Automated Detection and Fingerprinting of Censorship Block Pages

Ben Jones, Tzu-Wen Lee*, Nick Feamster, Phillipa Gill*
Georgia Tech *Stony Brook University

Abstract

One means of enforcing Web censorship is to return a *block page*, which informs the user that an attempt to access a webpage is unsuccessful. Detecting block pages can provide a more complete picture of Web censorship, but automatically identifying block pages is difficult because Web content is dynamic, personalized, and may even be in different languages. Previous work has manually detected and identified block pages, which is difficult to reproduce; it is also time-consuming, which makes it difficult to perform continuous, longitudinal studies of censorship. This paper presents an automated method both to detect block pages and to fingerprint the filtering products that generate them. Our automated method enables continuous measurements of block pages; we found that our methods successfully detect 95% of block pages and identify five filtering tools, including a tool that had not been previously identified "in the wild".

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]:[Security and protection (e.g., firewalls)]; C.2.3 [**Network Operations**]: [Network Monitoring]

Keywords

Censorship; Internet Measurement

1 Introduction

Internet censorship is pervasive; for example, the OpenNet Initiative's latest measurements detected censorship in 38 of 74 countries [10]. Censorship mechanisms can take many forms, ranging from injected TCP RST packets or DNS responses, to explicit Web pages notifying users that the content has been blocked, or *block pages*. Previous work has developed mechanisms to automatically detect TCP RST packets and altered DNS responses [13, 1], but a significant share of censorship still consists of block pages. To provide a more complete picture of Internet censorship, we must develop automated methods to detect block pages and identify the filtering tools that create them.

Differentiating accessible content from block pages are difficult for several reasons:

1. *Dynamic Content.* Some sites may update content between requests, returning different versions of the same page. Figures 1a and 1b illustrate this effect for `cnn.com`. Block pages may also change (*e.g.*, to display the blocked URL or category), adding yet more problems. Figure 1c illustrates this problem because the block page includes the URL for `cnn.com`.
2. *Content is personalized.* Web sites may personalize content for an individual or region, thereby decreasing the similarity between versions of the same page.
3. *Content is in different languages.* Languages vary across regions, making keyword matching challenging.

Today, those who wish to measure block pages must manually create regular expressions to detect specific block pages and identify filtering tools. Unfortunately, this approach is too slow and resource-intensive to support consistent, continuous measurements because a person must manually create new regular expressions. This process also cannot identify unknown block page templates *a priori*.

In this paper, we present techniques to automatically detect block pages and identify the products that serve them. Our detection technique is based on the insight that it is easier to detect the *difference* between a block page and legitimate content than the similarity between a block page and known block pages. Based upon this insight, we develop a block page detection method that correctly identifies 95% of block pages and 98.6% of accessible pages.

Our fingerprinting technique is based on the insight that block page templates uniquely identify the filtering tool that generated them. Using this method, we identify five known filtering tools, including one that has not been previously observed "in the wild". We extend the work of Dalek *et al.* [2] by automatically identifying filtering tools where possible, and flagging unidentified templates for researchers to label. Since these methods do not require active probing, we can apply them to archival censorship measurements from the OpenNet Initiative and provide the first glimpse into changes in filtering tools across time.

The rest of this paper describes our methods for detecting block pages; techniques for fingerprinting block pages to uniquely identify block page vendors; the accuracy of these detection and fingerprinting methods; and an application of these techniques to five years of measurements of block pages from 49 countries.

2 Background

In this section, we describe various web filtering mechanisms and survey related work.

