I(ra)nconsistencies: Novel Insights into Iran's Censorship

Felix Lange
Paderborn University

Niklas Niere
Paderborn University

Jonathan von Niessen
Paderborn University

Dennis Suermann
Paderborn University

Nico Heitmann
Paderborn University

Juraj Somorovsky
Paderborn University

ABSTRACT

Iran employs one of the most prominent Internet censors in the world. An important part of Iran's censorship apparatus is its analysis of unencrypted protocols such as HTTP and DNS. During routine evaluations of Iran's HTTP and DNS censorship, we noticed several properties we believe to be unknown today. For instance, we found injections of correct static IPs for some domains such as google.com on the DNS level, unclear HTTP version parsing, and correlations between DNS and HTTP censorship. In this paper, we present our findings to the community and discuss possible takeaways for affected people and the censorship circumvention community. As some of our findings left us bewildered, we hope to ignite a discussion about Iran's censorship behavior. We aim to use the discussion of our work to execute a thorough analysis and explanation of Iran's censorship behavior in the future.

KEYWORDS

Censorship, Iran, HTTP, DNS

1 INTRODUCTION

Internet censorship is widely applied by various countries [27]. Countries such as Russia [32, 37, 38], China [2, 7–9, 14, 19, 33], and Iran [1, 3, 6] restrict their residents' Internet access by analyzing and manipulating Internet traffic across different protocols. Using so-called deep packet inspection (DPI), censors analyze protocols such as TLS [7, 37], VPN protocols [13], HTTP [18], IP [27], and DNS [29]. To intercept connections they deem harmful, censors employ a plethora of techniques: they drop packets [37], throttle Internet speed [38], and inject forged packets such as DNS responses [19], HTTP block pages [3], and TCP RSTs [7]. As affected people continuously circumvent censors, censors continuously alter and improve their techniques.

Censorship of Unencrypted Protocols. Due to their age and prevalence, HTTP and DNS are two protocols with a long history of censorship [3, 18]. While encrypted alternatives such as HTTP over TLS (HTTPS) [11], DNS over TLS [21], and DNS over HTTPS [20] rise in importance, both HTTP and DNS are still widely used in countries affected by censorship. For instance, around 1% of HTTP requests from Iran to Cloudflare are unencrypted—13% when including bot traffic [10]. Second, some censored websites are only accessible over unencrypted HTTP [23]. Third, encrypted DNS services are often unreachable in countries affected by censorship [4, 12, 25], forcing users to rely on unencrypted DNS. Lastly, censors are incentivized to block unencrypted protocols as long as affected people use them to access forbidden resources.

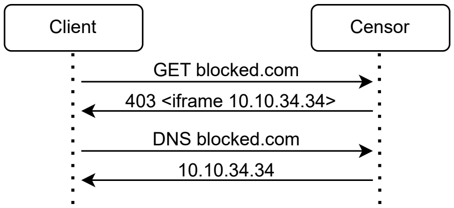


Figure 1: HTTP and DNS censorship in Iran as described by previous work [3].

HTTP and DNS Censorship in Iran. Due to their continued importance, HTTP and DNS are censored in Iran. To censor unwanted HTTP requests, Iran has been found to inject a block page based on censored strings in the path and Host header field that informs the user about accessing a forbidden resource [3]. Iran also injects false DNS responses to queries about unwanted domains [3]. We showcase both techniques in Figure 1. Notably, both DNS and HTTP censorship utilize the same block page. In this work, we introduce further details about Iran's HTTP and DNS censors and find possible correlations between the two.

Contributions. In this work, we uncovered several previously unexamined techniques inside Iran's censorship structure:

- Iran's HTTP censorship behavior—injecting block pages and TCP RSTs, and dropping packets—depends on the censored website.
- Iran's HTTP censorship fails for different HTTP headers and HTTP versions.
- Iran's HTTP censorship is not limited to port 80.
- Iran's DNS censor injects correct IP addresses for some domains and overblocks domains with the substring `wpad`.
- Iran's DNS and HTTP censorship show correlations regarding their censorship techniques for specific domains.

With these findings, we hope to motivate a discussion about Iran's censorship structure. We plan to analyze Iran's censorship structure systematically in the future.

2 METHODOLOGY

Our goal was to obtain an updated view of the current state of HTTP and DNS censorship in Iran. We analyzed the censorship of both protocols in Iran with several scans. For all scans, we rented a vantage point in Iran and a reference server in Germany. Specifications of the rented servers can be found in Appendix A. In the following, we outline the concrete methodology of our evaluations.

