On the Importance of Encrypted-SNI (ESNI) to Censorship Circumvention

Zimo Chai
University of Massachusetts Amherst

Amirhossein Ghafari
University of Massachusetts Amherst

Amir Houmansadr
University of Massachusetts Amherst

Abstract

With the increasing use of TLS encryption over web traffic, the censors start to deploy SNI filtering for more effective censorship. Specifically, a censor can identify the web domain being accessed by a client via the SNI extension in the TLS ClientHello message. In response, in August 2019, a new extension called ESNI (Encrypted-SNI) is proposed for TLS 1.3, aiming at fixing this server name leakage.

In this paper, we first characterize SNI-based censorship in China by measuring its prevalence and effectiveness. We outline its assisting role in censorship by comparing it with other commonly used censorship methods. We then measure the deployment prevalence of ESNI and further analyze its current and potential effectiveness in censorship circumvention. We also monitor the censorship associated with ESNI from 14 areas all around the world. Based on our analysis, we discuss the key factors to the success of ESNI and potential problems in a post-ESNI era. We hope our work will make ESNI a more promising and effective censorship circumvention strategy.

1 Introduction

With the increasing fraction of web traffic encrypted with TLS [22], more and more censors start using SNI filtering to constrain users' Internet access [11,34]. Specifically, as shown in Figure 1, a censor can learn the website a client is trying to access via the server name indication (SNI) extension [12] in the TLS ClientHello message. In response, a new extension called ESNI (Encrypted-SNI) is recently proposed for TLS 1.3, fixing this decade long hostname leakage issue. Since the first Internet draft of ESNI rolled out, Internet freedom communities have expressed great interest in it, considering it as "the biggest thing since the ascendance of TLS" [1, 19].

In this study, we take China as the major studying country, answering two important questions. First, comparing to other common censorship methods, what role does SNI-based censorship play? Second, what is the current and potential effectiveness of ESNI in censorship circumvention?

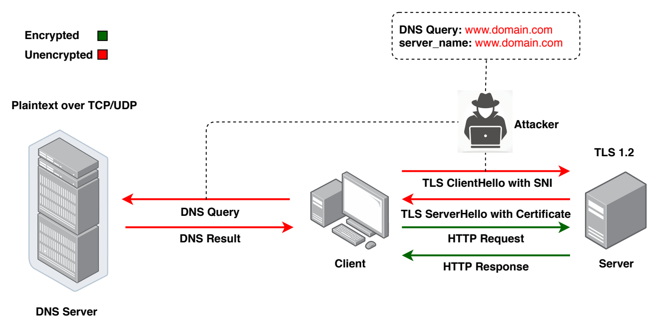

Figure 1: Unencrypted DNS channel and TLS 1.2.

In order to have a better understanding of the role played by SNI filtering in censorship, we measure how the Alexa top 1 million websites are censored by different censorship techniques in China (§3.1). We select China as the main studying country because of its infamous sophistication and comprehensiveness in censorship methods [13, 25, 35]. Our findings outline the overlapping relationships between different censorship methods, revealing the assisting role SNI filtering plays in China's censorship (§4.1). Our experiment results also show that 84.5% of the blocked websites are under IP blocking, indicating a large portion of the sites will remain blocked even when SNI-based censorship is circumvented.

Based on the understanding of SNI filtering, we did the first evaluation on the use of ESNI as a censorship circumvention strategy. In specific, we measure the deployment prevalence of ESNI as well as its effectiveness in censorship circumvention. From the results, we find around 10.9% of the Alexa top 1 million sites are already supporting ESNI (§4.2). Furthermore, while using ESNI along with encrypted DNS channel helps to unblock only 66 sites currently censored in China, we argue the deployment of ESNI is still a progressive move as it essentially makes more than 101 thousand websites more censorship-resistant. We also find at least 85 websites hosted on CDNs are indeed blocked by IP, suggesting the collateral damage of CDNs may be overestimated (§4.3).

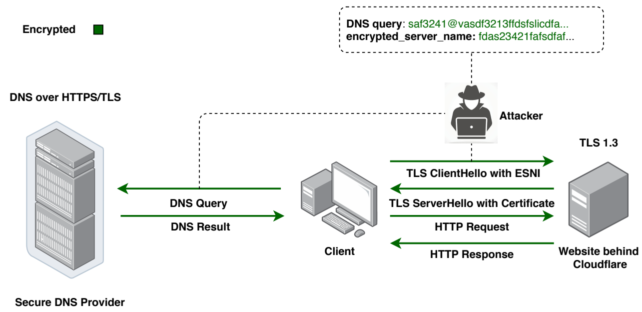

Figure 2: Encrypted DNS channel and TLS 1.3 with ESNI.