2.1 Censorship and Block Pages

A censor may return a block page using a variety of mechanisms, such as injecting DNS responses, redirecting traffic through transparent proxies, and inserting packets directly into a TCP stream. In *DNS redirection*, the censor injects a fake DNS response when the user tries to resolve a hostname that contains blocked content, thereby redirecting the user to a server hosting a block page. *Transparent proxies* can provide more granularity than DNS injection by inspecting the content of HTTP streams for restricted keywords or URLs. If the user tries to access restricted content, the proxy could

IMC'14, November 5–7, 2014, Vancouver, BC, Canada.
ACM 978-1-4503-3213-2/14/11.
http://dx.doi.org/10.1145/2663716.2663722

(a) `cnn.com` *on October 24* **(b)** `cnn.com` *on October 30* **(c)** *Block Page for* `cnn.com`

Figure 1: *Differences between accessible pages inhibit block page detection. Figure 1a and Figure 1b are structurally similar, but contain different text and links. Both of these versions of the page differ from the block page in Figure 1c. The block page in Figure 1c also shows that block pages can contain custom content, in this case, the blocked URL,* `cnn.com` *. Variations in block page content make block page detection challenging and time consuming.*

drop the request and return a block page. Because a block page is overt, it is generally safe to assume that a censor who returns a block page to the user is not trying to hide the fact that they are censoring the page and are thus not genearally interested in evading our detection techniques.

2.2 Related Work

OONI [11] is the only other censorship measurement tool that has implemented an automated block page detection method. In Section 4.1, we describe OONI's DOM similarity measure, and in Section 4.2 we compare its method to other block page detection techniques.

Block page detection relates to both document classification and web page classification. *Document classification* aims to classify documents based on features within the documents. The most relevant document classification technique is term-based classification, which clusters pages based on the words in a document [4]. *Web page classification* is a type of document classification that operates on web pages. Web page classification may leverage the semantic content of HTML markup [7], which provides information for visualizing and linking documents. Some classification methods strip the HTML structure from pages and use existing document classification schemes on the stripped content.

Previous work has aimed to identify other types of censorship techniques, such as methods that reset connections or otherwise interfere with connections. Weaver *et al.* detected injected TCP RST packets and tried to isolate the source and purpose of the injected RST packets [13]. They also fingerprinted filtering tools that reset TCP connections. More recently, Weaver *et al.* focus on identifying the existence and purpose of transparent proxies, but did not extend the work to measuring censorship due to user safety concerns [12].

Marqui-Boire *et al.* scanned networks for Blue Coat devices and confirmed the manufacturer by actively probing the devices [5]. Noman *et al.* and the Citizen Lab explored how censorship changes in response to changes in the URL list for a particular product [8, 6]. Dalek *et al.* identified URL filtering tools by network scanning and validated their results with in-network testing [2]. Unfortunately, these methods are time consuming because each measurement requires generating new URLs and inconsistent because network scanning may miss devices that do not have public IPs.

3 Data

We used the OpenNet Initiative (ONI) [10] block page corpus, which has over 500,000 entries. The ONI collected measurements in 49 countries from 2007 to 2012 using locally and globally sensitive URLs as defined by ONI researchers and collaborators around the globe. Each entry in the database corresponds to a single measurement for a URL and contains an uncensored version of the page collected in Toronto, and a test page, collected in the censoring country at approximately the same time. We assume that the Internet in Toronto is not censored and therefore that page represents a known good page. Amongst other data, the dataset has a measurement timestamp, a manually-assigned label indicating whether the test page is blocked or not, the location of the test, and the test network. An anonymized version of the dataset and more information are available online [3].

The ONI dataset has a label for each measurement indicating if the test page was blocked or accessible. To generate this labeling, an ONI staff member generated a regular expression for each block page in each region; about 28,000 test pages were labeled as blocked, and the remaining test pages (about 480,000) were labeled as accessible. We use both sets in our evaluation of detection methods. Because this labeling process created new regular expressions for each new test, the labeling accounts for changes in block pages. Although we identified a few misclassified pages, we have found the labeling to be mostly accurate; these labels have themselves served as a means to identify censorship in previous studies [3] and are thus a reasonable source of an independent label.