Free and Open Communications on the Internet 2025(1), 7–12

2.1 DNS Evaluation

For our DNS evaluations, we configured the server in Germany as a DNS resolver that answers all DNS requests with a static DNS response. To detect whether a DNS request was censored, we sent it from our vantage point in Iran to port 53 on our DNS resolver in Germany and measured differences to the expected static answer by our DNS resolver. This procedure is similar to the techniques described by Jin et al. [22]. On our vantage points, we analyzed all domains from the Tranco top one million list with subdomains [24][1] and the Citizen Lab global test list[2] once. During the scan, we recorded all received responses for further analysis.

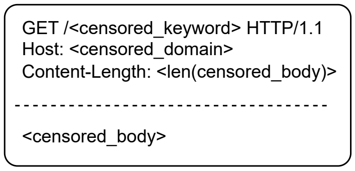
2.2 HTTP Evaluation

Similar to our DNS evaluation and the techniques defined by Jin et al. [22], we analyzed HTTP censorship in Iran by sending crafted HTTP messages from our vantage point in Iran to varying ports of our server in Germany—as we could reproduce identical censorship across different ports, we did not have to constrain our scan to port 80. Our German server, configured as a simple TCP server, always responded with predefined bytes. We detected censorship by comparing received responses against these static bytes. We accommodated Iran's residual censorship by alternating between different client ports: Iran's residual censorship only triggers on the 4-tuple (client IP, client port, server IP, server port) [5]. This allowed us to avoid waiting for the otherwise necessary time of residual censorship. We sent each crafted HTTP message 20 times, recorded all observed packets for further analysis, and determined censorship behavior by majority voting.

Analyzed Domains and Keywords. We analyzed Iran's HTTP censorship of forbidden hostnames and keywords. We considered domains from the Tranco top 100 [24][3] and the Citizen Lab test list for Iran[4], leading to 976 distinct domains. For analyzing keyword-based censorship, we used keywords from an OpenNet Initiative report[5] and from Nazeri and Anderson [28] as a starting point. To increase the number of keywords we could analyze, we utilized the large language model GPT-4o[6] to generate two lists, each containing 100 sexual terms. One list comprises English words such as *porn*, *nude*, and *gay*, while the other list includes Persian equivalents. To have a larger sample size for our later correlation study between HTTP and DNS censorship, we used the Tranco top 10,000 [24][3].

Censorship Locations. We placed potentially censored keywords and hostnames in three locations of an HTTP request: the path, the Host header, or the message body. An example of an HTTP request with censored strings in these three locations can be seen in Figure 2. Additionally, we altered the structure of our HTTP requests by using different HTTP methods and HTTP versions, and including or omitting the HTTP body and Host header.

HTTP Methods. We evaluated Host header- and path-based HTTP censorship for all standardized HTTP methods and additional special strings. For example, we explicitly chose the lower-case gET method as case-sensitivity was successfully employed in the past for other countries [18]. A complete list can be found in Appendix B. We evaluated each considered HTTP method by combining it with 10 random censored domains and keywords.



```
GET /<censored_keyword> HTTP/1.1
Host: <censored_domain>
Content-Length: <len(censored_body)>
------------------------------------
<censored_body>
```

Figure 2: Example HTTP/1.1 request with censored strings in the path, Host header, and the message body.

HTTP Versions. Next to changing the HTTP method, we also manipulated the HTTP version of our HTTP methods in requests targeted by Host header- and path-based HTTP censorship. As a starting point, we chose standard-compliant values. Additionally, we chose uncommon and non-standard values to analyze the HTTP censorship filter for different versions. A complete list can be found in Appendix B. For each version string, we sent censored requests with and without a Host header and either a censored hostname in the Host header or a censored keyword in the request path.

3 FINDINGS

Using the above methodology, we analyzed the Iranian censor's DNS and HTTP censorship mechanisms. We performed our scans of Iran's DNS and HTTP censorship in April 2024 and August 2024, respectively. In this section, we detail our findings for both protocols separately and then correlate them.

3.1 DNS Censorship

At our vantage point, the Iranian censor intercepted to-be-censored DNS requests without forwarding them to the DNS resolver and injected its own DNS response (cf. Figure 3). This injected DNS response then contains an IP that leads to an HTTP block page. This process of inline censorship aligns with previous research [3].

