Is Nobody There? Good! Globally Measuring Connection Tampering without Responsive Endhosts

Sadia Nourin[*†] Erik Rye[*] Kevin Bock[*] Nguyen Phong Hoang[‡] Dave Levin[*]

[*]*University of Maryland* [†]*Max Planck Institute for Informatics* [‡]*University of British Columbia*

Abstract—**Many techniques have been introduced to measure network interference—tampering performed by nation-state censors or corporate firewalls to block unwanted traffic. However, virtually all prior measurement techniques require some degree of participation from endpoints within each country of study: including VPNs, cloud providers, or volunteers willing to run measurement software on their personal devices at their own risk. However, such endpoints are not always available in all countries that tamper with connections, leaving many networks unmeasurable.**

In this paper, we present the first *global*, active, network interference measurements that require *no participating endpoints within any country of study*. Our techniques extend two recent studies that use packet sequences that trigger network interference from outside the country of study by tricking middleboxes into believing a connection exists. Our system, Mint, generalizes and automates this approach—which had previously only been applied to two countries—to allow it to apply to the global IPv4 and IPv6 Internet. We use Mint to conduct the first global measurements of network interference without using any participating endpoints, and the first comprehensive scans of IPv6 interference. We show that we are able to measure networks, autonomous systems, and even entire countries that previous methods could not. We also present several case studies that highlight how our tool can be used to perform new measurement studies of network interference.

1. Introduction

Network interference occurs when a third-party middlebox (such as a firewall) drops or resets a connection. This is a common technique performed by censoring regimes to restrict access to information, but also in more benign settings like corporate firewalls or schools.

Empirically measuring network interference is critical for informing policy makers and censorship circumvention efforts as to what is being interfered with, who is performing interference, and how they do it. Moreover, because network interference can vary drastically from one network to another [19], [43], [30], it is important that measurement efforts be as *broad* (cover as many networks) and as *deep* (cover as many domains) as possible.

To this end, a diverse set of measurement platforms and techniques have been introduced, but they share common limitations. In particular, nearly all existing measurement techniques require participating endpoints within each country of study, such as: (1) recruiting willing volunteers to run measurement software on their personal devices at their own risk [32], [14], (2) renting vantage points from commercial cloud or VPN providers [28], [19], [20], or (3) making use of unwitting live servers within each country of study [48], [35], [53], [40], [39].

When participating endpoints are available, these techniques are successful, but they face two key limitations:

Participating endpoints are often unavailable. Many autonomous systems (ASes)—especially cellular networks—lack live servers, cloud providers, or users willing to take on the ethical risks of volunteering their personal devices to be used in censorship measurement research. This problem is exacerbated in IPv6, in which it can be extremely difficult to identify live servers [56], [47]; as a result, there are no comprehensive studies of censorship over IPv6 to date.

Even when available, participating endpoints offer limited resources. To avoid saturating endpoints, researchers must limit the number of measurements they perform when using live servers or volunteers.[1] For instance, OONI [14] and Censored Planet [39] commonly test only a small number of URLs on each endpoint per measurement, e.g., less than 100 for OONI at the time of writing.

These limitations show that relying on participating endpoints within a country of study constrains the breadth and depth at which network interference can be measured.

Two recent studies introduced an alternative approach that does not require participation from within a country of study. Instead, they use "packet sequences"—subsets of a standard TCP connection—to essentially trick interfering middleboxes into believing that a connection has been established. Nourin et al. [30] used one packet sequence to study Turkmenistan's censorship infrastructure, and Hoang et al. [18] used a different one to study China's Great Firewall. These two isolated studies demonstrate that packet-sequence-based measurement can, in at least some cases, allow for measurement to take place without the standard limitations. However, we are aware of no work that has

[1] Rented VPNs and cloud providers typically do not have this limitation.

DOI 10.1109/SP61157.2025.00153

sought to apply this approach globally, nor to IPv6 anywhere.

In this paper, we present the first *global*, active, network interference measurements that require *no participating endpoints within any country of study*. In fact, our techniques do not send packets to in-use IP addresses. This permits us to study networks where there are no volunteers or servers, and we show that it is particularly effective in studying IPv6 networks, in which the vast majority of addresses are unused. Moreover, because our techniques send packets to IP addresses that are not in use, we can perform in-depth measurements without threatening the resources of an endpoint.

We present a system, Mint (Measuring Interference with Nonresponsive Targets), that generalizes this approach to the global IPv4 and IPv6 Internet. Mint takes a principled approach to discover where packet sequences could apply, and uses Geneva [9] to automatically discover which packet sequences work in a given network. We apply Mint globally—to every network in the world—using not one but six distinct packet sequences. Our results show that Mint works widely, and that all six packet sequences are necessary to get global coverage. Moreover, we show that Mint is able to trigger IPv6 tampering globally, allowing us to measure portions of the Internet that even the most popular tools, OONI and Censored Planet, have not yet been able to.

Collectively, our results show that Mint is an effective tool for *complementing* existing censorship measurement efforts: it works where many prior tools do not, and prior tools work in many places where Mint does not. To further demonstrate its utility, we perform three case studies that highlight studies it can perform that no existing platforms can. We perform the first comprehensive censorship measurement study of two popular networks within Kuwait and Pakistan, and we compare IPv4 and IPv6 censorship in a popular network in China. To further assist the measurement community, we make Mint's code publicly available at https://censorship.ai

Contributions We make the following contributions:

- We present Mint, the first tool for globally, actively measuring network interference without requiring responsive hosts within any country of study (§3).
- We evaluate Mint's applicability, finding that it can be used to widely measure IPv4 (1,462,852 /24s) and IPv6 (879,221 /48s), for a total of 9,483 ASes spanning 212 countries. To the best of our knowledge, our work is the first global, active measurement of IPv6-based network interference (§5).
- We apply Mint to perform several global studies of network interference that are possible now for the first time because of Mint's ability to measure an unprecedented number of networks (§6).
- We present three case study applications of Mint, none of which have been reported on in prior literature: two popular networks in Kuwait and Pakistan, and an in-depth comparison of IPv4 and IPv6 blocking in a popular network in China (§7).

2. Background and Related Work

Measuring Network Interference There are many active large-scale measurement platforms monitoring network interference globally. Most of these measure HTTP(S) network interference that is based on the domain name included in an HTTP GET request or in an TLS SNI (Server Name Indication) field. Some of these operate in an *inside-out* manner, in which the measurement researchers control some endpoints within the country of study and use it to issue potentially-tampered requests. For instance, the Open Observatory of Network Interference (OONI) [14] recruits volunteers around the world to run measurement software that attempts to trigger network interference. OONI faces some challenges in recruiting volunteers, especially in countries with a small population, low Internet penetration rate, or extremely repressive regime [30]. To avoid relying on local volunteers, other inside-out techniques use VPN servers within each country of study through which to issue potentially-censored requests [28], [19], [20]. VPNs are not always applicable, particularly in smaller countries where commercial VPNs are unavailable or even illegal [17], [45].

Another class of measurement techniques operate in an *outside-in* manner, in which the measurement researcher sends probes from outside the country of study to a live in-country server [48], [35], [53], [40], [39], [4], [5], [42]. The Censored Planet measurement platform [39] combines several such remote measurement techniques into a unified view of network interference. By relying on live servers, these efforts take great care in limiting which networks they probe—typically restricting to non-residential networks operating either routers or government-controlled devices. This, too, limits the networks to which prior outside-in approaches can apply. Conversely, because our approach involves sending packets only to non-responsive IP addresses, we can measure more networks and at greater load than if there were a responsive host. In §6.5, we compare the coverage of our tool to both Censored Planet's and OONI's.

All outside-in approaches are inherently limited to networks that tamper in a *bidirectional* manner; that is, they interfere whether the client is inside or outside the network. Not all networks do this; Bock et al. [8] discovered that tricking some Kazakhstan middleboxes into thinking that the client was outside the network was effective at circumventing censorship. Our technique also operates in an outside-in manner, and is thus also reliant on bidirectional censorship. In §5.1, we perform the first global measurement of the prevalence of bidirectional censorship to evaluate where our tool can possibly apply.

