurement is the identification of the kind of blocking taking place. To detect the granular blocking of specific web pages, encore makes use of the iframe as well as script elements. Specific web page URLs are assigned by the coordination server to the client to be loaded within an iframe. As there is no direct way for the client to inform if the iframe load was successful or not, this inference is made by the loading timing information. It assumes, if a load was quicker that normal, the browser already had cached content and it is likely that the page is not blocked by the censor. Similarly, script based measurements are performed on Chrome as it allows the request of non-script elements from within a script tag. The non-script elements are essentially web page URLs which, on successfully loading, lead to the invocation of a Javascript `onload` event. This then triggers a report to the collection server about that specific web page. Script based measurements are only assigned to Chrome user agents by the coordination server due to security reasons.

For gathering measurements, Encore was voluntarily installed by 17 webmasters and resulted in a collection of 141,626 unique measurements from approximately 88K distinct IP addresses. The top countries that contributed to the measurements were China, India, UK, Brazil, Egypt, South Korea, Iran, Pakistan, Turkey and Saudi Arabia. All of these countries perform some sort of censorship and filtering. Furthermore, to validate the accuracy of the measurements, and gauge system feasibility, a controlled testbed is also setup, which performs various kinds of DNS, IP, and HTTP filtering. 30% of the client measurement tasks correspond to accessing various types of filtered and non-filtered content on from the testbed server.

Evaluation.

The deployment and integration of Encore are simple and efficient. However, the system raises considerable ethical concerns as it triggers unwanted requests to potentially sensitive websites from an individual's browser. Some of these requests are from geographical regions where accessing such content can have implications which exist beyond the cyber world. The authors further argue that informed consent can be impractical for continuous and long-term measurements. Considering the potential risk associated with this technique, individuals should submit consent before any measurements are performed on their behalf. Technically, this can be done by the coordination server, serving a popup which allows users to submit consent before starting a measurement. While the authors claim that Encore can be difficult to due describe due to language barriers, research has shown that more than 50% [26] Internet users can interpret English. For non-English countries, language specific consent popups can be presented based on IP-geolocation.

The proper functioning of Encore is highly dependent on webmasters volunteering to add it to their websites. While techniques to evade censors blocking Encore measurement traffic have been suggested, the websites hosting Encore can also be blocked by the censors. Apart from the associated collateral damage, this is also an obvious economic downside for webmasters. The current system does not provide enough incentive for webmasters to install it in the first place.

As all the measurements are executed within the browser framework, it is unable to distinguish between the types of filtering that is taking place. This, however, can be solved in future implementations if the Web API provides more versatile requests that allow web-pages to execute more specific network communication such as DNS requests etc. What makes Encore unique in its design is the ability to perform continuous measurements. However, the current version is limited in independently detecting new censored topics and URLs in different regions of the world. This is because the set of measurement URLs generated by Encore are based on a seed list which has already been aggregated by previous measurement systems [7].

Furthermore, as browsers become more secure, the nature of cross-origin requests might change with time. This can impact how much measurements the Encore system will be able to collect from within future browsers. One relevant example is how modern browsers, to prevent partial encryption, block iframes on web pages served over HTTPS [4]. As iframes are an integral part of the Encore's framework, and the web services are is increasingly adopting HTTPS [25], this can substantially affect Encore's ability to collect accurate measurements. Hence there is room for consistent improvement in the system to be able to perform with contemporary updates in browsers, which serve as the primary vantage points for measurements.

4. DISCUSSION

Based on the evaluation of the systems, there are certain insights that can be derived from the current state of global censorship. Over the years of the Internet expansion, researchers have come up with various detection and measurement methodologies, however, after studying them in detail, we come to the realization that there is no obvious silver bullet that provides complete measurement coverage. Each of the discussed systems has its own domain where it is applicable. This, however, provides us with insight on how we can develop more comprehensive collection mechanisms. Researchers working towards the common goal of measuring Internet censorship can use their individual systems to collect domain specific data, and later on, centralize it into a single global repository. This will allow current researchers to validate the accuracy of their results as well as promote the development of future projects due to the increased data fidelity.