The fate of ESNI deeply depends on whether any censorship action has been taken against it before it becomes an essential part of the Internet (§5.1). We, therefore, monitor censorship associated with ESNI in 14 different areas across the globe (§3.2). Contrary to a report claiming ESNI traffic is already blocked in South Korea [11], no anomaly associated with ESNI censorship is detected from all our tests (§4.4).

Finally, based on the findings from our experiments, we discuss the key factors to the success of ESNI as a censorship circumvention strategy (§5.1). We further leave notes on new challenges we may face when ESNI becomes an essential part of the Internet (§5.2).

2 Background

SNI-based censorship. The Server Name Indication (SNI) extension inside the TLS ClientHello message is used to tell a web server which website's certificate should be given to the client. Since a ClientHello message is always sent before the establishment of TLS encryption channel, it remains in cleartext. Consequently, censors can determine the website to which a client is trying to connect via the SNI extension.

Two most commonly used strategies to avoid this SNI leakage are domain fronting [20] and omitting SNI [22]. Domain fronting sets the SNI value to popular services within the same cloud infrastructure as the intended website. It then specifies the intended website in the HTTP header which will be transferred over established TLS encryption channel. However, with announcements [2, 4, 5] that cloud providers plan to disable the domain fronting usage on their infrastructures, this method becomes less viable. Omitting SNI, as another way to evade SNI-based censorship, is widely used by censorship circumvention tools, including Psiphon, Lantern, and Massbrowser [38]. Since the server does not receive an SNI extension or an empty one, it provides a general certificate to establish the TLS connection [38]. Frolov et al. [22] report that as of August 2018, only 1.41% of the TLS ClientHello messages do not contain a SNI extension, indicating omitting SNI strategy may be fingerprinted by censors.

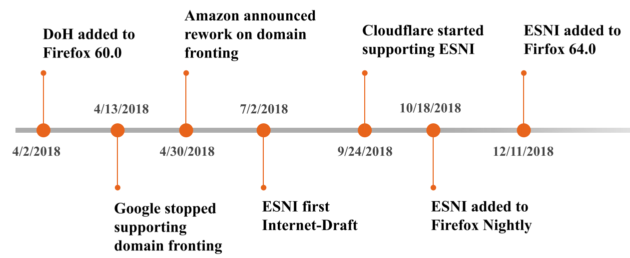

Figure 3: Deployment timeline of ESNI.

ESNI mechanism. We note our introduction is mainly based on the third Internet draft of Encrypted-SNI (ESNI) [30], which is subject to changes. In general, ESNI works as follows. First, as shown in Figure 2, the client acquires the public ESNIKey associated with destination server via a trusted channel. While the trusted channel is left unspecified, it can typically be an encrypted DNS channel. After obtaining the ESNIKey, an encryption key is derived using both ESNIKey and a key chosen by the client. The client then sends a TLS ClientHello message with the server name encrypted by this derived encryption key. The server can then decrypt the encrypted server name. The rest of the connection is similar to a typical TLS 1.3 connection [29].

It is worth noting that clients with ESNI enabled must not fall back to cleartext SNI [30] since, otherwise, censors can simply block all ESNI traffic without worrying about the disruption to legitimate traffic.

ESNI adoption timeline. We have summarized ESNI related events in Figure 3. On September 24, 2018, Cloudflare CDN announced that they had enabled ESNI support on all of their platforms [3]. A week later, Firefox started to support ESNI in their Nightly release and ESNI has been available as a non-default feature in the stable releases since Firefox 64. While there is no official news about Google Chrome's plan to support ESNI, one of the Chromium developers claimed that they expected to add this feature to BoringSSL and Chromium by the end of 2019 [7]. On the other hand, many application and TLS library developers, such as Golang's crypto/tls library, prefer waiting for the widely deployment and adoption of ESNI before implementing it.[1] Meanwhile, no plan to support ESNI are announced by other CDN providers.

3 Methodology

Our evaluation on the use of ESNI in censorship circumvention consists of two main experiments. First, we characterize the role of SNI filtering, comparing it with other commonly used censorship techniques in China. In the second experiment, we measure the prevalence and effectiveness of ESNI.

[1] https://golang.org/pkg/crypto/tls/

3.1 Censorship Techniques Used by China

To understand the role SNI filtering plays in China's censorship, we measure how different censorship techniques are used to block the Alexa top 1 million websites. We select China as the testing country due to its sophistication and comprehensiveness in censorship techniques.

Vantage points. Unless specified, we do experiments on three virtual private servers (VPSes) running Ubuntu 16.02 LTS, located in US, Hong Kong and Mainland China. We note here that the methods to detect DNS injection and SNI filtering do not require control over any host inside China. In fact, any server located in China that accepts a full TCP 3-way handshake can help with the SNI filtering detection; any Chinese IP can help with the experiment of DNS injection. We decide to use servers under our control to conduct the experiment in an accurate, efficient and ethical way.