In addition to the above *active* means of measuring network interference, Raman et al. [38] introduced a technique for *passive* measurement. Their technique runs at web servers, and looks for tampering "signatures": packet sequences indicative of connection tampering (such as receiving multiple RSTs immediately following a request). As they note, passive and active measurement techniques are highly complementary; passive approaches are inherently constrained to what users are actively trying to access.

TCP Non-compliant Middleboxes Our work extends and generalizes prior work that exploits the fact that middleboxes are typically not TCP compliant. Bock et al. [6], in studying TCP-based amplification attacks, observed that middleboxes typically assume that they may have missed some packets within a TCP connection, and thus can be triggered to inject a blockpage by sending only a subset of the packets of a TCP connection. For instance, they showed that by simply sending a PSH+ACK packet with an offending HTTP GET request can suffice in triggering interference—with no preceding TCP three-way handshake whatsoever. They introduced several other *packet sequences* to perform amplification attacks, many of which we use in this work.

Nourin et al. [30] were the first to apply this packet sequence approach to study censorship. They studied Turkmenistan, a country that had long gone unstudied because of a lack of volunteers, VPNs, and live servers, arguably due to its extensive restrictions on Internet usage. They sent a single packet sequence to virtually all IP addresses within Turkmenistan, providing the most comprehensive view of that country's censorship to date. Hoang et al. [18] similarly used a single packet sequence, sent to non-responsive IP addresses under their control within China. Both of these studies were able to test for far more domains than other outside-in approaches that rely on live servers.

These prior studies focused on only a single country each. Our work extends theirs by asking: to what extent does packet sequence-based measurement apply *globally*?

IPv6 Censorship There has been anecdotal evidence of IPv6 network interference being deployed across various regions in the world. However, current active large-scale measurement platforms have a difficult time studying this network interference as it is difficult to find live servers within networks that we want to study in IPv6 [16], [47]. At the moment, only OONI attempts to study IPv6 network interference if a volunteer's network also supports IPv6 connectivity.

However, our measurement technique is able to fill in the gaps of IPv6 measurement. Our packet sequences are meant to be sent to non-responsive IP addresses in the first place—something that the IPv6 space is full of. As a result, we are able to conduct the first IPv6 network interference measurement using these packet sequences.

3. Mint Design

Mint's goal is to trigger interference from middleboxes without requiring any communication from an endpoint within a country of study. While prior work has done this to a couple countries (§2), Mint is the first system to do so on a global scale. Figure 1 presents a high-level overview of Mint's design, which consists of three broad components: finding non-responsive hosts, selecting domains to test, and scanning non-responsive hosts with packet sequences. We present each of these in turn.

Top ASes by /48		**Top Countries by /48**	
AS	**# /48s**	**Country**	**# /48s**
China Net (AS4134)	369,286	CN	823,674
China Mobile (AS9808)	157,698	BR	358,880
Deutsche Telekom (AS3320)	141,520	US	344,809
British Telecom (AS2856)	89,167	DE	270,894
China Unicom (AS4837)	72,158	GB	186,615
10,085 others	1,785,643	165 others	630,600
Total	2,615,472	Total	2,615,472

TABLE 1: Number of /48s learned by AS and country from a day of running 22 geographically diverse NTP servers.

3.1. Finding non-responsive hosts

Mint limits all of its interference measurement to IP addresses that are not in use. To do so, it first performs an Internet-wide scan to identify IP addresses that are non-responsive.

Non-responsive IPv4 addresses For IPv4, this is rather straightforward: we simply use ZMap [13] to send TCP SYN packets to ports 80 (HTTP) and 443 (HTTPS) for each potential IP address. To limit the amount of traffic we send to any /24 network, we did not scan every IP address, but rather chose one at random from each /24. This amounts to 2^{24} total IPv4 addresses scanned, or approximately 16M. If the selected address within the /24 responds with any packet (e.g., a SYN+ACK or a RST), we consider it responsive and thus remove the entire /24 from any future measurements.

Non-responsive IPv6 addresses Scanning IPv6 is considerably more difficult; indeed, developing scanning techniques to account for IPv6's massive address space is an active area of ongoing work [16], [47], [56]. On the one hand, it is trivial to find non-responsive IPv6 addresses, but rather than risk studying prefixes with no hosts, we sought to scan non-responsive addresses *in active prefixes*.

To this end, we followed the technique of Rye and Levin [47] to passively learn active IPv6 networks by hosting Network Time Protocol (NTP) servers. We operated 22 IPv6 NTP servers as part of the NTP Pool [31], a crowd-sourced project that distributes NTP requests to volunteers' servers using a DNS round-robin. The NTP Pool attempts to direct clients to a geographically-proximal server via IP geolocation, so we ran our servers in locations distributed across six continents: two in the US and one in each of 20 other countries[2].

Over the course of a day in October 2024, we learned 279,520,576 unique active IPv6 addresses spanning 2,615,472 distinct /48 prefixes in 10,090 ASes. Table 1 lists the top ASes and countries of the /48 prefixes we learned.

Mint then seeks to find non-responsive addresses within the known-active /48 prefixes it learns from NTP. For each of these /48s, we choose three random IPv6 addresses from three unique /64 subnets, and perform TCP SYN scans to

[2] Australia, Brazil, Cyprus, Estonia, France, Germany, Hong Kong, Hungary, India, Israel, Kazakhstan, Poland, Singapore, South Africa, South Korea, Spain, Türkiye, Ukraine, the United Arab Emirates, and the United Kingdom.

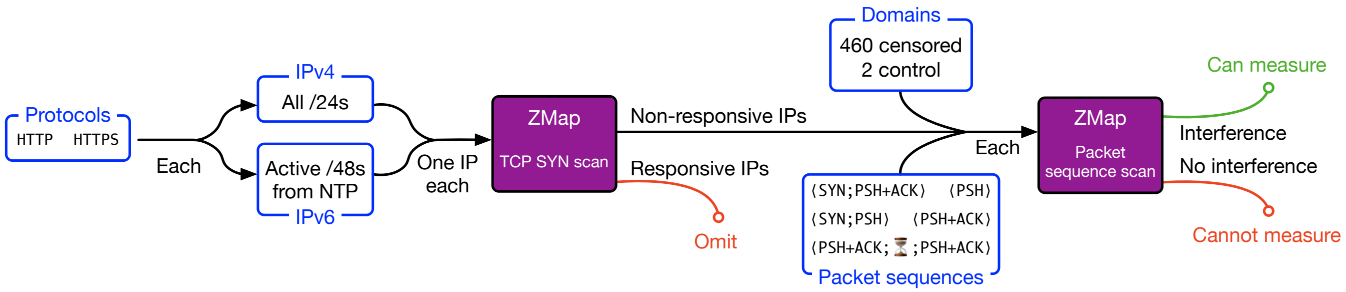


Figure 1: Overview of Mint's global measurement pipeline. We first perform a SYN scan to find non-responsive IP addresses in each network (/24 for IPv4, /48 for IPv6), and then perform subsequent packet sequence scans only to the non-responsive destinations.

them on port 80 and 443 using ZMap6, an IPv6 extension of ZMap [52], [16]. It is extremely unlikely that we would guess a responsive address; for most prior work on IPv6 scanning, this is a challenge, but because we want non-responsive addresses, it is a benefit! If any of the targets do respond, we conclude that it is due to *aliasing*, in which a single host responds to all addresses within a prefix [15]—when this happens, we remove the entire /48 from further measurements.

3.2. Domain selection

Like all active measurement methods, Mint must choose the set of domains to test for interference. Mint is agnostic to the specific domains chosen, and can be applied in a wide range of network interference measurement studies. Moreover, because it involves sending traffic to non-responsive end-hosts, it need not limit the number of probes it sends as severely as prior efforts [39], [14].

For our global scans, we used 2 control domains (example.com and our opt-out domain) and 460 domains obtained by extracting the top five domains with the most "confirmed" network interferences or anomaly responses for each country that OONI had data on, in August 2023. For our more in-depth case studies (§7), we used 100,000 domains, comprising all 17,802 domains from the Citizen Lab Test List and the top 82,198 of the Tranco-1M domain list.

3.3. Packet sequence scans

Once Mint has identified non-responsive IP addresses and a set of domains to test, it then issues packet sequences to them in an effort to trigger interference.