Injected Block Page IPs. While analyzing the DNS responses from our scan, we identified that all responses contain exactly one record in the answers section. The IP address 10.10.34.36 was injected for 41,285 (≈87%) domains and 10.10.34.34 for 6,348 (≈13%). A complete mapping is available via GitHub[7]. Notably, this is a difference compared to observations made by Aryan et al. [3] as they did not report any occurrences of the IP 10.10.34.36, which is now used for a majority of injections. Initially, we suspected ISP-specific censorship was the cause of different IP addresses being injected, but further analyses with increasing Time To Live (TTL) values—a similar approach as CenTrace created by Raman et al. [31]—revealed that the same network node injected both IP addresses. We discuss this further in Section 3.3.

Injected Valid IPs. Next to block page IPs, Iran's censor injects a DNS response with a static but correct IP for some hostnames. Figure 3 depicts this behavior in contrast to Iran's default DNS block page injection. Iran's censor intercepts DNS requests with 372 Google-related domains, such as google.com and google.ca, and injects a DNS record with the IP 216.239.38.120—an IP that corresponds to Google. Next to hostnames owned by Google, Iran's

[1] https://tranco-list.eu/list/244Y9/1000000, Accessed 3.4.2024
[2] https://github.com/citizenlab/test-lists/commit/c1cdb35, Accessed 28.03.2024
[3] https://tranco-list.eu/list/KJWPW Accessed 19.5.2024
[4] https://github.com/citizenlab/test-lists/commit/e9e3bfb Accessed 10.04.2024
[5] https://www.opennet.net/iranreport2013, Accessed 4.8.2024
[6] https://openai.com/index/hello-gpt-4o/, Accessed 4.8.2024

[7] https://github.com/UPB-SysSec/IranconsistenciesData

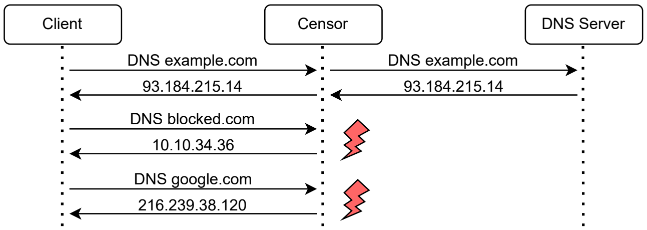


Figure 3: Injections by Iran's DNS censor. Note that for google.com, a valid IP is explicitly injected by the censor. This way, all DNS requests for google.com resolve to a single IP address.

censor exhibits this behavior for 13 other domains, including search engines, foreign secret service websites, and a video game provider. We provide a complete list of injected correct IPs in Appendix C. The correct IP addresses are injected by the same network node as the block page IPs (cf. Section 3.3). As the injected IP address remains the same for all DNS queries, all traffic affected by the censor connects to the same IP address. We provide two theories on why Iran's censor directs traffic for affected domains to a static IP address. Firstly, access to the affected domain could potentially be tracked by filtering traffic to the static IP address and would be resilient to changing DNS records. Secondly, resolving domains to a static IP address might allow Iran's censor to swiftly censor domains by blocking access to the injected static IP address instead of relying on advanced DPI or tracking multiple—potentially cached—IP addresses. While these could explain Iran's injection of correct IPs, the exact reasoning remains unclear, making it an interesting avenue for future research. This injection can be circumvented by using encrypted DNS or resolving IP addresses over a different source than DNS.

WPAD Censorship. An unexpected finding was the overblocking of the term *wpad*. All requests for domains that include this term were censored by null-routing the initial DNS request. We provide a list of all censored domains containing the string `wpad` in Appendix D. Notably, this overblocking seems only temporary as it is not reproducible anymore. We suspect this overblocking happened due to the Web Proxy Auto-Discovery Protocol (WPAD) [30]. During the discovery protocol, several DNS requests are made for domains that include *wpad* as a prepended subdomain. Before adjusting the censorship mechanism, it seems like the Iranian censor blocked all DNS requests that contained this term at any position.

3.2 HTTP Censorship

In the following, we detail our findings about Iran's HTTP censorship which we measured from our vantage point.

Blocking Techniques. We evaluated Iran's HTTP censorship by domain. For most of the 322 websites, we observed a block page. Observed block pages always contained an iframe with the source pointing to the IP 10.10.34.36—one of the injected IPs by Iran's DNS censor. For 13 websites, we received no response; for five websites, we received a TCP RST packet. A complete list of domain names for each censorship method is available via GitHub[8]. Following our approach for DNS (cf. Section 3.1), we located the censorship device by increasing TTL values of censored packets. All domains from the three categories were censored on the same hop, suggesting that the same censorship system applies different censorship mechanisms (cf. Section 3.3). These findings show that the Iranian censor's HTTP censorship is more complex than outlined in related work. Unlike previous work, Iran's HTTP censorship was not limited to port 80 [6]. We consider this an indication of Iran's continuous development of its censor and hope to validate this behavior from other vantage points.