Detecting DNS injection. Similar to the method used in [13], we send DNS queries from US to China. Since the destination server in China has no DNS resolving or forwarding functionality, we expect any DNS response to be actually injected by the GFW (Great Firewall of China). We, therefore, can learn a certain domain is censored by DNS injection. We prepend "www." to any domain without this prefix, as we observe the GFW only reacts to these refinements for certain domains. This observation confirms the finding in [13].

Detecting SNI filtering. Taking the advantage of the bi-directional feature of the GFW [33], we probe the GFW remotely from US by sending TLS ClientHello messages with various SNI values. We configure the destination server in China to accept TCP handshake requests but will never tear down a connection before receiving a FIN or a RST packet. Therefore, we expect any RST packet sent to our machine in US before the probing tool times out to be actually injected by the GFW. We, thus, can learn whether a domain is censored by SNI filtering.

We note that a complete TCP 3-way handshake is required before a TLS ClientHello message can trigger the GFW, confirming the GFW is now of full state [16]. We also observed a 60 seconds residual censorship period after the first RST sent by the GFW. During this period, any SYN packet associated with the 3-tuple (src IP, dst IP, dst port) will trigger a forged SYN/ACK with incorrect sequence number; any other packets will trigger the GFW to send multiple RSTs to both ends. The duration of this period was reported to be 90 seconds in previous work [35]. To avoid false positive caused by the residual censorship, we make sure a different destination port is used for each probing within a 60 seconds timing window.

Detecting IP blocking. To reduce the DNS resolving overhead during the probing time, we first resolve each domain to its ultimate answer via a VPS in Hong Kong. When multiple IPs are in one answer, we only select the first IP address.

We then use Nmap [24] and `masscan` to SYN-ping the port 80 and port 443 for each IP address from both US and China.[2] For Nmap, we use its default T3 timeout template. For `masscan`, since its sending and receiving threads are independent from each other, the timeout can be as long as the duration of one full scan. We mark an IP as filtered when we observe no open port from China, but indeed observe open ports from US control group. Finally, we mark any domain name on the extracted IP blacklist as blocked by IP.

Accuracy issues. Facing the same issue as in many previous works [13], we observe some non-negligible false negative in all three detection experiments. For example, a forged DNS reply serves as a ground truth that the associated domain is censored. If no forged reply is triggered by the same DNS query in another test, we know the detection is false negative. Thanks to the extremely fast speed of our censorship detection tools, we address the issue by running multiple independent tests per day. We manage to bound the false negative rate of all detection to 6.15×10^{-9} or lower. We also observe a 4.99×10^{-5} false positive rate in the SNI filtering test and manage to reduce it by repeating the experiment after the false negative rate has been bounded.

Limitations. Our work detects censorship from a limited number of vantage points. Ideally, a complete bipartite graph between all clients and all servers should be formed to show a comprehensive picture of the censorship. However, we note that no geo-location inconsistency was found in Chinese censorship by previous work [17].

We resolve domains to their ultimate answers from Hong Kong, rather than Mainland China, to make sure the DNS responses are not injected or poisoned by the GFW. However, authoritative name servers with GeoDNS enabled may return different answers to machines in different geo-locations. Although Hong Kong is geographically close to Mainland China, we still further mitigate the potential geo-location bias. In particular, we send DNS queries to a popular recursive resolver with ECS (EDNS-Client-Subnet) feature disabled. This way, since the authoritative servers can only assign answers based on the IP of the egress resolver, the answers we get are more consistent with any client using the same egress resolver.

We mark a domain as censored by IP blocking as long as one of its IP addresses is censored. Ideally, all IP addresses associated with a domain should be tested and the percentage of its IPs blocked should be reported.

3.2 Prevalence and Effectiveness of ESNI

We now describe our approaches to evaluating the prevalence of ESNI and its effectiveness in censorship circumvention. In particular, we measure which websites among the Alexa top 1 million support ESNI and measure if any tested country is blocking ESNI traffic already.

Debugging page. Cloudflare, as currently the only known CDN provider supporting ESNI, offers an informative de-

[2]masscan: https://github.com/robertdavidgraham/masscan

bugging page for every website it hosts. Specifically, for a given domain named example.com, the path to its debugging page will be https://example.com/cdn-cgi/trace. The debugging page shows information about the current connection, including the SNI status (e.g, plaintext, encrypted or off), the hostname and the TLS protocol version.

Testing tools. Firefox 64.0 is the first stable release that supports ESNI. We control it with the help of Geckodriver 0.24.0 and Python3 Selenium library.[3] We configure a Firefox profile that strictly enables ESNI and DNS-over-HTTPS (DoH) and sets https://1.1.1.1/dns-query as the URI for DoH.