4 Block Page Detection

We present methods for detecting block pages based on a simple insight: block pages are less similar to accessible pages than different versions of accessible pages are to one another. Thus, while accessible pages may be non-identical, block pages will exhibit more significant differences. To classify a test page as blocked or not, we compare the test page to a known unblocked version of the page using a similarity measure. Each test page was collected from the region of interest at about the same time as the known unblocked version of the page. To find the best classifier, we evaluated many similarity metrics, including page length, cosine similarity, and DOM similarity (the metric that OONI [11] uses). We evaluated several other document classification methods such as inverse document frequency (IDF), which performed poorly; and

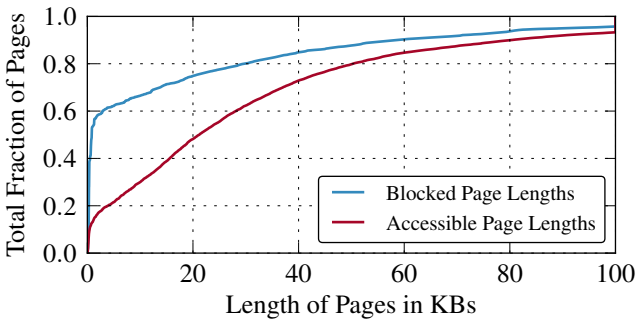


Figure 2: *Block pages are typically smaller than accessible pages. The knee of the blocked page curve is inset.*

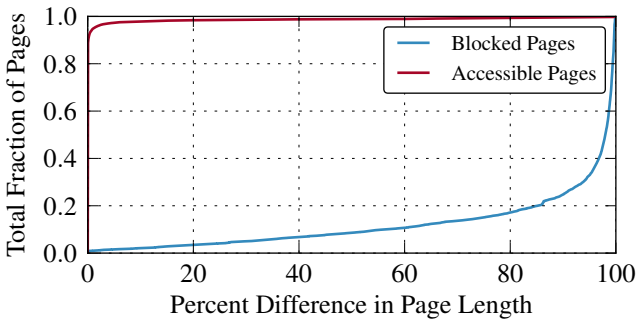


Figure 3: *Differences in page length for blocked and accessible sites imply that the length comparison measure can differentiate between blocked and accessible pages. The knee of the accessible page scores curve is shown inset.*

other methods such as fuzzy hashing. We present the most salient results below.

4.1 Metrics

Length. In the length similarity metric, we compare the sizes of the test page and the known unblocked page. The intuition is that block pages will be smaller than accessible pages, so a test page may be a block page if its size significantly differs from the size of the known good page. This simple approach works well because accessible pages tend to be larger than blocked pages, as Figure 2 illustrates. To compare pages, we compute the page length difference using Equation 1, where $len1$ is the length of the known unblocked page and $len2$ is the length of the test page.

$$\text{Length Percent Diff} = \frac{|len1 - len2|}{max\{len1, len2\}} \quad (1)$$

Cosine Similarity. The cosine similarity metric [4] compares pages based on a term frequency vector, which is a data structure that stores the number of times the words in a document occur. In the context of block page identification, the terms are HTML tags; the term frequency vector stores the number of times each HTML tag appears within a page. Representing a page by its HTML structure allows us to elide most dynamic content, which reduces the variance between accessible pages and (hence) the false positive rate.

DOM Similarity. The developers of OONI [11] proposed comparing the HTML structure of block pages using a DOM similarity measure. The metric creates an adjacency matrix with the probabilities of transitioning between HTML tags. The DOM similarity measure then compresses the adjacency matrix for each page into a vector of its eigenvalues and compares these vectors with a normalized dot product.

4.2 Results

We find that our automated detection methods are accurate, and that the page length similarity measure works best. To evaluate these measures, we compute precision, recall, and false positive rates for each metric using a ten-fold cross-validation and compare precision-recall and ROC curves.

The length comparison measure scored blocked and accessible pages differently, as Figure 3 shows: A threshold that marks any difference in size over 30% as blocked achieves a true positive rate of 95% and a false positive rate of 1.37%. These numbers compare favorably to other similarity measures, as shown in Table 1. Furthermore, the low standard deviation shows that the length comparison metric performed consistently well during cross-validation, implying that these results will generalize to other block pages. Figure 4 shows the precision-recall and ROC curves for each metric, further illustrating that length comparison is the best metric.