Censored Methods. All defined HTTP methods (e.g., GET, CONNECT) are analyzed and censored by the Iranian censor. Notably, the lower-case variant gET is not being analyzed and censored, meaning that the censor is case-sensitive for HTTP methods. The Iranian censor also ignored all non-standard-compliant values that we tried, such as an empty string, a space, or a censored keyword. We consider this an interesting oversight by the Iranian censor as Harrity et al. have previously shown the viability of such techniques—especially lower-case changes—in India [18].

Analyzed Parts of the HTTP Message. The Iranian censor analyzes the Host header and path of HTTP requests but ignores the body. Furthermore, censored domains are only censored in the Host header and censored keywords only in the path. We suspect this is due to performance optimizations, as it is more likely for a censored domain to appear in the Host header and a censored keyword to appear in the path value. This decision by Iran's censor leads to interesting behavior in combination with different HTTP versions.

Censored HTTP Versions. We observed different censorship behavior for different version values, dependent on the presence of the Host header, as depicted in Table 1. When a Host header is present, the path value is always analyzed for censored keywords, independent of the tested HTTP version. The Host header is not censored for the version strings `HTTP`, `1.1`, and `example`. This suggests that the censor uses a version regex of `HTTP/.*` for Host header censorship. The most unexpected behavior is present when the Host header is omitted. In this case, the path is not censored for the version strings `HTTP` and `HTTP/1`. However, censorship of the path is still applied for cases like `1.1` and `example`. We suspect this behavior to be a byproduct of performance optimizations of Iran's HTTP parser, but we are ultimately uncertain. This behavior leaves room for potential circumvention techniques and stresses the importance of automated tools such as Geneva [7] that combine different manipulations.

Table 1: Censorship of different HTTP version values in Iran. Censorship depends on the request's structure and the location of the censored keyword. ●=censored, –=not censored.

Value	Path only	Path and Host header	
		Censored string in path	Censored string in Host
HTTP	–	●	–
HTTP/0	●	●	●
HTTP/1	–	●	●
HTTP/1.2	●	●	●
HTTP/1.9	●	●	●
HTTP/1.10	●	●	●
HTTP/1.a	●	●	●
HTTP/10	●	●	●
HTTP/3	●	●	●
1.1	●	●	–
example	●	●	–

[8] https://github.com/UPB-SysSec/IranconsistenciesData

3.3 Correlation of HTTP and DNS Censorship

After analyzing Iran's HTTP and DNS censorship, we evaluated correlations between injected IPs and HTTP censorship methods. As our initial DNS scan was conducted four months before the HTTP scan, we re-evaluated DNS censorship for the Tranco top 10,000 domains in August 2024. Table 2 shows a strong correlation between TCP RSTs being used as an HTTP censorship method and the IP 10.10.34.34 being injected on the DNS level. Similarly, most domains that are censored with the IP 10.10.34.36 on the DNS level are censored with a block page. While we detect a strong correlation between DNS and HTTP censorship, it is incomplete: some domains are censored with a TCP RST via HTTP and `10.10.34.36` via DNS. We are unsure why this is the case and plan to execute extended tests with more domains to find an explanation for this correlation.

Censoring Network Node. All HTTP and DNS censorship we encountered in Iran occurred at the same network node. Altering the TTL value of our packets, we deduced that all censorship happened at the last network node in Iran. This hints towards a centralized censorship architecture using border nodes rather than ISP-level censorship. On the other hand, we could not reproduce all the censorship detected recently by Bock et al. [5] and residents in the country [16, 34, 36] in our evaluation. For instance, we did not encounter block pages that contain 10.10.34.35 or 10.10.34.36 and did not encounter DNS injections of 10.10.34.35. Opposing previous research [5], the observed HTTP censorship was also not limited to port 80. This suggests that some parts of Iranian censorship only happen from certain vantage points, and while censorship happens at border nodes, different border nodes might employ different censorship techniques. We are interested in executing our analyses on further vantage points to clarify Iran's censorship architecture.