Packet sequences Prior work [6], [30] has identified six packet sequences known to trigger network interference in at least some networks around the world: Four packet sequences are subsets of traditional TCP connections, with the potentially-censored payload in a PSH or PSH+ACK packet: ⟨SYN; PSH+ACK⟩, ⟨SYN; PSH⟩, ⟨PSH⟩, and ⟨PSH+ACK⟩. One packet sequence includes the sensitive payload in a single ⟨SYN ⟩ packet, which is atypical (but allowed) in TCP. The final packet sequence, ⟨PSH+ACK;sleep;PSH+ACK⟩, sends two sensitive payloads separated by a 5 second sleep [30].

Mint can apply any of these packet sequences to any selection of the networks under study. In our experiments, we applied all packet sequences to all networks; in §6, we show that each of them is able to trigger networks and even countries that the others cannot.

We developed custom ZMap probe modules to send these packet sequences at high speeds. To implement the "sleep" primitive without slowing down our scans, we implemented a module that sends in the standard connectionless fashion, but then after five seconds resets the generator to revisit the IP addresses it had earlier sent to.

Reasons packet sequences can fail There are several reasons why a packet sequence may not trigger interference. *First*, the network under study may not employ bidirectional censorship; in this case, no outside-in technique could measure the network. *Second*, the network may not censor any of the domains used in testing. In this case, as well, it is not a limitation of Mint but of the domains provided to it. *Finally*, the packet sequence itself may not successfully trigger the network's middleboxes. Mint attempts to remedy this.

Discovering new packet sequences Mint can attempt to discover new packet sequences using Geneva [9], a genetic algorithm that was originally used to manipulate packet sequences to evade censorship. Geneva's fitness function can be modified so that it can manipulate packet sequences to trigger network middleboxes instead of evading them, as was done for TCP amplification attacks [6]. Yet, these existing modifications are not enough to discover new packet sequences for measuring censorship. We further modified Geneva by (1) adding in sleep functionality during training, (2) modifying the fitness function so that instead of only triggering amplification attacks, Geneva triggers all types of injections, and (3) adding an HTTPS payload.

4. Experimental Methodology

In this section, we provide details about how we performed our measurements, and also discuss ethical considerations.

4.1. Selecting non-responsive addresses

Recall from §3 that we chose the IPv4 addresses to send probes to by randomly choosing one IP address per /24. One could imagine choosing a single address within a /24 and re-using it for all of our scans—for each protocol and packet sequence. However, because our scans spanned months, picking a single address within a /24 would likely result in some scans where the address was responsive, because of IP address churn. Thus, we chose a new random address within each /24 for each packet sequence scan, and for each protocol.

We analyzed any given /24 only if it was successfully probed in *all six* of the packet sequence scans per each protocol; otherwise, we omitted it from our results. The omitted /24s are detailed in Table 13.

While we verified that the addresses we probed were non-responsive, we did not verify whether there were any responsive hosts also present in those same /24s. It is therefore possible that some of the /24s we probed had no hosts that could have experienced interference. Nonetheless, we still detect interference policies, regardless of whether anyone was there to experience them. On the contrary, in IPv6, every /48 we probed had at least one active host, because we only learned about any given /48 when one of its hosts connected to one of our NTP servers (see §3).

4.2. IP geolocation

To analyze network interference on a per-country basis, we geolocated all of the IPv4 and IPv6 networks used in our study. IP geolocation measurements can oftentimes be inaccurate, so instead of relying on one database to conduct geolocation, we opted to use three different databases: MaxMind [26], DB-IP [12], and ip2location [21]. If at least two out of the three databases agreed on the country, we assigned that country as the server's location. Otherwise, we omitted the IP address (and thus, for the non-responsive scans, the entire prefix) from further study.

Unfortunately, the version of ip2location we were using did not have IPv6 geolocation data, so for our IPv6 prefixes, we only geolocated via MaxMind and DB-IP and omitted IP addresses from our analysis that did not geolocate to the same country with both databases. We furthermore mapped the AS of each IP address in our scans via Team Cymru's IP to ASN mapping service and via Routeviews data [50].

4.3. Choosing where to train for packet sequences

As mentioned in §3, Mint is able to automatically train to discover new packet sequences. However, since training is time-intensive, we did not train globally. Here, we describe how we selected where to train.

We excluded from training any country that had: (1) Less than 35% of its /24s exhibit bidirectional interference; (2) Fewer than ten /24s that exhibited bidirectional interference; but did not have (3) A press freedom index (as determined by Reporters without Borders [44]) greater than 0.7. Of the remaining countries, we chose those with the most number of /24s that exhibit bidirectional interference.

We applied this process to both HTTP and HTTPS, and trained in the top 80 countries for HTTP and the top 10 for HTTPS. Since it is not feasible to train in every single /24 of every country, we trained in 3% of the /24s of each AS in a country. If an AS in a country had three or fewer /24s, we trained in all of them.

4.4. Measurement infrastructure

We used multiple machines located within the US to conduct all of our scans while we used two machines in Japan to conduct our Geneva training. We used an 800 Mbps scanning rate for our packet sequence scans and used the default ZGrab2 scanning rate for our bidirectional scans [58]. For our IPv4 case study, our scanning rate was at most 30 Mbps while for our IPv6 case study, our scanning rate was at most 750 Mbps. Our IPv4 packet sequence scans were run from November 2023 to December 2023 while our IPv4 bidirectional scans were run from June 2024 to July 2024. Our IPv6 packet sequence scans were run in October 2024. Our case studies were run in November 2024. For both the IPv4 and IPv6 packet sequence and the IPv4 bidirectional scans we used domains extracted from OONI from August 2023.

4.5. Ethical considerations

Our institution's IRB reviewed our experiment plan and concluded that it was not human subjects research. Nonetheless, there are potential risks to Internet users, even if they are not official human subjects. In particular, it is possible that merely sending a request for censored content to a user's machine could be misinterpreted as that user either soliciting or hosting that content.

Here, we describe the extensive steps we took in designing our experiments to minimize potential risks to users. We performed two sets of experiments, both with their own set of ethical considerations.

Experiments with non-responsive hosts Our primary set of experiments involve sending requests to non-responsive IP addresses (both IPv4 and IPv6). Moreover, the specific packets Mint sends to non-responsive IP addresses do not mimic real user behavior; they lack a full TCP handshake and application-layer interactions. We believe this inherently makes it less likely that our packets will be attributed to any actions by real users.

To ensure that non-responsive IP addresses did not have any live machines behind them, we checked for responsiveness via ZMap TCP SYN scans shortly before sending our probes with potentially-censored domains. It is possible that an IP address we found to be non-responsive was assigned to a user before or after our scans, due to IP churn. This is arguably a risk that all measurement efforts face—papers that use reflective servers in "institutional" networks [53], [35], [34], [39], [28] could similarly be subject to IP address reassignment. We minimized the risk of IP churn by ensuring that our responsiveness tests (with ZMap) were done shortly before our scans. Nonetheless, whenever we found that the destination we were going to probe was responsive, we not only did not probe it, we removed the entire prefix from further scanning altogether.

Experiments with bidirectional scans The second set of experiments we performed—our "bidirectional scans" (§5.1)—operated in a different fashion. We connected to *responsive* IP addresses openly serving on port 80 and/or 443, and sent requests to them for domains they *did not host*: three different censored domains, and our own opt-out domain. The purpose of this experiment was to largely serve as a means of validating Mint's techniques. We therefore performed far fewer bidirectional scans (a total of six) than we did with Mint, and we emphasize that the bidirectional scans are not in and of themselves part of the Mint technique (which only sends probes to non-responsive IP addresses).

Because we connected to all live servers with open ports 80 and/or 443, our scan includes both "institutional" servers (e.g., those being run by a web hosting service)—which have been used for many censorship measurements in the past—as well as residential hosts potentially associated with users. However, the manner that we interact with these hosts imposes what we believe to be minimal risk. In particular, we are only sending requests for potentially censored content to them—we are not somehow getting them to issue those requests themselves. Thus, the primary risk that our experiments pose to users would be that a network observer might think that, because the user's machine is receiving a request for a particular piece of content, then that user must therefore be hosting that content. Yet, if such a network observer were to examine the traffic, they would also observe (1) our request for our opt-out domain, which hosts a web page explaining our experiment, and (2) that the destination only receives such requests from a single IP address—our measurement server, which also hosted the same webpage. We believe that this minimizes any potential risk to users.