5 Block Page Fingerprinting

To fingerprint filtering tools, we identify block page templates and match signatures for each template. Though signature matching for block pages is not new, automated detection of block page templates reduces the effort and increases consistency of filtering product identification.

5.1 Approach

We fingerprint block pages using two features from the block page detection methods: page length and term frequency vectors. Using these features, we cluster the block pages and label each cluster with the filtering tool that generated the template. We assume that filtering tools generate block pages from a template, and that each template is unique to the filtering tool, though a single filtering tool may have many templates. Our analysis of changes in censorship in Section 5.2 and prior work[3] validate this assumption.

We used both term frequency vectors and page length as features for clustering. The intuition behind page length clustering is that block-page templates change at most a few words between different URLs. We used single-link hierarchical clustering to generate clusters on the basis of the block page sizes without knowing the number of clusters *a priori*.

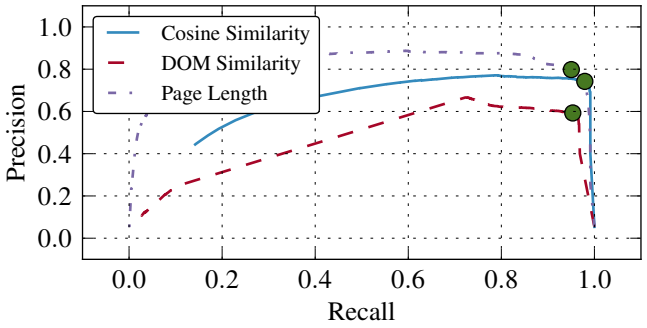
Similarly, the intuition behind term frequency clustering is that the censor will not vary the structure of a block page within the same template. This intuition appears to be accurate because there are only 37 *distinct* term frequency vectors from the 5 years of data. We partitioned the data into different clusters on the basis of unique term frequency vectors.

5.2 Results

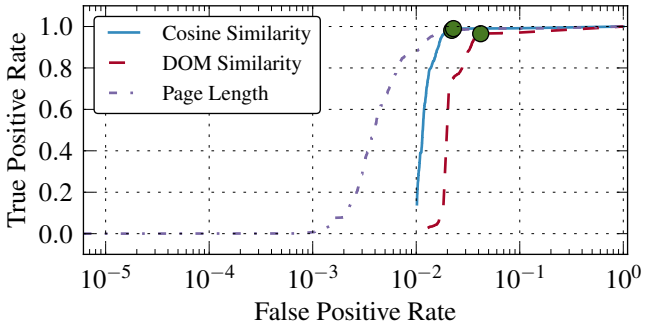
To validate the clusters that our algorithm produced, we compared the clusterings to manually labeled block page templates. We manually identified 27 block page templates and computed the longest common subsequences for each template. We used these subsequences to represent the ground truth for distinct clusters and them to calculate precision and recall for each cluster. In this context, precision is the number of pages in the cluster which come from the same block page template. Recall is the number of pages in the cluster which match a block page template out of all the pages that match a block page template. This method may not capture all block page templates, but this should not be a problem because of our evaluation method. Because we are using precision and recall,

Similarity Measure	True Positive/ Recall (%)	False Positive (%)	Precision (%)	Threshold
Page Length	$95.03 \pm 1.128 \cdot 10^{-3}$	$1.371 \pm 1.829 \cdot 10^{-16}$	$79.80 \pm 1.915 \cdot 10^{-4}$	30.19%
Cosine Similarity	$97.94 \pm 2.341 \cdot 10^{-14}$	$1.938 \pm 3.657 \cdot 10^{-16}$	$74.23 \pm 1.170 \cdot 10^{-14}$	0.816
DOM Similarity	$95.35 \pm 1.242 \cdot 10^{-2}$	$3.732 \pm 1.866 \cdot 10^{-3}$	$59.28 \pm 8.929 \cdot 10^{-3}$	0.995
Diff	99.13	30.95	15.44	n/a