The current systems are more research oriented, developed based on heuristics and certain assumptions. They do not take into account certain corner cases which arise as an implication of measurements in the wild. Moving forward, there is definitely room for improvement, and one way to achieve this is the development of a censorship measurement product. Current variations of developed products include Herdict [3], and the IC Lab platform [35]. While these developments suffer certain deployment limitations, they provide evidence that this is possibly a future direction that requires more attention from both the research and development community.

Measurement systems discussed in the paper cover a span of ten years. ConceptDoppler [20] performed measurements in 2007, Censmon [36] in 2011, and the most recent development, Encore [18], is from 2015. Looking at these systems across different years helps us understand how global censorship has evolved over time. While Concept Doppler, specifically looks at keyword filtering, which is a very basic mechanism, the systems developed in latter years are more generic, and perform measurements to encapsulate all sorts of blocking mechanisms. This indicates that censors continually advance in their blocking techniques and censorship measurement will continue to be an arms race between the advocates of Internet freedom who aim to quantify global censorship and the censoring regimes, which block content.

We also understand the associated challenges while developing measurement systems. Our survey of the developed systems provides us insight that researchers have increasingly focused on overcoming two major technical challenges. The first being the minimization of user involvement in collecting measurement data, and secondly, creating more globally diverse vantage points. Moving forward, these specific challenges should be taken into account while developing future systems. Another major challenge associated with measurement is the disambiguation of various types of blocking, as it requires a lot of manual intervention. Furthermore, crowdsourced reporting suffers from erroneous measurements which lack ground truth data to compare against. While most of these challenges can be addressed by creating more advanced systems there are also certain-non-technical challenges associated with censorship. As censorship has strong ties with the certain political and religious situations, researchers need to be educated about the possible implications of measuring censorship in a specific region. This can be achieved by technologists closely collaborating with activists as acquiring feedback from the locals of a region. In turn, this will aid in the development of more accurate systems and reduce the associated risk of measurement.

Finally, as a reaction to global censorship, individual belonging to censored regions have adopted various circumvention mechanisms. Tschantz et al. perform a detailed study of the research tools developed for the purpose of evading censorship [37]. We believe that censorship measurement and circumvention are greatly overlapping areas, and the study of measurement systems will eventually provide researchers with better insights for developing robust circumvention tools.

5. CONCLUSION

This paper presents an evaluation of the systems and methodologies developed within the research community to measure Internet censorship. It provides a background of the common blocking mechanisms used. It then looks at the individual systems which aim at specifically measuring these techniques. We focus on keyword filtering measurement, detection of URL filters for censorship as well as other distributed server based censorship mechanisms that leverage vantage points within the actual networks. We see that the current approaches still suffer certain limitations and censorship measurement has its associated challenges. Finally, we suggest some insights and plausible future directions for researchers in the field of measuring and evaluating censorship.

6. REFERENCES

[1] Blue coat | network + security + cloud. https://www.bluecoat.com/. (Accessed on 02/19/2017).

[2] Cross-origin resource sharing. https://www.w3.org/TR/cors/. (Accessed on 02/25/2017).

[3] Herdict : Home. http://www.herdict.org/. (Accessed on 02/28/2017).

[4] How to fix a website with blocked mixed content - web security | mdn. https://developer.mozilla.org/en-US/docs/Web/Security/Mixed_content/How_to_fix_website_with_mixed_content. (Accessed on 02/28/2017).

[5] Intel security-mcafee—antivirus, encryption, firewall, email security, web security, network security. https://www.mcafee.com/us/index.html. (Accessed on 02/19/2017).

[6] The long history of censorship. http://www.beaconforfreedom.org/liste.html?tid=415&art_id=475. (Accessed on 03/03/2017).

[7] Opennet initiative. https://opennet.net/. (Accessed on 02/21/2017).