ESNI prevalence measurement. For better efficiency, we first use curl to check the existence of the debugging page for the Alexa top 1 million websites from a VPS located in US.[4] We then let Firefox automatically browse and save those discovered debugging pages. We mark a website as supporting ESNI when the string "sni=encrypted" appears on its debugging page. We repeat the test multiple times for each site to bound the false negative rate.

ESNI censorship measurement. We monitor ESNI related censorship from 14 different areas all around the world, including Mainland China, Hong Kong, South Korea, Japan, Singapore, Indonesia, India, Iran, United Arab Emirates, France, Netherlands, UK, US and Canada. In particular, we let Firefox browse around 101K ESNI-supported websites and check if there is any connection disruption associated with ESNI censorship.

Due to the difficulty of getting a South Korean VPS, we conduct the experiment from US and tunnel all traffic to South Korea via a VPN vantage point. Considering the geo-location of VPN services may be falsely advertised [36], we carefully confirm our vantage point is under the same censoring network used by residential South Koreans. Specifically, SNI-based censorship is detected when we access sites known to be censored in South Korea via the VPN vantage point. We note that for the other 13 areas, we do the detection tests on VPSes, rather than through VPNs.

Limitations. Our work uses one single vantage point in each area to detect censorship associated with ESNI. Ideally, a complete bipartite graph between all clients and all ESNI supported websites should be formed to show a comprehensive picture of the ESNI censorship.

We send DNS TXT record queries to one recursive resolver in each area, checking if the expected ESNIKey is provided. Ideally, all combinations of the 3-tuple (client IP, DNS resolver, domain name) should be tested.

4 Results

In this section, we discuss the key findings of the study. We outline the relationships between various censorship techniques and characterize the role of SNI filtering. We analyze the prevalence and effectiveness of ESNI. We also report no anomaly associated with ESNI censorship were detected in any area we tested.

[3] Selenium: https://www.seleniumhq.org/

[4] curl: https://github.com/curl/curl

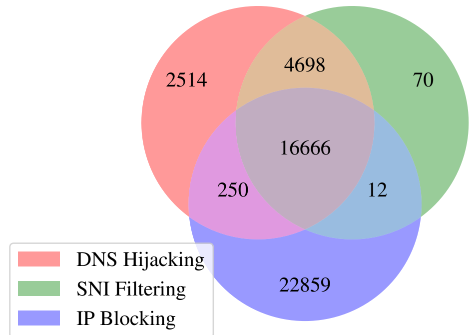

Figure 4: Top 1M sites under different censorship methods.

IP blocking. We managed to get 539,456 unique IP addresses by resolving 1 million domains multiple times and selecting only the first IP in the answer for each domain. The relatively low number of unique IP addresses reveals the high co-hosting rate of websites on the Internet [32].

We then identify 39,787 sites are blocked by IP in China, revealing IP blocking is the predominant censorship technique. In other words, assuming both SNI filtering and DNS hijacking censorship have been circumvented, there are still around 84.5% websites cannot be accessed in China because their IPs are blocked. This observation reminds us censorship circumvention techniques that can hide true destination IP addresses (e.g, proxy [15], decoy-routing [14, 21, 27, 37]) are still of superior importance.

4.1 Characterizing Censorship Techniques

As shown in Figure 4, a large number of websites are exclusively blocked by IP. We argue, from three different points, that those websites are not intended to be censored, but are suffering from the collateral damage of IP blocking. First, for those websites we sample, we cannot identify the censoring motivations except they are co-hosted with some sensitive websites. Second, we can clearly observe from Figure 5 that, while the trend of different censorship techniques is same, the number of censored websites caused by IP blocking is significantly higher than the other two censorship techniques. Third, due to the simple and dynamic nature of DNS hijacking, it has been used as the primary censorship approach [13, 33]. Thus, the absence of those websites from the DNS hijacking blacklist strongly suggests they are not intended to be blocked by the censor.

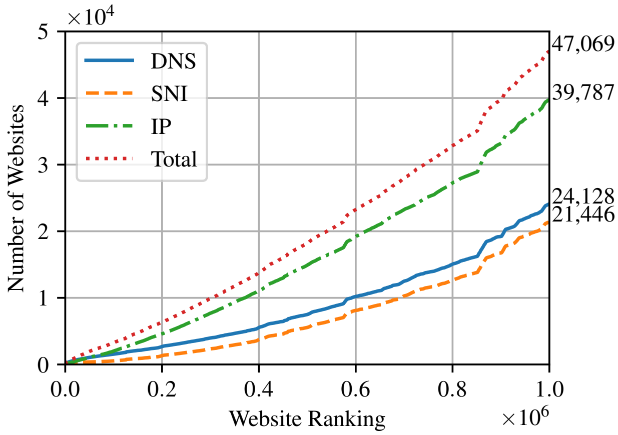

Figure 5: Number of censored sites associated with ranking.