Comparison to prior approaches Comparing our approach to other censorship measurement efforts, we believe that the techniques we have taken in this paper expose comparable or less risk overall. Most existing censorship measurement techniques [14], [53], [35], [34], [39], [28] elicit machines in censored countries to request the censored domain, either by controlling the vantage point in the censored country or by using the Echo protocol. Unlike such methods, we have only sent *requests* for the censored domain to the destination IP addresses—which we believe reduces risk, as explained above. Although some of these previous measurement techniques limit their vantage points to data centers and institutional networks [53], [35], [34], [39], [28], measurement techniques that do use vantage points in residential networks [14], [22] send the measurement probe directly from an in-country machine. Conversely, Mint, and measurement studies it builds off of [18], [30] may also send probes to residential destination IP addresses, but we never cause requests for the censored domain to be initiated by any in-country machine, in a residential network or otherwise.

Scanning considerations We restricted the amount of traffic sent to individual routers during our scans. Since we used ZMap for our scans to non-responsive IP addresses—which randomizes IPs while scanning—and we chose to only scan approximately one IP address per /24 prefix, the volume of traffic to individual routers was kept low during our scans. For the bidirectional scans to live servers, we limited our measurement only to two requests: one to a non-sensitive domain and one to a sensitive domain. In addition, we slept for 60 seconds between each request to the live servers as to not overwhelm the routers on the path to the live servers as well as the live server itself.

During all of our scanning, we included our research opt-out domain as the control domain in our experiments and hosted the opt-out page on port 80 from our scanning machine. We received opt-out requests from two networks via these domains and promptly removed their IP addresses from future scans.

5. Validation and Evaluation of Mint

Mint does not necessarily work in all networks. For it to succeed, three criteria must be met: the network must employ bidirectional interference, it must have non-responsive IP addresses, and we must have a packet sequence that can trigger interference when sent to a non-responsive address. In this section, we analyze each of these three criteria to evaluate how widely Mint can apply.

5.1. Global prevalence of bidirectional interference

To understand how common bidirectional interference is, we performed a validation experiment in which we connected to live HTTP and HTTPS servers. We emphasize that this is merely a validation experiment and not part of Mint's core design.

Methodology The basic intuition behind this experiment is that if we issue two separate requests to a live server—one for a control domain that is very unlikely to trigger interference, and another for a domain that is likely to trigger interference—then we can detect interference if the responses to the two requests are significantly "different." We do not expect successful responses for either domain (since neither is hosted by the servers), but we do expect the two responses from a given IP address to be the same (e.g., an HTTP 404)—unless there is interference. The fact that the destination IP address does not host the content likely does not stop middleboxes from interfering, as they cannot easily verify the correct host for a given domain.

We first ran a ZMap TCP `SYN` scan on port 80 and 443 across the entire IPv4 space, providing us with the set of all live HTTP and HTTPS servers, respectively. We then geolocated the IP addresses of these live servers to learn the country they were in.

To determine if middleboxes tamper from the *outside-in* direction, we performed what we call *bidirectional scans*. We used ZGrab2 [58] from clients outside the networks of study to request censored and uncensored domains from responsive servers inside. For each destination, the ZGrab2 client completed the TCP three-way handshake, issued an HTTP(S) request for a domain, waited for a response, and then terminated the connection. We split our experiment into two scans: the first sent requests for the control domain—our own domain that, if visited over port 80, provides a website allowing users to opt-out of further study (§4.5). The second scan sent requests for the domain we expected to trigger network interference in that live server's country. In particular, for each country, we chose the domain that, according to OONI, had the most "confirmed" network interference incidents *or* anomaly responses. We slept for 60 seconds between the request to the control domain and the request to the sensitive domain. We repeated this process for the second-most and third-most interfered-with domains, for a total of six scans.

We then analyzed the responses to our control and likely-censored requests to determine if they were "different," which is a sign of bidirectional network interference. For HTTP, we deemed the responses different if they had different HTTP status codes, different network statuses, or different errors, as defined by ZGrab2. In the event that both the censored and non-censored queries resulted in the same status code and also returned an HTML page, we evaluate difference by computing the cosine similarity of the two HTML pages, as suggested by Jones et al. to detect block pages [23]. In such cases we did not find any instances where the cosine similarity was less than 0.816, a threshold set by Jones et al. for equality, indicating that we did not experience any block pages for censored content while receiving legitimate pages for non-censored content.

For HTTPS, we considered the responses different if they had either different network statuses or if ZGrab2 provided different errors in its certificate validation. A limitation with this approach is that, for HTTPS, networks and ISPs can mandate customers to accept government root certificate authorities, or can serve self-signed certificates [37], [24]. This would result in responses that do not appear to be "different," but should nonetheless be considered tampering. We analyzed whether the responses from the servers we received were self-signed and discovered that 14,475 of the servers we probed returned self-signed certificates for both the censored and the uncensored domain. However, there are plausible benign explanations for this; many in-home devices use self-signed certificates [10]. One area of future work would be to develop more nuanced methods of detecting censorship from these responses.

Results Collectively, we performed bidirectional scans for 48,517,741 live server IP addresses for HTTP and 46,609,131 for HTTPS. We observed different responses for 7,779,282 (16.0%) of the IP addresses for HTTP and 2,514,724 (5.4%) for HTTPS. This corresponds to 967,726 /24s (5.8%) for HTTP and 618,577 /24s (3.7%) for HTTPS. While, these percentages may seem small, the overall number of networks is in the millions, far more than any existing censorship measurement platform has today.

Country	# IPs w/ differing responses	# IPs probed	Ratio	Press Freedom Score
Belarus	26,715	31,073	0.860	0.268
Myanmar	3,992	3,680	0.853	0.244
Eritrea	22	27	0.815	0.166
Djibouti	198	267	0.742	0.301
Mali	4,486	6,246	0.718	0.506

TABLE 2: Top 5 countries with the highest ratio of bidirectional tampering (HTTP).

Country	# IPs w/ differing responses	# IPs probed	Ratio	Press Freedom Score
Belarus	20,529	25,520	0.804	0.268
Djibouti	250	335	0.746	0.301
Myanmar	3,590	482	0.745	0.244
Mali	3,165	5,811	0.545	0.506
Tanzania	2,636	5,001	0.527	0.548

TABLE 3: Top 5 countries with the highest ratio of bidirectional tampering (HTTPS).

The results of these bidirectional scans do not map directly to the number of networks measurable by Mint. For the /24s where there was at least one responsive host, the bidirectional scan represents an upper bound of Mint's coverage. This is because bidirectional interference is a necessary but not sufficient condition for Mint's packet sequence measurements to work; they also require middleboxes to operate in a stateless manner. For the /24s where there were no responsive hosts, we could not perform a bidirectional scan, but it is still possible that Mint would be able to trigger interference.

Tables 2 and 3 present the five countries with the highest fraction of IP addresses that bidirectionally interfere for HTTP and HTTPS, respectively. Interestingly, these are only loosely correlated with their Press Freedom Scores.

5.2. Prevalence of non-responsive IP addresses

The second necessary condition for Mint to work is that it be able to find non-responsive IP addresses. Recall from §4 that we analyzed a /24 only if we were able to find a non-responsive IP address within it for each of the six separate packet sequence scans. Likewise, we analyzed an IPv6 /48 only if *none* of the random addresses we selected within it were responsive.

IPv4 results Out of the 2^{24} /24s on the IPv4 Internet, we were able to probe 15,158,447 (90.4%) of them for *all six* HTTP packet sequence scans and 14,981,549 (89.3%) of them for all six of the HTTPS packet sequence scans. This corresponds to 72,183 (94.2%) ASes across 242 (97.2%) countries for HTTP, and 72,514 (94.7%) ASes across 241 (96.8%) countries for HTTPS. These are highly encouraging numbers; they indicate that IPv4 addresses are not so densely in-use that it is not possible to find non-responsive hosts to probe.