Table 2: Correlation between HTTP censorship method and DNS censorship IP in Iran for the domains of the Tranco top 10,000 affected by HTTP censorship. It shows a strong correlation between DNS and HTTP censorship methods.

DNS\HTTP	Block Page	No response	TCP RST
10.10.34.34	46 (3.2%)	5 (9.1%)	**33 (86.8%)**
10.10.34.36	**1,233 (84.6%)**	**42 (76.4%)**	3 (7.9%)
Correct IP	3 (0.2%)	0 (0.0%)	0 (0.0%)
Not Censored	175 (12.0%)	8 (14.6%)	2 (5.3%)
Total	1,457 (100%)	55 (100%)	38 (100%)

4 RELATED WORK

HTTP Censorship in Iran. In 2013, Aryan et al. [3] presented the first analysis of Internet censorship in Iran. They discovered HTTP censorship based on the Host header and path-based keyword censorship. According to Aryan et al., the censorship node intercepts HTTP requests without forwarding them to the server, instead responding with a 403 (Forbidden) status code. The response contains an iframe with a block page on 10.10.34.34. This behavior has also been observed by Gill [17] and Bock et al. [5]. Responses with an iframe with the IPs 10.10.34.34, 10.10.34.35, and 10.10.34.36 have also been observed in the past [16, 34, 35]; we only observed the IP 10.10.34.34.

Bock et al. [5] have also described Iran's HTTP censor to inject TCP RST packets and null-route connections while at the same time serving a block page. We detected that the censorship behavior of Iran's HTTP censor changes with the censored hostname. Furthermore, we noticed parsing differences in Iran's HTTP censor for different HTTP methods and versions. While Harrity et al. [18] evaluated them in China, India, and Kazakhstan, we are unaware of a similar work that performed such an analysis for Iran.

DNS Censorship in Iran. Next to Iran's HTTP censorship, Aryan et al. [3] detected Iranian DNS censorship through DNS record interception and injection. Iran's DNS censor intercepts DNS requests without forwarding them to the DNS server and responds with its own answer record pointing to the IP 10.10.34.34. The injected IP is only accessible from Iran's national network and matches the IP address used in the block page of Iran's HTTP censorship. While the authors only describe injection of the IP 10.10.34.34, we could detect injections of 10.10.34.34 and 10.10.34.36 depending on the censored hostname. Users have reported the injection of three IP addresses 10.10.34.34, 10.10.34.35, and 10.10.34.36 [26, 35, 36]. We could not trigger the injection of 10.10.34.35 for any domain during our scans. Instead, we detected that Iran's DNS censor injects correct IP addresses; we are unaware of previous research that detected this. Similarly, we could not find reports of Iran's overblocking of all domains that contain the string `wpad`.

5 ETHICAL CONSIDERATIONS

We designed our methodology to have a minimal impact on real websites, servers, and people living in Iran. We conducted all scans from a rented vantage point in Iran to a rented server in Germany. In light of this, we did not impose any traffic on real websites or servers, and we did not involve any human subjects in our experiments. While renting our vantage points, we verified that neither the company nor any other person involved in the payment process is listed on the applying sanctions list [15]. We adhere to current export regulations by the European Union as the censorship analysis software we ran on our vantage point does not aid Iranian oppression. We conferred with our university's Export Control Officer who affirmed our procedure. While the Iranian censor might adjust their censorship system based on our findings, we strongly believe that our efforts to understand Iranian censorship techniques provide greater benefit to affected people than to the Iranian censor.

6 CONCLUSIONS

In this paper, we evaluated the current state of Iran's DNS and HTTP censorship. We discovered previously undetected techniques, such as Iran's DNS censor injecting correct IP addresses for some domains. Some discoveries have left us wondering about their reason and potential purpose. Why does Iran's DNS censor inject correct IP addresses for some domains? How can we utilize irregularities in Iran's HTTP parser to find new circumvention techniques? Where does Iranian censorship occur? All censorship we encountered occurred at border nodes, suggesting a centralized censorship architecture. To answer these questions, we aim to evaluate Iranian censorship from multiple vantage points served by different ISPs. In our evaluations, we will include additional protocols such as IP, TLS, and encrypted DNS to achieve a holistic picture of the Iranian censorship system. Before conducting our final evaluations, we provide insights into Iran's censorship for affected people with this work. We hope to gather feedback that will aid our final evaluations.

ACKNOWLEDGMENTS

Niklas Niere and Felix Lange were supported by the German Federal Ministry of Education and Research (BMBF) through the project KoTeBi (16KIS1556K). Nico Heitmann was supported by the research project "North-Rhine Westphalian Experts in Research on Digitalization (NERD II)", sponsored by the state of North Rhine-Westphalia – NERD II 005-2201-0014.