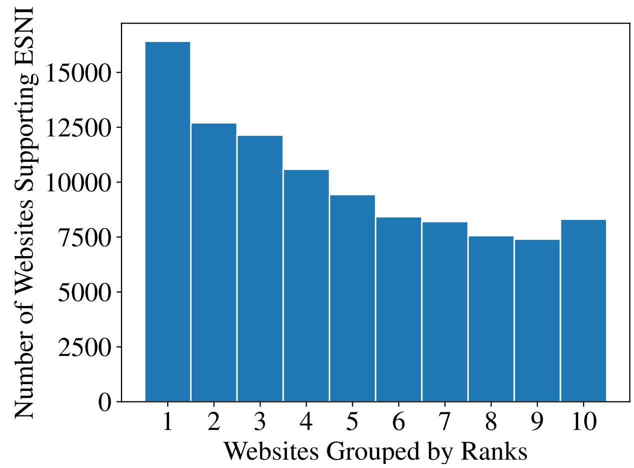

Figure 6: ESNI supported sites aggregated by ranks.

SNI filtering. We detect 21,446 sites under SNI filtering censorship in China. One interesting observation is that only 70 websites are exclusively censored by SNI filtering. In other words, SNI filtering is almost always used in a combination fashion with other censorship methods. This phenomena strongly indicates SNI filtering is playing an assisting role in China's censorship, serving as the second gatekeeper in case DNS hijacking censorship is circumvented.

While the majority of websites under DNS hijacking and SNI filtering overlap, we find 2,764 sites that are under DNS hijacking but not SNI filtering. Further investigation reveals that some of those websites do not support HTTPS, making SNI filtering inapplicable. However, we indeed find HTTPS websites that are exclusively censored by DNS hijacking. This observation not only implies using DoH along with HTTPS can be an effective strategy to unblock many websites, but also reveals that two different blacklists are maintained by the GFW for SNI filtering and DNS hijacking. Further, although this simple strategy can be easily blocked by adding corresponding domains into the existing SNI blacklist, the lack of actions from censors suggest the inconsistency on GFW administration and may also reflect that censors are relatively satisfied with the current effectiveness of DNS censorship.

4.2 The Prevalence of ESNI

The core idea of using ESNI as a censorship circumvention strategy is to put the censor into a censoring all ESNI traffic or none dilemma [18]. Therefore the prevalence of ESNI can significantly affect the cost for the censor to block it [19]. One deterministic factor to the amount of traffic using ESNI is the number of websites supporting ESNI. We, therefore, measure and evaluate the deployment prevalence of ESNI by checking the ratio of websites supporting ESNI among the Alexa top 1 million sites.

First, we find 109,322 sites have a Cloudflare debugging page, meaning around 10.93% of the top 1 million sites are hosted on Cloudflare CDN. Further, we determine that around 92.56% sites behind the Cloudflare CDN support ESNI. As shown in the Table 1, the remaining 7.44% sites only support SNI, or do not support SNI at all. Part of the reason why those websites do not support ESNI is that they are not using TLS 1.3 or do not support TLS at all. However, there are still 1.17% sites using TLS 1.3 but not supporting encrypted SNI, which we are not aware of the reasons behind it.

SNI Status	TLS Version	Number	Portion
encrypted	TLS1.3	101,190	92.56%
plaintext	TLS1.3	1,288	1.17%
	TLS1.2	6,825	6.24%
off	TLS1.2	5	0.005%
	-	14	0.012 %
Total		109,322	100%

Table 1: SNI and TLS status of sites behind Cloudflare CDN.

We divide the Alexa top 1 million sites into 10 groups by their rankings, with group 1 representing websites ranking from 1 to 0.1 million. Figure 6 shows a descending trend of the number of websites supporting ESNI with the increase in ranking range. This result matches our expectation as popular websites need a better quality of service for their visitors and will consequently be more likely to use a CDN to host their sites. As the ranking of a website is strongly correlated with the amount of traffic it receives [31], Figure 6 may also indicate more TLS traffic can benefit from ESNI.

4.3 The Effectiveness of ESNI in China

The key motivation of using ESNI is to prevent server names from leaking, therefore, it is pointless to discuss ESNI if server names can be leaked via DNS channel. We thus assume an encrypted DNS channel exists when analyzing the effectiveness of ESNI. We note this assumption implies the DNS hijacking censorship has been successfully circumvented.

Effectiveness in unblocking sites. Figure 7 demonstrates the overlapping relationships among sites under SNI filtering, sites under IP blocking and sites supporting ESNI. ESNI can successfully unblock a website under SNI filtering only if the IPs of the site are not blocked. Therefore, represented by the golden color area, only 66 websites can be unblocked with the help of ESNI currently. However, we argue that the deployment of ESNI is still a very progressive and meaningful move as it makes more than 101K websites more resistant to the potential censorship in the future.