Table 1: *Mean detection rates for similarity measures ± standard deviation are much better than a simple diff.*



(a) *Precision-recall curve for similarity measures.*



(b) *ROC curve for similarity measures.*

Figure 4: *Precision-recall and ROC curves demonstrate that the length comparison measure is the best similarity measure. Green dots mark the selected threshold values on the graph.*

we are evaluating our clustering based upon how well it identifies the templates *we know about*. Our method may also find templates we did not manually identify, but we do not evaluate the quality of those clusters.

To evaluate the quality of each clustering, we computed an F-1 measure for each cluster. The F-1 measure is a common clustering evaluation measure that combines precision and recall equally: $\frac{2 \cdot precision \cdot recall}{precision + recall}$. Finally, we calculated the overall F-1 measure by summing the maximum F-measure for each subsequence: $\sum_{i \in clusters} \frac{n_i}{N} \cdot max_{j \in subsequences}\{f(i,j)\}$, where N is the total number of block pages, n_i is the number of block pages in the i'th cluster, and $f(i,j)$ is the F-measure for cluster i and subsequence j. By taking the maximum F-measure for each subsequence, we associate the block page template, or template subsequence, with the cluster that best matches the template. We then average the F-measures by weighting the F-measure for each subsequence according to the number of pages using that template. Intuitively, this weighting ensures that an outcome with 20 clusters each with one element and an F-measure of 1 and one cluster with 1000 elements and an F-measure of 0.01 does not score well. The F-1 measure scales between 0 and 1; a higher score correlates with higher precision and recall. When the F-1 measure is higher, the clustering strongly corresponds with the identified common subsequences. Because we generate common subsequences from a random sample of block pages and each page matched at most one subsequence, clusters with a high F-measure also strongly correspond to a single template.

Term frequency clustering performs well, with an F-1 measure of 0.98; clustering based on page length is much worse, with an F-1 measure of 0.64. This result makes sense because block pages are generated from a template. Therefore, they often share the same structure. Term frequency clustering identifies block page templates despite noise introduced by pages mislabeled as blocked and a significant amount of noise introduced by standard HTTP error messages. For instance, the data set included a large number of HTTP 302 and 404 responses. On the other hand, we were surprised that page length produced poor-quality clusters. Block

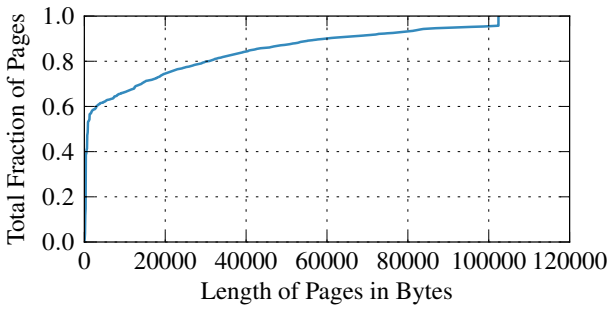


Figure 5: *Block pages have only a few distinct lengths.*

page templates only replace small amounts of text (*e.g.*the URL of the blocked Web site), so we expected similar page lengths within a cluster. Interestingly, Figure 5 shows that block pages have few distinct sizes; there are fewer clusters than templates.

Fingerprinting Filtering Tools. Where possible, we label each cluster according to known signatures (*e.g.*, from previous work [2]); otherwise, we attempt to manually identify the block page vendor based on features of each template. Using this method, we identified five filtering tools that generated 7 out of 36 clusters from the dataset. In these cases, copyright notices within HTML comments, HTTP header fields, or other signatures offered a definitive identification. The remainder of the clusters had no identifying information in the fingerprint, although unique HTTP headers indicated the use of a distinct tool. Table 2 summarizes these results.