REFERENCES

[1] Collin Anderson. 2013. Dimming the Internet: Detecting Throttling as a Mechanism of Censorship in Iran. https://doi.org/10.48550/arXiv.1306.4361 arXiv:1306.4361 [cs].

[2] Anonymous. 2014. Towards a Comprehensive Picture of the Great Firewall's DNS Censorship. In *4th USENIX Workshop on Free and Open Communications on the Internet (FOCI 14)*. USENIX Association, San Diego, CA. https://www.usenix.org/conference/foci14/workshop-program/presentation/anonymous

[3] Simurgh Aryan, Homa Aryan, and J. Alex Halderman. 2013. Internet Censorship in Iran: A First Look. In *3rd USENIX Workshop on Free and Open Communications on the Internet (FOCI 13)*. USENIX Association, Washington, D.C. https://www.usenix.org/conference/foci13/workshop-program/presentation/aryan

[4] Simone Basso. 2022. Measuring DoT/DoH Blocking Using OONI Probe: A Preliminary Study. https://ooni.org/post/2022-doh-dot-paper-dnsprivacy21/

[5] Kevin Bock, Pranav Bharadwaj, Jasraj Singh, and Dave Levin. 2021. Your Censor is My Censor: Weaponizing Censorship Infrastructure for Availability Attacks. In *2021 IEEE Security and Privacy Workshops (SPW)*. 398–409. https://doi.org/10.1109/SPW53761.2021.00059

[6] Kevin Bock, Yair Fax, Kyle Reese, Jasraj Singh, and Dave Levin. 2020. Detecting and Evading Censorship-in-Depth: A Case Study of Iran's Protocol Whitelister. In *10th USENIX Workshop on Free and Open Communications on the Internet (FOCI 20)*. USENIX Association. https://www.usenix.org/conference/foci20/presentation/bock

[7] Kevin Bock, George Hughey, Xiao Qiang, and Dave Levin. 2019. Geneva: Evolving Censorship Evasion Strategies. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS '19)*. Association for Computing Machinery, New York, NY, USA, 2199–2214. https://doi.org/10.1145/3319535.3363189

[8] Kevin Bock, iyouport, Anonymous, Louis-Henri Merino, David Fifield, Amir Houmansadr, and Dave Levin. 2020. Exposing and Circumventing China's Censorship of ESNI. https://gfw.report/blog/gfw_esni_blocking/en/

[9] Richard Clayton, Steven J. Murdoch, and Robert N. M. Watson. 2006. Ignoring the Great Firewall of China. In *Privacy Enhancing Technologies*, George Danezis and Philippe Golle (Eds.). Springer, Berlin, Heidelberg, 20–35. https://doi.org/10.1007/11957454_2

[10] Cloudflare. 2024. *Adoption & Usage in Iran | Cloudflare Radar*. https://radar.cloudflare.com/adoption-and-usage/ir

[11] T. Dierks and E. Rescorla. 2008. *The Transport Layer Security (TLS) Protocol Version 1.2*. RFC 5246. IETF. http://tools.ietf.org/rfc/rfc5246.txt

[12] Kathrin Elmenhorst, Bertram Schütz, Nils Aschenbruck, and Simone Basso. 2021. Web censorship measurements of HTTP/3 over QUIC. In *Proceedings of the 21st ACM Internet Measurement Conference (IMC '21)*. Association for Computing Machinery, New York, NY, USA, 276–282. https://doi.org/10.1145/3487552.3487836

[13] Roya Ensafi, David Fifield, Philipp Winter, Nick Feamster, Nicholas Weaver, and Vern Paxson. 2015. Examining How the Great Firewall Discovers Hidden Circumvention Servers. In *Proceedings of the 2015 Internet Measurement Conference*. ACM, Tokyo Japan, 445–458. https://doi.org/10.1145/2815675.2815690

[14] Roya Ensafi, Philipp Winter, Abdullah Mueen, and Jedidiah R. Crandall. 2015. Analyzing the Great Firewall of China Over Space and Time. *Proceedings on Privacy Enhancing Technologies* (2015). https://petsymposium.org/popets/2015/popets-2015-0005.php

[15] European Union. 2025. *EU Sanctions Map*. https://www.sanctionsmap.eu/

[16] ftfws. 2023. *Internet Censorship in Iran: A First Look (FOCI 2013) · Issue #226 · net4people/bbs*. https://github.com/net4people/bbs/issues/226#issuecomment-1485926584