A few CDN's IPs are blocked. Although Zolfaghari et al. [38] state that websites assigned dedicated IP by CDN providers are vulnerable to IP filtering, no IP filtering on CDN edge servers were detected by previous work [23]. We, however, indeed observe 47 IP addresses belonging to Cloudflare CDN get blocked, resulting in at least 85 sites censored. This finding not only advances our understanding of the behaviors and willingness of censors but also suggests the collateral damage of CDNs may be overestimated by circumvention designers.

4.4 No ESNI-based Censorship Detected

It is vital to the success of ESNI in censorship circumvention that no prevailing censorship actions are taken against it before it becomes too expensive to censor. In other words, if ESNI traffic got censored in its early stage, clients and websites will be less motivated to deploy it. A rumor claims that ESNI traffic is already blocked in South Korea [11]. However, we note that no censorship associated with ESNI is observed from our experiments. To be more specific, we are able to successfully access websites using ESNI from all 14 areas we tested, which includes Mainland China, Hong Kong, South Korea, Japan, Singapore, Indonesia, India, Iran, United Arab Emirates, France, Netherlands, UK, US and Canada. We further state no anomaly is observed when obtaining ESNIKeys from the recursive resolvers via DNS TXT record queries.

5 Discussions

5.1 The Success of ESNI

Be big. Fifield [18] summarizes the nature of circumvention as forcing "the censor to trade false positives for false negatives". To one extreme, the fundamental motivation behind

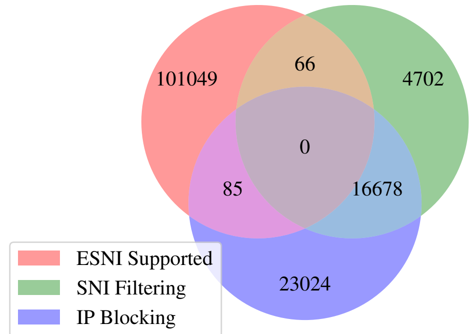

Figure 7: Relationships among censored sites and ESNI supported sites, assuming DNS-based censorship is evaded.

ESNI is to put censors into a dilemma, where they either censor all ESNI traffic or none [19]. Therefore, the success of ESNI crucially depends on its predominant adoption. More precisely, ESNI should be used in a significant amount of TLS connections, rather than exclusively used for circumvention. Although our experiments show 10.9% of the top 1 million websites are supporting ESNI, as of July 2019, less than 0.01% of TLS ClientHello messages are observed to contain an ESNI extension [8]. This strongly suggests the importance of having more clients and CDN providers supporting ESNI by default.

Be quick and quiet. Strategically, the benefits to censorship evasion brought by ESNI should not be emphasized before ESNI has a significant amount of daily usage. That is to say, if ESNI traffic gets censored in its early stage because censors find it is mostly used for circumvention, there will be less motivation for application developers and CDN providers to facilitate its deployment. On the other hand, the quicker ESNI gets widely used, the less time if left for censors.

5.2 Post-ESNI Era

More pressure on third parties. Suggested by Many previous experiences [25, 28], when ESNI is widely adopted, it is likely that censors will give more pressure to browser and CDN providers, forcing them to do self-censorship [26]. For instance, a large portion of Internet users in China are using re-branded browsers modified by local companies [9]. Those companies are often reported to conduct self-censorship in compliance with the law, making it not surprising if the ESNI feature is removed from those browsers. When it comes to CDN providers, previous experience [6] shows censors may bear with a high collateral damage and block a large number of IPs, forcing CDN providers to give up on ESNI. Our finding that a few CDN's IPs are already blocked also suggests this possibility.

Leakage in OCSP. For completeness, we note the certificate serial number in unencrypted OCSP (Online Certificate Status Protocol) messages can leak the server name [10]. It can be, consequently, exploited for censorship purposes.

6 Conclusion

In response to the increasing deployment of SNI filtering, ESNI is proposed to prevent censors from learning the server name. Through our work, we manage to understand the nature of SNI-based censorship by measuring its prevalence and effectiveness in China. We further explore its role in censorship by comparing it with other common censorship techniques. Our findings outline the overlapping relationships between different censorship methods, revealing the assisting role of SNI filtering in China's Internet censorship. Experimental result shows that 84.5% of the blocked websites are under IP blocking inside China, indicating a large portion of the sites will remain blocked even when SNI-based censorship is circumvented. During the probing test, we also find the duration of residual censorship by GFW has changed to 60 seconds.