[8] Pakistan's accidental youtube re-routing exposes trust flaw in net | wired. https://www.wired.com/2008/02/pakistans-accid/. (Accessed on 02/24/2017).

[9] Planetlab | an open platform for developing, deploying, and accessing planetary-scale services. https://www.planet-lab.org/. (Accessed on 02/21/2017).

[10] Readability. https://www.readability.com/. (Accessed on 02/21/2017).

[11] Rfc 6066 - transport layer security (tls) extensions: Extension definitions. https://tools.ietf.org/html/rfc6066#section-3. (Accessed on 02/22/2017).

[12] Shodan. https://www.shodan.io/. (Accessed on 02/19/2017).

[13] Tor project: Anonymity online. https://www.torproject.org/. (Accessed on 02/28/2017).

[14] Web filtering software for isp's, business and schools. https://www.netsweeper.com/. (Accessed on 02/19/2017).

[15] Whatweb. https://www.morningstarsecurity.com/research/whatweb. (Accessed on 02/19/2017).

[16] Aceto, G., and Pescapé, A. Internet censorship detection: A survey. *Computer Networks 83* (2015), 381–421.

[17] Aryan, S., Aryan, H., and Halderman, J. A. Internet censorship in iran: A first look. In *FOCI* (2013).

[18] Burnett, S., and Feamster, N. Encore: Lightweight measurement of web censorship with cross-origin requests. *ACM SIGCOMM Computer Communication Review 45*, 4 (2015), 653–667.

[19] Clayton, R., Murdoch, S. J., and Watson, R. N.

Ignoring the great firewall of china. In *International Workshop on Privacy Enhancing Technologies* (2006), Springer, pp. 20–35.

[20] Crandall, J. R., Zinn, D., Byrd, M., Barr, E. T., and East, R. Conceptdoppler: a weather tracker for internet censorship. In *ACM Conference on Computer and Communications Security* (2007), pp. 352–365.

[21] Dainotti, A., Squarcella, C., Aben, E., Claffy, K. C., Chiesa, M., Russo, M., and Pescapé, A. Analysis of country-wide internet outages caused by censorship. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference* (2011), ACM, pp. 1–18.

[22] Dalek, J., Haselton, B., Noman, H., Senft, A., Crete-Nishihata, M., Gill, P., and Deibert, R. J. A method for identifying and confirming the use of url filtering products for censorship. In *Proceedings of the 2013 conference on Internet measurement conference* (2013), ACM, pp. 23–30.

[23] Danezis, G., and Anderson, R. The economics of resisting censorship. *IEEE Security & Privacy 3*, 1 (2005), 45–50.

[24] Duan, H., Weaver, N., Zhao, Z., Hu, M., Liang, J., Jiang, J., Li, K., and Paxson, V. Hold-on: Protecting against on-path dns poisoning. In *Proc. Workshop on Securing and Trusting Internet Names, SATIN* (2012).

[25] Durumeric, Z., Wustrow, E., and Halderman, J. A. Zmap: Fast internet-wide scanning and its security applications. In *Usenix Security* (2013), vol. 2013.

[26] Grefenstette, G., and Nioche, J. Estimation of english and non-english language use on the www. In *Content-Based Multimedia Information Access-Volume 1* (2000), LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE, pp. 237–246.

[27] Hofmann, T. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence* (1999), Morgan Kaufmann Publishers Inc., pp. 289–296.

[28] Katz, S. M. Distribution of content words and phrases in text and language modelling. *Natural Language Engineering 2*, 01 (1996), 15–59.

[29] Landauer, T. K., Foltz, P. W., and Laham, D. An introduction to latent semantic analysis. *Discourse processes 25*, 2-3 (1998), 259–284.

[30] Levis, P. The collateral damage of internet censorship by dns injection. *ACM SIGCOMM CCR 42*, 3 (2012).

[31] Murdoch, S. J., and Anderson, R. Tools and technology of internet filtering. *Access denied: The practice and policy of global internet filtering 1*, 1 (2008), 58.