6 Case Studies

We now apply our detection and fingerprinting techniques to the ONI dataset to explore how the use of various block page methods has evolved over time. Unfortunately, the ONI did not continuously gather measurements, so we can only explore measurements from small snapshots. Fortunately, each snapshot contains enough measurements to make inferences about the presence of specific filtering tools. Because the censor could always choose to return nothing or

Number of Clusters	Product Manufacturer	Network	Time Frame	Fingerprint
2	FortiGuard	AS 24090 (Malaysia)	2009	Block page contains the text "Powered by FortiGuard"
1	Squid Proxy Server	AS 2609 (Tunisia)	2010	HTTP Headers contain the text "Server: squid/2.6.STABLE16"
1	Netsweeper	AS 12486 (United States), AS 15802 (United Arab Emirates), and AS 12586 (Yemen)	2010-2012	"webadmin/deny" in URL, which indicates that Netsweeper is in use [2]
1	Websense	AS 29584 (Azerbaijan)	2010	Websense copyright disclaimer is included in HTML comments
2	WireFilter	AS 25019 (Saudi Arabia)	2011	HTTP Headers contain the text "Server: Protected by WireFilter"

Table 2: *Filtering tools identified from block page templates*

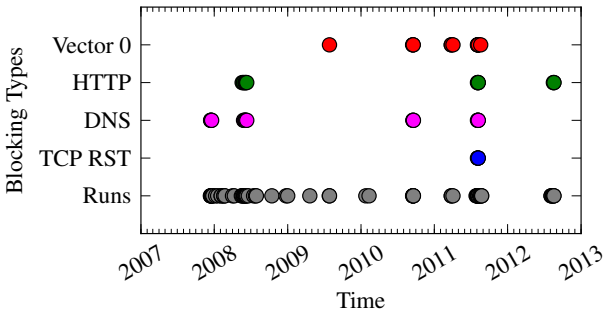


Figure 6: *Filtering mechanisms used in AS 18399 in Burma.*

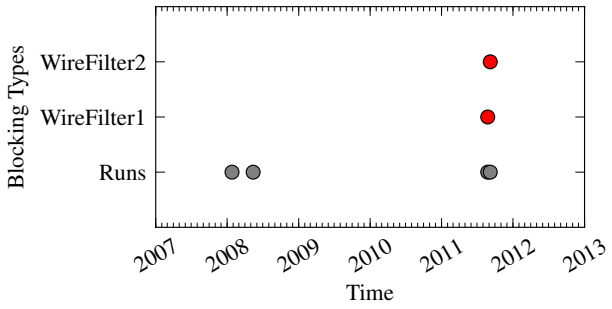


Figure 7: *The appearance of two block page templates in AS 25019 marks the use of WireFilter, a new filtering tool in Saudi Arabia.*

change the block page, the absence of a block page cluster does not indicate that the given filtering tool is no longer in use. To account for this ambiguity, we include other types of blocking to give further insight into changes in filtering tools and capabilities. Specifically, vectors refer to block page templates, HTTP indicates no response to an HTTP request, DNS corresponds to either manipulation by redirection or the lack of a response, TCP RST refers to TCP RST filtering, and runs shows when measurements were taken.

We also extend our analysis by determining if block pages are the result of DNS redirection or not. We assume that the censor can only use a few IP addresses to host block pages, so if DNS redirection is in use, we expect the block pages to resolve to a limited number of IP addresses. Though the number of resolved IP addresses can fluctuate due to CDNs, DNS redirection should return significantly fewer IP addresses than the number of distinct URLs measured.