Protocol	# Prefixes	# ASes	# Countries
HTTP / IPv4	1,303,570	6,871	200
HTTPS / IPv4	1,262,779	6,551	201
HTTP / IPv6	862,892	383	80
HTTPS / IPv6	860,780	528	74

TABLE 4: Number of prefixes (/24 for IPv4, /48 for IPv6), ASes, and countries for which Mint is able to trigger censorship without any endpoint participation.

To better understand where Mint cannot apply, we analyzed the 10% of the /24s where not all of our scans identified non-responsive hosts. Most of these networks were registered by telecom companies and cloud providers: two network types that often use the vast majority of the IPv4 addresses. We present the top 10 least-covered ASNs for both HTTP and HTTPS in Tables 13.

IPv6 results Of the 2,615,472 distinct /48s we learned of, only 10,892 of them responded to any of our SYN scans for HTTP and 10,573 for HTTPS, meaning that we could probe more than 99% of the prefixes for both protocols. This corresponds to us being able to probe 10,056 (99.7%) of the 10,090 active IPv6 ASes we learned of for both HTTP and HTTPS. These are also highly encouraging results; they show that IPv6 aliasing is not widespread, and thus that Mint is able to at least attempt to probe nearly all active IPv6 prefixes.

5.3. Success rate of packet sequences

The final necessary condition for Mint to work is that Mint should be able to successfully trigger interference by sending packet sequences to a non-responsive host. Table 4 summarizes the number of /24s, ASes, and countries we could trigger results across all protocols. We tabulate interference to an AS or country if we observe at least one /24 prefix in our results.

IPv4 results We were able to successfully trigger HTTP interference to 1,303,570 distinct /24s (8.6% of the 15,158,447 that we were able to probe). This spans 6,871 ASes (9.5% of those we could probe) and 200 countries (82.6% of those we could probe). Similarly, for HTTPS, we could trigger interference to 1,262,779 /24s (8.4%), 6,551 ASes (9.0%), and 201 countries (83.4%). Again, while the fraction is relatively small, the raw number of networks is orders of magnitude more than what is possible with existing techniques.

IPv6 results For IPv6, we were able to trigger HTTP interference to 862,292 /48s (36.9% of the /48s we could probe), which corresponds to 383 ASes (3.9%) and 80 countries. For HTTPS, we were able to trigger 860,780 /48s (33.0%), 528 ASes (5.3%) and 74 countries. Interestingly, for IPv6, we are able to trigger a smaller fraction of ASes, but a significantly higher fraction of /48s.

Collectively, these results show that Mint *is able to successfully trigger censorship without participating endpoints* in a huge swath of the Internet. Moreover, the set of countries we can study is widely diverse. Figure 2 shows a map of all countries where we are able to trigger for IPv4 and IPv6. Table 5 shows the top 5 countries with the most connection tampering that we could detect using Mint.

5.4. Detecting and filtering overblocking

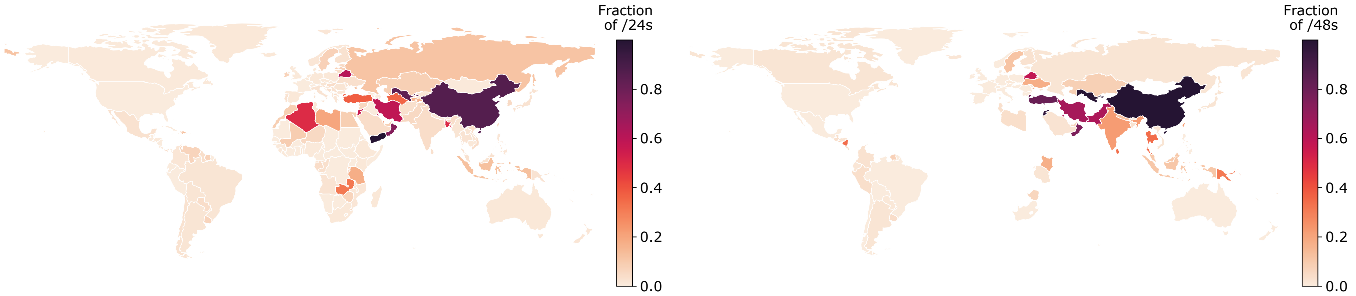
We have identified some networks that *overblock* by interfering with more domains than one would expect from a sane blocklist policy. For example, we observe some networks that block *all* of the domains we sent, including our control domains.

Why do we observe overblocking? There are several possible causes of overblocking. *First*, what appears to us to be overblocking could be middleboxes applying an *allow-list*: only domains on the allow-list would pass through without interference. This has been observed in prior work, and even associated with censorship within countries that also block via blocklists, as evidenced by AS201558 in Turkmenistan [30]. *Second*, overblocking could indicate a middlebox that potentially tampers with all incoming traffic depending on properties beyond just the domains in the traffic. For example, it could indicate an intrusion detection system (IDS) that tolerates the first few packet sequences before actively engaging against subsequent packets. Thus, overblocking may represent intentional policy, but most likely not the kind of blocklist policies that we are analyzing for in this paper.

Identifying and filtering out overblockers Whatever the cause, overblockers risk over-inflating our results, so we filter them out. In all of our analyses (in the preceding and following sections), we have omitted each /24 for which we observe interference with our control domains *or* interference with over 95% of the domains we test with.

Table 6 presents the number and type of overblockers by IPv4/IPv6 and HTTP/HTTPS. Ultimately, we omitted 17.4% of all interfering prefixes for HTTP/IPv4, 17.1% for HTTPS/IPv4, 1.8% for HTTP/IPv6, and 2.8% for HTTPS/IPv6. We observe a significantly lower fraction of overblockers in IPv6, possibly indicating fewer IDS deployments in IPv6. We found no correlation between overblocking and specific ASes, network types, or countries; all seemed to have roughly equal probability of containing overblockers. This indicates that omitting overblockers does not bias our results with respect to any particular network, network type, or country.

We find that control domains alone do not suffice in identifying all overblocking. A small fraction of prefixes did not interfere with the control domain but did interfere with more than 95% of our test domains. For example, this accounted for only 0.4% of the overblockers for HTTP/IPv4. During our scans, we always sent our control domains first. Thus, we believe this form of overblocking can be explained by middleboxes that, after a certain number of probes, begin to interfere with *all* of our packet sequences. We analyze this further in §8 where we evaluate the presence of IDSes in our data.



(a) HTTP IPv4 Network Interference

(b) HTTP IPv6 Network Interference

Figure 2: Map of where Mint can trigger HTTP network interference over IPv4 and IPv6

IPv4 HTTP Network Interference				**IPv6 HTTP Network Interference**			
Country	**# of Responding /24s**	**# of /24s Measured**	**Ratio**	**Country**	**# of Responding /48s**	**# of /48s Measured**	**Ratio**
Yemen	832	913	0.911	China	821,194	821,277	0.999
China	1,062,312	1,349,991	0.787	Uzbekistan	1,711	1,723	0.993
Uzbekistan	790	1033	0.765	Jordan	2,924	3,046	0.960
Oman	2,653	3,934	0.674	Turkey	7,025	8,711	0.806
Belarus	3,584	6,430	0.557	Oman	49	65	0.754

TABLE 5: Top 5 countries with highest fraction of IPs that Mint can trigger without endpoint participation (HTTP).

	IPv4		**IPv6**	
Overblocks	**HTTP**	**HTTPS**	**HTTP**	**HTTPS**
>95% of domains	116,436	97,796	9,419	16,390
Controls	272,827	259,199	15,016	24,927
Both	115,353	96,611	8,556	16,124

TABLE 6: Number of prefixes (/24s and /48s) that overblock, either by interfering with >95% of the domains we tested or by interfering with the control domains.

We emphasize that all of the results throughout this paper (including the preceding sections) have already filtered out overblockers, leaving the networks that apply more traditional blocklist policies.

5.5. Discovering new packet sequences

We analyzed the /24s where we were able to observe a difference between the uncensored and censored HTTP and HTTPS bidirectional scans and compared these to the /24s where we were able to trigger interference with at least one domain. We discovered that we could not trigger injections from 779,404 /24s in HTTP (and 502,931 /24s in HTTPS). We wanted to understand whether our existing packet sequences were not exhaustive enough to cover these /24s.