[17] Phillipa Gill. 2016. A Case Study in Iran. https://iclab.gitlab.io/post/iran_case_study_2016/

[18] Michael Harrity, Kevin Bock, Frederick Sell, and Dave Levin. 2022. GET /out: Automated Discovery of Application-Layer Censorship Evasion Strategies. In *31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, Boston, MA, 465–483. https://www.usenix.org/conference/usenixsecurity22/presentation/harrity

[19] Nguyen Phong Hoang, Arian Akhavan Niaki, Jakub Dalek, Jeffrey Knockel, Pellaeon Lin, Bill Marczak, Masashi Crete-Nishihata, Phillipa Gill, and Michalis Polychronakis. 2021. How Great is the Great Firewall? Measuring China's DNS Censorship. In *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, 3381–3398. https://www.usenix.org/conference/usenixsecurity21/presentation/hoang

[20] P. Hoffman and P. McManus. 2018. *DNS Queries over HTTPS (DoH)*. RFC 8484. IETF. http://tools.ietf.org/rfc/rfc8484.txt

[21] Z. Hu, L. Zhu, J. Heidemann, A. Mankin, D. Wessels, and P. Hoffman. 2016. *Specification for DNS over Transport Layer Security (TLS)*. RFC 7858. IETF. http://tools.ietf.org/rfc/rfc7858.txt

[22] Lin Jin, Shuai Hao, Haining Wang, and Chase Cotton. 2021. Understanding the Practices of Global Censorship through Accurate, End-to-End Measurements. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 5, 3 (Dec. 2021), 43:1–43:25. https://doi.org/10.1145/3491055

[23] Citizen Lab and Others. 2014. URL testing lists intended for discovering website censorship. https://github.com/citizenlab/test-lists

[24] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczynski, and Wouter Joosen. 2019. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *Proceedings 2019 Network and Distributed System Security Symposium*. Internet Society, San Diego, CA. https://doi.org/10.14722/ndss.2019.23386

[25] Ruixuan Li, Baojun Liu, Chaoyi Lu, Haixin Duan, and Jun Shao. 2024. A Worldwide View on the Reachability of Encrypted DNS Services. In *Proceedings of the ACM Web Conference 2024*. ACM, Singapore Singapore, 1193–1202. https://doi.org/10.1145/3589334.3645539

[26] markpash. 2023. *Cloudflare workers has blocked in iran · Issue #215 · net4people/bbs*. https://github.com/net4people/bbs/issues/215#issuecomment-1445436281

[27] Alexander Master and Christina Garman. 2023. A Worldwide View of Nation-state Internet Censorship. *Free and Open Communications on the Internet* (2023). https://petsymposium.org/foci/2023/foci-2023-0008.php

[28] Nima Nazeri and Collin Anderson. 2013. Citation Filtered: Iran's Censorship of Wikipedia. https://www.semanticscholar.org/paper/Citation-Filtered%3A-Iran%E2%80%99s-Censorship-of-Wikipedia-Nazeri-Anderson/15f13eb5c7fa7128cd6551d0fe1c285a7763c0ea

[29] Paul Pearce, Ben Jones, Frank Li, Roya Ensafi, Nick Feamster, Nick Weaver, and Vern Paxson. 2017. Global Measurement of DNS Manipulation. 307–323. https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/pearce

[30] Charles E. Perkins, Josh Cohen, Mark Dunsmuir, Paul A. Gauthier, Ian Cooper, and John W. Cohen M.A. 1999. *Web Proxy Auto-Discovery Protocol*. Internet Draft draft-ietf-wrec-wpad-01. Internet Engineering Task Force. https://datatracker.ietf.org/doc/draft-ietf-wrec-wpad-01 Num Pages: 18.