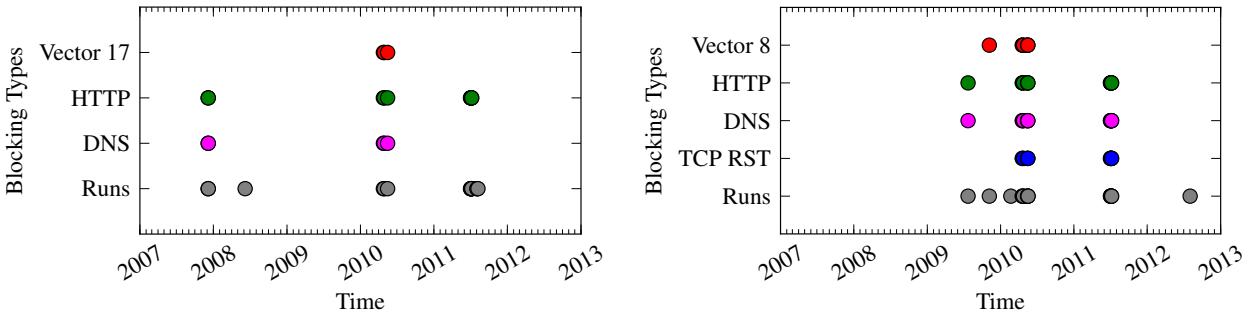
Political Shifts (Burma). Analyzing changes in filtering mechanisms and block pages in Burma (Myanmar) provides insight into how censorship evolves as filtering tools and regimes change. Figure 6 shows the censorship enforcement mechanisms and block page clusters used in AS 18399 in Burma between 2007 and 2012. Until mid 2009, AS 18399 used DNS redirection as a form of censorship. In mid-2009, a custom block page template for AS 18399, vector 0, appears. Because the block pages in vector 0 resolved 568 URLs to 659 IPs, vector 0 does not appear to be using DNS redirection. Unfortunately, we could not identify the product behind vector 0, but these results indicate that AS 18399 in Burma may have acquired a new filtering tool in mid 2009. In late 2011, Burma underwent a massive political shift and significantly reduced the extent of censorship [9], which may be reflected in the lack of detected block pages after this time.

New Filtering Tools (Saudi Arabia). We observed that Saudi Arabia, like many countries, has upgraded its censorship equipment in recent years. Figure 7 illustrates this shift for AS 25019. Although we do not know what type of filtering equipment was used in AS 25019 prior to 2011, we can conclude that a new filtering tool, WireFilter, begins censoring content in 2011. Oddly, WireFilter appears to be using multiple block page templates concurrently, which implies that multiple devices are in use.

Different Techniques in Different ISPs (Thailand). Different ISPs implement Thailand's censorship mandate differently. Figure 8 illustrates that AS 9737 and AS 17552 use different filtering tools and mechanisms to enforce censorship.

Figure 8a shows that AS 9737 has changed censorship mechanisms over time. The first set of measurements in 2008 show that AS 9737 uses DNS redirection and drops HTTP requests to censor content. In 2010, AS 9737 starts using a new filtering tool, represented by vector 17. We could not identify vector 17, but it appears to be a transparent proxy because the 30 URLS blocked by vector 17 resolved to 48 unique IP addresses, indicating that DNS injection was not used. It appears that AS 9737 is also trying to mask the identity of its filtering software because all HTTP headers for vector 17 contain the string "Server: Apache/2.2.9 (Debian)".

Figure 8b shows that AS 17552 and AS 9737 use different censorship enforcement mechanisms though they have both changed their filtering over time. Our data shows that AS 17552 switched from DNS redirection to a new filtering tool in late 2009, shown by vector 8. Vector 8 does not appear to be the result of DNS redirection because its 19 URLS resolved to 28 IP addresses. AS 17552 also appears to obfuscate the identity of their filtering tool because all HTTP headers for the block page contain the string "Server: Apache".



(a) *Filtering mechanisms used in AS 9737 in Thailand*

(b) *Filtering mechanisms used in AS 17552 in Thailand*

Figure 8: *ASes 9737 and 17552 show that government mandated censorship can vary by ISP*

Though ASes 9737 and 17552 use different block page templates, they may be using different configurations of the same filtering tool. Vectors 8 and 17, the block pages for ASes 9737 and 17552, have different structures, as vector 8 uses tables for layout and is around 6000 bytes in length, whereas vector 17 uses div tags for layout and is around 1000 bytes in length. Despite these differences, the filtering tools both appear to be transparent proxies, the filtering tools return similar HTTP headers, and both block pages contain similar strings such as "The page you are trying to visit has been blocked by the Ministry of Information and Communication Technology" (vector 8) and "This website has been blocked by ICT" (vector 17).