We used Geneva to train for more packet sequences, as described in §3. We prioritized training in countries with the largest fraction of /24s with bidirectional censorship but without packet sequences that trigger it. We trained in the top 80 countries for HTTP and top 10 for HTTPS.

Unfortunately our runs with Geneva did not discover new unique packet sequences. However, there were several instances of discovering *similar* or "subset" packet sequences. For example, Geneva reported that sending a PSH+ACK packet twice triggers network interference in Tanzania (example AS: AS33765). However, this is a subset of an existing packet sequence (⟨PSH+ACK;sleep;PSH+ACK⟩), and therefore is simply a superset of this new packet sequence of sending a PSH+ACK packet twice.

A key limitation of this approach is that we are only able to measure network tampering that occurs due to packet injections, so we may not have been able to trigger network interference in these /24s because either (1) the interference is bidirectional but *stateful* (the middleboxes in the network only interfere with traffic that is fully TCP-compliant, maintaining state only for actual connections), or (2) the interference drops traffic instead of sending injections. For example, our stateful scans identify Myanmar as one of the top countries with the most bidirectional network interference, but we did not find any packet sequences that can trigger this interference. This implies that their middleboxes may be more TCP-compliant compared to those employed by other countries. Similarly, our stateful scans find bidirectional network interference over HTTPS in Uganda (AS327724) by dropping traffic, which we cannot trigger using our packet sequences.

5.6. Summary

These validation results demonstrate that Mint is able to trigger interference without endpoint participation in a large number of prefixes, ASes, and countries. That said, it is not a panacea; there are still many networks where it does not apply, because either the networks do not tamper bidirectionally, or we could not identify a non-responsive IP address, or we could not find a packet sequence that triggers tampering. Additional training could potentially improve coverage of packet sequences, but already this technique

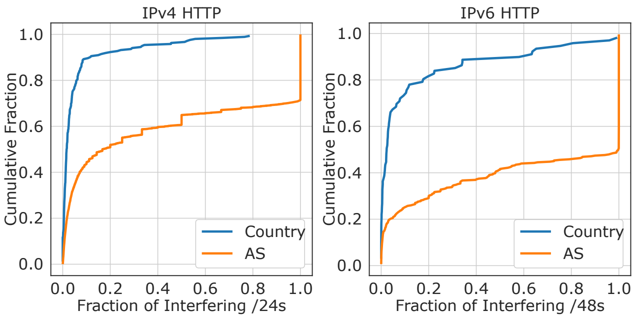


Figure 3: CDF of the fraction of interfering /24s and /48s in IPv4 and IPv6 over HTTP

Frac. of /24s	AS	CC
0.947	Viettel-Tanzania (AS327885)	TZ
0.993	Iran Telecommunication Company PJS (AS58224)	IR
0.927	Dreamline Co. (AS9457)	KR
0.981	Kungliga Tekniska Hogskolan (AS2839)	SE
0.992	Universidade Estadual Paulista (AS53166)	BR

TABLE 7: Fraction of /24s interfered with for selected ASes over HTTP in IPv4.

applies to far more networks than prior techniques have been able to.

In the following sections, we apply Mint to study network interference globally (§6) and to perform narrower case studies (§7).

6. Global Results

Having validated Mint in the previous section, we now present results from having applied it to perform global scans. We scanned each non-responsive network with all six packet sequences and 460 censored domains, as outlined in Figure 1.

6.1. What fraction of countries/ASes experience interference?

We begin by analyzing how many networks within each country experience network interference. This question would be virtually impossible for most other tools to be able to answer, as they do not have as much purview into networks.

We plot in Figure 3 the fraction of each AS's and country's /24s for which we were able to trigger interference for HTTP. Figure 6 in the appendix shows similar results for HTTPS interference. Those with a low fraction experience very little interference; these tend to include non-coordinated, individuals' firewalls. Conversely, ASes with a large fraction of prefixes interfered with tend to reflect corporate or university firewalls, and countries with large fractions tend to be nation-state censors.

The ASes with the highest fraction of /24s or /48s with interference range from large Telecom providers, such as Iran Telecommunication Company PJS (AS58224, 99.3%), to University networks, such as Universidade Estadual Paulista (AS53166, 99.2%). Other top ASes for IPv4 with the most interference include Viettel-Tanzania (AS327885, 94.7%), Dreamline Co. (AS9457, 92.7%), and Kungliga Tekniska Hogskolan (AS2839, 98.1%). For IPv6, China Mobile (AS9808) has one of the highest fraction of /48s we observed interference, at 99.9%. Other top IPv6 ASes with a high fraction of interference are Kyivstar PJSC (AS15895, 84.7%), Tunisie Telecom (AS327934, 84.5%), National Mobile Telecommunications Company K.S.C.P. (AS29357, 79.2%), and Bharti Airtel Ltd. (AS45609, 84.5%). We do not see many educational institutions interfering with traffic over IPv6 compared to IPv4, however, this may be an artifact of how we obtained the /48s to use for our IPv6 measurements.

Frac. of /24s	AS	CC
0.999	China Mobile (AS9808)	CN
0.845	Tunisie Telecom (AS327934)	TN
0.792	Nat'l Mobile Telecom. Company KSCP (AS29357)	KW
0.845	Bharti Airtel Ltd. (AS45609)	IN
0.847	Kyivstar PJSC (AS15895)	UA

TABLE 8: Fraction of /48s interfered with for selected ASes over HTTP in IPv6.

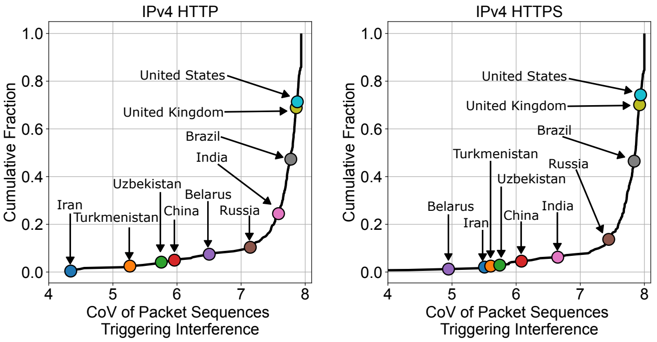


Figure 4: CDF of the CoV of packet sequences triggering interference for IPv4 over HTTP(S) with selected countries annotated

Collectively, these results demonstrate Mint's strength in being able to study a very large number of /24s at a nation-wide and global scale.

6.2. How centralized is interference infrastructure?

Censorship measurement researchers commonly categorize countries as having "centralized" or "distributed" censorship, depending on whether there is a single coordinated censorship infrastructure (like China) or if there is a broadly distributed infrastructure (like India).

One of the surprising benefits of Mint's reliance on packet sequences to trigger interference is that they can serve as a coarse-grained fingerprint of censorship infrastructure. Fundamentally, packet sequences exploit some idiosyncratic behavior of a middlebox, and thus it is likely that two different manufacturers would be triggered by a different set of packet sequences.

As such, we use the variability in the sets of packet sequences that work in /24s across a country as a measure for how diverse its censorship infrastructure is. If centralized, the same packet sequences should work everywhere; if highly decentralized, the successful packet sequences should vary. More precisely, for each country, we computed the packet sequences combinations that each /24 and /48 can trigger. Since we have 6 packet sequences, we had 64 (2^6) combinations. We computed, for each country, the coefficient of variance (CoV; the variance normalized by the mean) over these 64 possibilities. Essentially, if there is a low CoV, then there is evidence of centralized network tampering; if there is a high CoV, then there is evidence of distributed or decentralized tampering.

Figure 4 shows the cumulative fraction of CoVs, and indeed it strongly matches our hypothesis. The nations states known to conduct centralized tampering (e.g., Iran, Turkmenistan, Uzbekistan) have low coefficient of variances, while the nations states known to tamper with traffic in a distributed (Russia) and decentralized (US) manner have high coefficients of variance. This phenomenon also appears over IPv6, as shown in Figure 9 in the Appendix. It is interesting to note, however, that countries do not necessarily have the same coefficient of variance between both HTTP and HTTPS protocols. For example, Iran has a higher coefficient of variance for HTTPS than HTTP, hinting at different, decentralized infrastructure being used to conduct HTTPS network tampering compared to HTTP.