[32] Nabi, Z. The anatomy of web censorship in pakistan. In *FOCI* (2013).

[33] Nisar, A., Kashaf, A., Uzmi, Z. A., and Qazi, I. A. A case for marrying censorship measurements with circumvention. In *Proceedings of the 14th ACM Workshop on Hot Topics in Networks* (2015), ACM, p. 2.

[34] Park, J. C., and Crandall, J. R. Empirical study of a national-scale distributed intrusion detection system: Backbone-level filtering of html responses in china. In *Distributed Computing Systems (ICDCS), 2010 IEEE 30th International Conference on* (2010), IEEE, pp. 315–326.

[35] Razaghpanah, A., Li, A., Filastò, A., Nithyanand, R., Ververis, V., Scott, W., and Gill, P. Exploring the design space of longitudinal censorship measurement platforms. *arXiv preprint arXiv:1606.01979* (2016).

[36] Sfakianakis, A., Athanasopoulos, E., and Ioannidis, S. Censmon: A web censorship monitor. In *USENIX Workshop on Free and Open Communication on the Internet (FOCI)* (2011).

[37] Tschantz, M. C., Afroz, S., Paxson, V., et al. Sok: Towards grounding censorship circumvention in empiricism. In *Security and Privacy (SP), 2016 IEEE Symposium on* (2016), IEEE, pp. 914–933.

[38] Warf, B. Geographies of global internet censorship. *GeoJournal 76*, 1 (2011), 1–23.

[39] Xu, X., Mao, Z. M., and Halderman, J. A. Internet censorship in china: Where does the filtering occur? In *International Conference on Passive and Active Network Measurement* (2011), Springer, pp. 133–142.

APPENDIX

A. LATENT SEMANTIC ANALYSIS

Latent Semantic Analysis (LSA) [29] is a common technique in the domain of natural language processing that is performed on a set of documents and terms. It enables the extraction of the concepts that are related within the documents by creating a vector representation of a large corpus of text. The vector representation can then be used to calculate similarity with a certain concept vector, by evaluating the distance between the the two component vectors.

This technique involves some pre-processing. As a first step the n documents with a total of m terms are transformed into a m x n matrix known as the term-document matrix, where each of the n columns represent the occurrences of each of the m terms in that specific document. To provide each term a weight based on its importance in the document, a $tf-idf$ (term frequency-inverse document frequency) transformation is applied. The term frequency is calculated as follows:

$$tf = \frac{o_i}{\sum_k o_k}$$

Where o_i is the number of occurrences of the term t_i in the document and $\sum_k o_k$ being the occurrences of all the terms in the document. The inverse document frequency is evaluated using the following equation:

$$idf = log\frac{|D|}{|t_i \in d|}$$

Where $|D|$ is the total number of documents and $|t_i \in d|$ is the count of the documents in which the term t_i appears. The final weighted value for each element in the matrix is the dot product $tf.idf$. This value removes biases of the common terms present within all the documents.

The next step in LSA is to perform a dimensionality reduction using singular value decomposition (SVD). This has two major purposes, the first one is to reduce the the matrix rank and make vector computation easier. The second major purpose is to remove the noise of the terms as a result of the freedom of choice of words that the authors have in the documents. The resultant matrix X_k maps documents to terms, however, this is based on concepts rather than just the weighted counts. The SVD is defined as $X = U\sum V^T$, where U and V are the orthonormal matrices, each implying the correlations between the terms and documents respectively. $\sum$ is the diagonal matrix containing singular values. To perform the reduction, the top k singular values are selected from $\sum_k$, resulting in the final reduced concept matrix to be $X_k = U_k \sum_k V_k^T$.

Once a concept matrix is generated, to calculate the correlation between individual terms, which correspond to the the component vectors in the original term document matrix X, same LSA transformation is applied to the vector. This transforms the term vector into the concept space. To eventually find closely correlation between two concepts, the cosine similarity between the concept vectors is calculated. This summary provides on how LSA can be used to provide insight on how much two concepts relate to each other.