[31] Ram Sundara Raman, Mona Wang, Jakub Dalek, Jonathan Mayer, and Roya Ensafi. 2022. Network measurement methods for locating and examining censorship devices. In *Proceedings of the 18th International Conference on emerging Networking EXperiments and Technologies (CoNEXT '22)*. Association for Computing Machinery, New York, NY, USA, 18–34. https://doi.org/10.1145/3555050.3569133

[32] Reethika Ramesh, Ram Sundara Raman, Matthew Bernhard, Victor Ongkowijaya, Leonid Evdokimov, Anne Edmundson, Steven Sprecher, Muhammad Ikram, and Roya Ensafi. 2020. Decentralized Control: A Case Study of Russia. In *Proceedings 2020 Network and Distributed System Security Symposium*. Internet Society, San Diego, CA. https://doi.org/10.14722/ndss.2020.23098

[33] Philipp Winter and Stefan Lindskog. 2012. How the Great Firewall of China is Blocking Tor. In *2nd USENIX Workshop on Free and Open Communications on the Internet (FOCI 12)*. USENIX Association, Bellevue, WA. https://www.usenix.org/conference/foci12/workshop-program/presentation/Winter

[34] wkrp. 2023. *Cloudflare workers has blocked in iran · Issue #215 · net4people/bbs*. https://github.com/net4people/bbs/issues/215#issuecomment-1446861291

[35] wkrp. 2023. *Internet Censorship in Iran: A First Look (FOCI 2013) · Issue #226 · net4people/bbs*. https://github.com/net4people/bbs/issues/226#issuecomment-1483869382

[36] wkrp. 2023. *Snowflake Domain Front Blocked in Some ISPs in Iran; Suggested Workarounds · Issue #197 · net4people/bbs*. https://github.com/net4people/bbs/issues/197

[37] Diwen Xue, Benjamin Mixon-Baca, ValdikSS, Anna Ablove, Beau Kujath, Jedidiah R. Crandall, and Roya Ensafi. 2022. TSPU: Russia's decentralized censorship system. In *Proceedings of the 22nd ACM Internet Measurement Conference (IMC '22)*. Association for Computing Machinery, New York, NY, USA, 179–194. https://doi.org/10.1145/3517745.3561461

[38] Diwen Xue, Reethika Ramesh, Valdik S S, Leonid Evdokimov, Andrey Viktorov, Arham Jain, Eric Wustrow, Simone Basso, and Roya Ensafi. 2021. Throttling Twitter: an emerging censorship technique in Russia. In *Proceedings of the 21st ACM Internet Measurement Conference (IMC '21)*. Association for Computing Machinery, New York, NY, USA, 435–443. https://doi.org/10.1145/3487552.3487858

A SERVER SPECIFICATIONS

Table 3: Specifications of the server in Iran.

Country:	Mashhad, Iran
Autonomous System Number:	201295
Vendor:	Avanetco
URL:	https://www.avanetco.com/
Internet Service Provider:	Shabakeh Ertebatat Artak Towseeh PJSC (private)

Table 4: Specifications of the server in Germany.

Country:	Berlin, Germany
Autonomous System Number:	201295
Vendor:	IONOS
URL:	https://www.ionos.de/
Internet Service Provider:	IONOS SE (private)

B HTTP METHODS AND VERSION STRINGS

Table 5: HTTP methods and versions we included in our evaluations of Iran's HTTP censor.

HTTP Methods	`GET`, `HEAD`, `POST`, `PUT`, `DELETE`, `CONNECT`, `OPTIONS`, `TRACE`¸ `PATCH`, `gET`
	`""` (empty string), `" "` (single space), `null`

HTTP Versions	`HTTP`, `HTTP/0`, `HTTP/1`, `HTTP/1.2`, `HTTP/1.9`, `HTTP/1.10`, `HTTP/1.a`, `HTTP/10`, `HTTP/3`, `1.1`, `example`

C COMPLETE LIST OF IRAN CORRECT IP INJECTIONS

Table 6: Listing of all domains that return a fixed IP which is related to the correct domain. A complete list of domains for each group is available via GitHub[9].

Domain Group	Count	IP
Google	372	216.239.38.120
Bing	2	204.79.197.220
DuckDuckGo	2	52.250.41.2
Yandex	1	213.180.193.56
CIA	1	93.115.151.123
MI5	3	185.130.45.94
Mossad	1	87.107.132.83
GJacky	1	10.202.7.212
IGameCJ	1	162.62.115.144
Public IGameCJ	1	162.62.116.251

D DOMAINS LEADING TO NULL-ROUTING IN IRAN'S DNS CENSOR

Table 7: List of domains that when contained in a DNS request trigger null-routing by Iran's DNS censor. Note that all domains share the string `wpad`, the name of a proxy discovery protocol.

Domain
wpad.net
wpad.com
wpad.box
wpad.casa
wpad.com.br
showpad.com
showpad.biz
wpadmngr.com
js.wpadmngr.com
wpadvancedads.com
wpadc.org
meowpad.me
ywpadmin.com

[9] https://github.com/UPB-SysSec/IranconsistenciesData