In addition to showing individual countries' centralization of interference, these results also show the global distribution. We see that, as expected, the vast majority of countries exhibit highly decentralized interference, indicating a lack of nation-state censorship infrastructures. However, it is surprising that 10–15% of countries exhibit centralized interference at a level more so than Russia, which is known to use a mix of centralized and distributed mechanisms. Mint's ability to measure all countries allows for unprecedented analysis like these.

This result not only confirms anecdotal evidence; it provides for the first time a concrete, quantifiable measure of precisely how centralized interference infrastructures are. Such a result is only possible because of the breadth of measurements that Mint can perform.

6.3. How centralized is interference policy?

Mint can also determine how centralized countries' network interference policies are. In particular, we investigate how consistent the blocklists are across different networks within a given country. This would be an especially difficult analysis to perform with prior tools, as their coverage is typically limited to a small number of networks.

For each pair of /24s or /48s in a given country, we calculated the number of domains that were interfered with in both. We normalized this number by the total number of domains, and then calculated the coefficient of variance over the normalized domain count for each pair of /24s or /48s for a country. Similar to §6.2, if there is a low CoV, then there is evidence of a centralized blocklist; if there is a high CoV, then there is evidence of a decentralized blocklist.

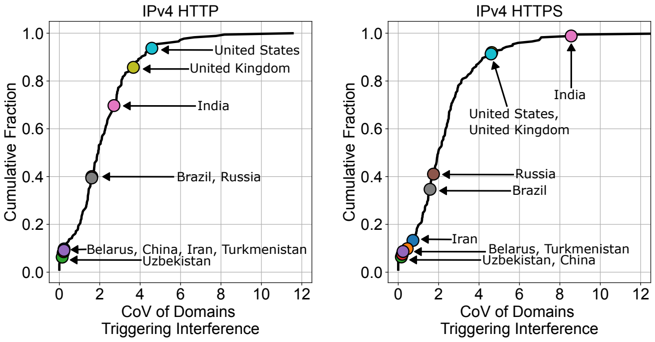


Figure 5: CDF of the CoV of domains triggering interference for IPv4 over HTTP(S) with selected countries annotated

Figure 5 shows the cumulative fraction of these CoVs. Countries with more centralized interference infrastructure (such as China and Iran) tend to also have more centralized interference policy. Likewise, countries with less centralized censorship infrastructure (such as Russia and India) tend to have less centralized interference policies. Similar results hold for IPv6, as shown in Figure 10 in the Appendix.

6.4. Are multiple packet sequences necessary?

In Section 3, we presented six distinct packet sequences—all from prior work—that can remotely trigger tampering without a responsive end-host in the country of study. Here, we ask: are all of these packet sequences necessary?

To answer this question, we analyzed how many of the countries, ASes, and /24s or /48s that we triggered were *uniquely* triggered by one of the six packet sequences. This is showcased in Table 9 for our HTTP packet sequence scans over IPv4. Overall, we find the ⟨SYN; PSH+ACK⟩ packet sequence is (by far) the most successful, being able to trigger 89.9% (1,172,413/1,303,570) of the total /24s we triggered for HTTP, which is similar to the findings of Bock et al. [6].

Still, we find having multiple different packet sequences is important for broad measurement coverage. For example, for HTTPS over IPv4, the ⟨SYN; PSH⟩ packet sequence uniquely triggers network interference in 5 countries. In fact, every packet sequence can uniquely trigger interference in at least one country (across HTTP/HTTPS and IPv4/IPv6) uncovered by any others. Similar analysis for the IPv6 HTTP and both the IPv4 and IPv6 HTTPS packet sequence scans can be found in Tables 15, and 16 in the Appendix.

6.5. Comparison to other techniques

We compare Mint to two popular network interference measurement platforms, OONI [14] and Censored Planet [39].

	Triggered			Uniquely Triggered		
Packet Sequence	Countries	ASes	/24s	Countries	ASes	/24s
⟨SYN; PSH+ACK⟩	190	5,468	1,172,413	7	1,057	71,578
⟨PSH+ACK⟩	178	4,588	108,296	1	617	23,830
⟨PSH+ACK;sleep;PSH+ACK⟩	177	4,796	129,138	3	705	26,439
⟨SYN; PSH⟩	172	4,333	1,123,123	0	528	33,123
⟨SYN⟩ (with payload)	159	2,397	59,482	1	423	18,109
⟨PSH⟩	107	1,906	44,328	1	49	2,266

TABLE 9: Mint uses six distinct packet sequences to trigger interference. Each of them spans a large number of networks, and is able to trigger networks the other packet sequences cannot.

	ASes		Countries	
Platform	HTTP	HTTPS	HTTP	HTTPS
Mint	6,871	6,551	200	201
Censored Planet	853	819	159	165
Common	523	522	153	158

TABLE 10: Number of ASes and countries measurable with Mint and Censored Planet.

Censored Planet We compared all of the ASes and countries where Censored Planet had an unexpected response rate of greater than 0%. This is because our tool can reach virtually all networks with non-responsive IPs in them, so a more fair comparison would be to compare the number of triggerable ASes and countries between the two tools. Censored Planet only runs over IPv4, so we compared our IPv4 packet sequence scans conducted from November 2023 to December 2023 with Censored Planet's measurements from November 2023 to December 2023.

The results from this comparison are presented in Table 10. While Mint is able to trigger in thousands more networks than Censored Planet, there are still hundreds of networks and even a handful of countries where Censored Planet is able to measure and Mint is not. We were curious whether Censored Planet and Mint are able to study the same kinds of networks, (Cable/DSL/ISP, NSP, Content, Educational/Research, Enterprise, etc.), or if either has a strong bias towards or against certain kinds of networks. Using PeeringDB [36], we analyzed the types of all ASes in which either tool was able trigger interference. We found no discernible difference; both tools predominantly trigger interference in Cable/DSL/ISP and NSP networks, the most common networks, and have representation within all other types of networks. We conclude from this that the tools do not fundamentally differ in the kinds of networks they can measure.

That said, there still are fundamental differences between the tools that make them powerful complements of one another. Censored Planet is able to measure interference from networks that have bidirectional and both stateful *and* stateless middleboxes, while Mint is only able to measure interference from networks that have only bidirectional and stateless middleboxes. For example, Mint can only measure interference in 2% of the /24 networks in Myanmar over both HTTP(S), even though Myanmar has bidirectional tampering in 85.3% of their /24 networks over HTTP and 74.5% of their /24 networks over HTTPS, as detailed in Table 2 and Table 3. However, Censored Planet can measure interference in Myanmar in 4 ASes over HTTP and 3 ASes over HTTPS, of which only 1 is in common with Mint over both protocols.

OONI We compare all of the ASes and countries where OONI flagged at least one measurement as being an anomaly. We did this instead of comparing with all of the networks that OONI can measure because, just like in the case of our Censored Planet comparison, Mint can measure virtually any network, so we opted to measure just the networks where both platforms could trigger interference. OONI is able to conduct both IPv4 and IPv6 network interference measurements, but they are limited by whether their volunteers are located in IPv4 or IPv6 networks when they run their network measurements. Therefore, we are able to compare both the IPv4 and IPv6 networks that both Mint and OONI can measure directly. We compared our IPv4 scans conducted from November 2023 to December 2023 to OONI's IPv4 measurements from the same time range. Likewise, we compared our IPv6 scans conducted in October 2024 with OONI's IPv6 measurements from the same time range.

This comparison is shown in Table 11. Over IPv4, Mint measures thousands more ASes and dozens more countries than OONI. In contrast, OONI outperforms Mint over IPv6, covering significantly more ASes and countries. This may be because we were only able to obtain /48 prefixes to measure from the limited number of live clients that sent requests to our NTP servers. There may be many more IPv6 networks where Mint can work successfully which we do not know of. Nonetheless, just like with comparison to Censored Planet, both Mint and OONI are able to measure networks that the other tool could not.

These results show that Mint is highly complementary to these existing techniques. In performing measurements of network interference, having multiple, complementary techniques is critically important. Moreover, even being able to redundantly measure the same networks has value; tools like OONI and Censored Planet can provide closer to ground-truth assessment of whether interference is actually happening, because they make real connections with live endpoints. Mint's value comes in its ability to measure where these tools cannot, and to measure at greater scale where these tools can.