Based on the understanding of SNI filtering, we did the first evaluation on the use of ESNI as a censorship circumvention strategy. From the experiments, we find around 10.9% of the Alexa top 1 million sites are already supporting ESNI. Furthermore, while using ESNI along with encrypted DNS channel helps to unblock only 66 currently censored sites, we argue the deployment of ESNI is still a progressive move as it essentially makes more than 101K websites more censorship-resistant. Contrary to the findings from previous works, we observe a few websites hosted on the CDN are indeed blocked by IP, suggesting the collateral damage of CDNs may be overestimated.

Since ESNI is still in its early stage, its fate deeply relies on whether any censorship action is taken against it. We, therefore, monitor censorship associated with ESNI from 14 areas all around the world. Contrary to a report claiming ESNI traffic is already blocked in South Korea, no censorship associated with ESNI is detected in any country we tested.

Finally, based on the findings from our experiments, we discuss the key to the success of ESNI as a censorship circumvention strategy. We conclude with an analysis on new challenges we may face when ESNI becomes an essential part of the Internet. We hope our work will make ESNI a more promising and effective censorship circumvention strategy.

We release all our probing tools and datasets to maintain reproducibility and to benefit future works on censorship measurement, obtainable at http://traces.cs.umass.edu/index.php/Network.

Acknowledgments

We would like to thank our shepherd Arturo Filastò for his thorough feedback and guidance. We are also deeply grateful to Philipp Winter, Nguyen Phong Hoang, Milad Nasr, Arian Akhavan Niaki and all the anonymous reviewers for offering us valuable assistance and constructive feedback on earlier versions of this paper. This work was supported by the NSF CAREER grant CNS-1553301. The opinions in this paper are those of the authors and do not necessarily reflect those of any funding agency or governmental organization.

References

[1] Report on the TLS fingerprint of meek with ESNI. https://www.bamsoftware.com/sec/meek-esni-tls-report. Accessed: May 2019.

[2] Domain fronting to App Engine stopped working. https://trac.torproject.org/projects/tor/ticket/25804, 2018. Accessed: May 2019.

[3] Encrypting SNI: Fixing One of the Core Internet Bugs. https://blog.cloudflare.com/esni, 2018. Accessed: April 2019.

[4] Enhanced Domain Protections for Amazon CloudFront Requests. https://aws.amazon.com/blogs/security/enhanced-domain-protections-for-amazon-cloudfront-requests, 2018. Accessed: May 2019.

[5] Google ends "domain fronting", a crucial way for tools to evade censors. https://www.accessnow.org/google-ends-domain-fronting-a-crucial-way-for-tools-to-evade-censors, 2018. Accessed: May 2019.

[6] Russia blocks millions of IP addresses in battle against Telegram app. https://www.theguardian.com/world/2018/apr/17/russia-blocks-millions-of-ip-addresses-in-battle-against-telegram-app, 2018. Accessed: May 2019.

[7] Support for Encrypted SNI (ESNI) for Chrome. https://bugs.chromium.org/p/chromium/issues/detail?id=908132, 2018. Accessed: May 2019.

[8] TLS Traffic with the ESNI Extension. https://web.archive.org/web/20190718002625/https://tlsfingerprint.io/find/extension/ffce, 2018. Accessed: July 2019.

[9] Browser Market Share in China. http://gs.statcounter.com/browser-market-share/all/china, 2019. Accessed: May 2019.

[10] Despite DoH and ESNI, with OCSP, web activity is insecure and not private. http://blog.seanmcelroy.com/2019/01/05/ocsp-web-activity-is-not-private, 2019. Accessed: May 2019.

[11] South Korea is Censoring the Internet by Snooping on SNI Traffic. https://web.archive.org/web/20190616020542/https://www.bleepingcomputer.com/news/security/south-korea-is-censoring-the-internet-by-snooping-on-sni-traffic, 2019. Accessed: July 2019.

[12] Donald E. Eastlake 3rd. Transport Layer Security (TLS) Extensions: Extension Definitions. RFC 6066, January 2011.

[13] Anonymous. Towards a comprehensive picture of the Great Firewall's DNS censorship. In *Free and Open Communications on the Internet*. USENIX, 2014.

[14] Cecylia Bocovich and Ian Goldberg. Slitheen: Perfectly imitated decoy routing through traffic replacement. In *Computer and Communications Security*. ACM, 2016.

[15] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. Technical report, Naval Research Lab Washington DC, 2004.

[16] Roya Ensafi, David Fifield, Philipp Winter, Nick Feamster, Nicholas Weaver, and Vern Paxson. Examining how the Great Firewall discovers hidden circumvention servers. In *Internet Measurement Conference*. ACM, 2015.

[17] Roya Ensafi, Philipp Winter, Abdullah Mueen, and Jedidiah R. Crandall. Analyzing the Great Firewall of China Over Space and Time. *Privacy Enhancing Technologies*, 2015(1), 2015.