7 Conclusion

We developed block page detection and filtering tool identification techniques to enable scalable, continuous, and accurate censorship measurements. Using these techniques, we built a block page detection method with a 95.03% true positive rate and a 1.371% false positive rate and a block page identification method which correctly identified block page templates for 5 known filtering tools. These methods significantly improve the state of the art in censorship measurement and set the stage for the next generation of censorship measurements. Because the vendor and product behind many clusters remains unidentified, future work could include fingerprinting existing block page products and using the fingerprints to find more template matches.

Acknowledgments

We would like to thank the OpenNet Initiative and the Citizen Lab for the use of their data. This work was supported by a Google Faculty Research Award, a Google Focused Research Award, and NSF awards CNS-1111723 and SaTC-1350720.

References

[1]: J. Crandall, D. Zinn, M. Byrd, E. Barr, and R. East. Conceptdoppler: A weather tracker for internet censorship. In *Proceedings of the 14th ACM Conference on Computer and Communications Security*, CCS '07a, pages 352–365, New York, NY, USA, 2007. ACM. (Not cited.)

[2]: J. Dalek, B. Haselton, H. Noman, A. Senft, M. Crete-Nishihata, P. Gill, and R. J. Deibert. A method for identifying and confirming the use of URL filtering products for censorship. In *IMC '13: Proceedings of the 2013 conference on Internet measurement conference*. ACM Request Permissions, Oct. 2013. (Not cited.)

[3]: P. Gill, M. Crete-Nishihata, J. Dalek, S. Goldberg, A. Senft, and G. Wiseman. Characterizing censorship of web content worldwide: Another look at the opennet initiative data. `http://www.cs.stonybrook.edu/~phillipa/papers/ONIAnaly.html`, 2013. (Not cited.)

[4]: E. H and G. Karypis. Centroid-based document classification: Analysis & experimental results. Technical Report 00-017, University of Minnesota, 2000. (Not cited.)

[5]: M. Marqui-Boire, J. Dalek, S. McKune, M. Carrieri, M. Crete-Nishihata, R. Deibert, S. O. Khan, H. Noman, J. Scott-Railton, and G. Wiseman. Planet blue coat: Mapping global censorship and surveillance tools. Technical report, The Citizen Lab, January 2013. (Not cited.)

[6]: H. Noman and J. C. York. West censoring east: The use of western technologies by middle east censors, 2010-2011. Technical report, The OpenNet Initiative, March 2011. (Not cited.)

[7]: X. Qi and B. D. Davison. Web page classification: Features and algorithms. *ACM Comput. Surv.*, 41(2):12:1–12:31, Feb. 2009. (Not cited.)

[8]: The Citizen Lab. Behind blue coat: Investigations of commercial filtering in syria and burma. Technical report, The Citizen Lab, November 2011. (Not cited.)

[9]: The OpenNet Initiative. Burma (myanmar). `https://opennet.net/research/profiles/burma`. (Not cited.)

[10]: The OpenNet Initiative. The opennet initiative. `https://opennet.net`. (Not cited.)

[11]: The Tor Project. Ooni: Open observatory of network interference. `https://ooni.torproject.org/`. (Not cited.)

[12]: N. Weaver, C. Kreibich, M. Dam, and V. Paxson. Here be web proxies. In M. Faloutsos and A. Kuzmanovic, editors, *Passive and Active Measurement*, volume 8362 of *Lecture Notes in Computer Science*, pages 183–192. Springer International Publishing, 2014. (Not cited.)

[13]: N. Weaver, R. Sommer, and V. Paxson. Detecting forged tcp reset packets. In *Presented as part of 16th Annual Network & Distributed System Security Symposium*, 2009. (Not cited.)