[18] David Fifield. *Threat modeling and circumvention of Internet censorship*. PhD thesis, EECS Department, University of California, Berkeley, Dec 2017.

[19] David Fifield. Anticipating a world of encrypted SNI: risks, opportunities, how to win big. https://www.bamsoftware.com/sec/esni.html, 2018. Accessed: February 2019.

[20] David Fifield, Chang Lan, Rod Hynes, Percy Wegmann, and Vern Paxson. Blocking-resistant communication through domain fronting. *Privacy Enhancing Technologies*, 2015(2), 2015.

[21] Sergey Frolov, Fred Douglas, Will Scott, Allison McDonald, Benjamin VanderSloot, Rod Hynes, Adam Kruger, Michalis Kallitsis, David G. Robinson, Steve Schultze, Nikita Borisov, J. Alex Halderman, and Eric Wustrow. An ISP-scale deployment of TapDance. In *Free and Open Communications on the Internet*. USENIX, 2017.

[22] Sergey Frolov and Eric Wustrow. The use of TLS in Censorship Circumvention. In *Network and Distributed System Security*. The Internet Society, 2019.

[23] John Holowczak and Amir Houmansadr. CacheBrowser: Bypassing Chinese censorship without proxies using cached content. In *Computer and Communications Security*. ACM, 2015.

[24] Gordon Fyodor Lyon. *Nmap network scanning: The official Nmap project guide to network discovery and security scanning*. Insecure, 2009.

[25] Bill Marczak, Nicholas Weaver, Jakub Dalek, Roya Ensafi, David Fifield, Sarah McKune, Arn Rey, John Scott-Railton, Ron Deibert, and Vern Paxson. An analysis of China's "Great Cannon". In *Free and Open Communications on the Internet*. USENIX, 2015.

[26] Allison McDonald, Matthew Bernhard, Luke Valenta, Benjamin VanderSloot, Will Scott, Nick Sullivan, J. Alex Halderman, and Roya Ensafi. 403 Forbidden: A Global View of CDN Geoblocking. In *Proceedings of the Internet Measurement Conference 2018*, IMC '18, pages 218–230, New York, NY, USA, 2018. ACM.

[27] Milad Nasr, Hadi Zolfaghari, and Amir Houmansadr. The waterfall of liberty: Decoy routing circumvention that resists routing attacks. In *Computer and Communications Security*. ACM, 2017.

[28] Xiao Qiang. The road to digital unfreedom: President xi's surveillance state. *Journal of Democracy*, 30(1):53–67, 2019.

[29] Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, August 2018.

[30] Eric Rescorla, Kazuho Oku, Nick Sullivan, and Christopher A. Wood. Encrypted Server Name Indication for TLS 1.3. Internet-Draft draft-ietf-tls-esni-03, Internet Engineering Task Force, March 2019. Work in Progress.

[31] Quirin Scheitle, Oliver Hohlfeld, Julien Gamba, Jonas Jelten, Torsten Zimmermann, Stephen D Strowes, and Narseo Vallina-Rodriguez. A long way to the top: significance, structure, and stability of internet top lists. In *Proceedings of the Internet Measurement Conference 2018*, pages 478–493. ACM, 2018.

[32] Craig A Shue, Andrew J Kalafut, and Minaxi Gupta. The Web is Smaller than it Seems. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 123–128. ACM, 2007.

[33] Sparks, Neo, Tank, Smith, and Dozer. The collateral damage of Internet censorship by DNS injection. *SIGCOMM Computer Communication Review*, 42(3):21–27, 2012.

[34] Benjamin VanderSloot, Allison McDonald, Will Scott, J. Alex Halderman, and Roya Ensafi. Quack: Scalable remote measurement of application-layer censorship. In *USENIX Security Symposium*. USENIX, 2018.

[35] Zhongjie Wang, Yue Cao, Zhiyun Qian, Chengyu Song, and Srikanth V. Krishnamurthy. Your state is not mine: A closer look at evading stateful Internet censorship. In *Internet Measurement Conference*. ACM, 2017.

[36] Zachary Weinberg, Shinyoung Cho, Nicolas Christin, Vyas Sekar, and Phillipa Gill. How to Catch when Proxies Lie: Verifying the Physical Locations of Network Proxies with Active Geolocation. In *Proceedings of the Internet Measurement Conference 2018*, pages 203–217. ACM, 2018.

[37] Eric Wustrow, Scott Wolchok, Ian Goldberg, and J. Alex Halderman. Telex: Anticensorship in the network infrastructure. In *USENIX Security Symposium*. USENIX, 2011.

[38] Hadi Zolfaghari and Amir Houmansadr. Practical censorship evasion leveraging content delivery networks. In *Computer and Communications Security*. ACM, 2016.