	IPv4				IPv6			
	ASes		Countries		ASes		Countries	
Platform	HTTP	HTTPS	HTTP	HTTPS	HTTP	HTTPS	HTTP	HTTPS
Mint	6,871	6,551	200	201	383	528	80	74
OONI	1,975	2,118	155	153	1,069	1,418	129	138
Common	994	1023	151	148	138	178	72	70

TABLE 11: Number of ASes and countries measurable with Mint and OONI.

7. Case Studies

Our global measurements reveal diverse patterns of network interference, spanning thousands of networks worldwide. These findings underscore the utility of Mint in detecting previously unmeasurable censorship phenomena. To demonstrate the practical applications of our approach, we examine specific networks with unique censorship behaviors. These case studies highlight how Mint facilitates a deeper understanding of both regional and protocol-specific interference mechanisms. In this section, we present three case studies to illustrate these insights in Kuwait, Pakistan, and China.

7.1. Methodology

For our case studies, we first manually confirmed that our chosen packet sequence for the given AS that we wanted to measure was indeed triggering a network middlebox. We did this by either inspecting the resulting blockpage, such as with Kuwait, to observe whether it was a blockpage from the destination country that we wanted to measure, or by TTL-limiting the packet sequence and observing whether the injection that we received came from a hop that was located within the country that we wanted to study, such as with Pakistan and China.

After we confirmed the validity of our packet sequence, we confirmed that the network interference occurred on all of the ports, and not only port 80 and 443. Finally, we tested for whether the country deployed any residual censorship, censorship that interferes with *all* traffic from a client for an extended period of time, triggered by the client previously attempting to access a censored domain. We test for residual censorship by sending a packet sequence with a censored domain in the payload, followed by a packet sequence with an uncensored domain in the payload.

If we did not observe a blocking response for the packet sequence with the uncensored domain in the payload, then there is no residual censorship, as was the case for Kuwait and Pakistan. However, if we did observe blocking, then the AS and country was deploying residual censorship, which we needed to evade when conducting measurements, as was the case for China. For our case studies, this was as simple as changing the destination port for each of our measurement probes, as China deploys residual censorship at the TCP 3-tuple level (source IP, destination IP, and destination port). Therefore, changing the destination port is enough to evade residual censorship.

7.2. IPv4 Case Study: AS6412 in Kuwait

AS6412 in Kuwait is a relatively understudied autonomous system. As of this writing, it has only 21 measurements from OONI [3] and no Censored Planet measurements within the past year [1], [2]. This contrasts sharply with AS42961, the largest AS in Kuwait, which has been extensively studied by both OONI and Censored Planet.

We discovered that the ⟨SYN; PSH+ACK⟩ packet sequence triggers blockpage injections for IPv4 in AS6412 when the payload contains a censored domain. This AS has no active IPv6 prefixes as of the time of our measurements. We scanned non-responsive addresses on November 4, 2024 for 100K domains, as described in §7.1.

We found 4,765 censored domains over HTTP and 4,896 over HTTPS. Among these, 3,089 domains were common to both protocols. Using VirusTotal's classification service [54], we could categorize 1,067 domains, with top categories being Pornography (303 domains), Business/Economy (207 domains), and Information Technology (89 domains).

Our analysis reveals that AS6412 blocks fewer domains than the more commonly measured AS42961, but the blocked domains in AS6412 are a strict subset of those blocked by AS42961. Specifically, for the 462 domains used in our packet sequence scans, AS42961 interfered with HTTP probes of 459 domains and HTTPS probes of 460 domains, while AS6412 interfered with only 104 HTTP domains and 22 HTTPS domains.

7.3. IPv6 Case Study: AS24499 in Pakistan

AS24499 in Pakistan exhibits distinct censorship policies for IPv4 and IPv6 traffic. Middleboxes of this AS drop traffic over IPv4, while they inject teardown packets for IPv6 traffic when conducting network interference. This difference allows us to use our methodology, which relies on injected packets as indicators of network interference, to measure IPv6 interference. However, the absence of injected packets over IPv4 limits direct measurement. Nevertheless, we use the IPv6 censored domain list as a proxy to infer potential IPv4 censorship policies.

Our measurements of IPv6 HTTP(S) interference on November 3, 2024, using the 100K domains described in §7.1, identified 1,866 censored domains for HTTP and 2,001 for HTTPS. Of these, 1,745 domains overlapped between the two protocols. VirusTotal's classification service [54] categorized 255 domains, with 194 in Pornography, 26 in Adult/Mature, and 7 in Business/Economy categories.

Protocol	IPv4	IPv6	Both
HTTP	2,685	2,330	2213
HTTPS	2,228	1,908	1,796
Both	820	859	791

TABLE 12: Number of domains blocked in AS4808 in China of the 100K tested.

These results underscore the significance of Mint's capabilities in detecting IPv6 network interference, especially in networks where IPv4 measurements are challenging.

7.4. IPv4 and IPv6 Case Study: AS4808 in China

Our analysis of the packet sequence scans showed that AS4808 censors different domains over IPv4 and IPv6, implying that there are different policies in place for both protocols. Table 12 summarizes the number of domains censored for HTTP and HTTPS across IPv4 and IPv6, using measurements from November 10th–11th, 2024. This table shows that in order to get a complete picture of the number of domains censored within an AS, we must measure both IPv4 and IPv6.

These results show that IPv4 and IPv6 censored domain lists are not identical, emphasizing the importance of measuring both protocols to obtain a more comprehensive view of China's censorship practices.

Residual Censorship During our case study, we noticed that China also conducts residual censorship over IPv6. Previous studies have noted that China deploys residual censorship on the basis of the 3-tuple of a TCP connection (source IP, destination IP, and destination port) [18], [11], [7], [41], [55], [57]. We discover that this is also the case over IPv6. We tested not only using the same exact /128 of both the client and the server, but the /127, /126, and so on, as well. The only address that experienced residual censorship was the precise /128 that had been censored in the first place.

We had anticipated that residual censorship may occur over more than just the TCP 3-tuple, because of how IPv6 addresses are used. In IPv6, end-hosts must be allocated at least a /64 prefix to allow for Stateless Address Autoconfiguration (SLAAC) [51], [27]; best practices for IPv6 recommend assigning a /56 or /48 [46]. It would therefore be trivial for a client to simply change the lower bits of the IPv6 address and avoid residual censorship.

Prior work has noted these IPv6-specific challenges with blocklisting prefixes containing abusive hosts, while simultaneously minimizing collateral damage to non-abusive hosts in adjacent prefixes [25]. These observations highlight the evolving dynamics of IPv6 censorship and its distinct challenges as well as opportunities for evasion compared to IPv4. To our knowledge, Mint is the first tool to be able to study IPv6 censorship at scale.

7.5. Domain List Comparisons

We wanted to determine how much of each domain list was interfered with by ASes in our case study. We discovered that all of the ASes had the most interference with the OONI domains, followed by the Citizen Lab domains, and the Tranco domains. Since Citizen Lab curates their list for testing censorship, and OONI derives their domain list from Citizen Lab, it is not surprising that domains from OONI and Citizen Lab experience the most interference. This is in contrast to the Tranco list, which is simply a ranking of the most popular domains, without any censorship-focused selection.

8. Limitations

The primary limitation of our tool is that we are only able to measure network interference if the middlebox interfering with the traffic interferes with traffic bidirectionally, is stateless, and interferes with traffic via injections (blockpages, teardown packets, etc.). This means that we are not able to measure any networks that interfere with traffic by dropping packets or measure any networks that launch TLS MITM attacks by mandating users to accept government or ISP controlled certificates.

Additional limitations may arise if there is interference done in ways that Mint is not designed to account for:

Are we triggering transit censors? *Transit censorship* occurs when neither the source nor destination countries of a connection are responsible for censorship, but rather one of the intermediate countries through which the traffic transits. Transit censorship has been studied with respect to Russia, and China's DNS censorship before [33], [49].

While transit censorship has the potential to impact any censorship measurement technique whose traffic traverses multiple countries, it is of particular relevance to our tool because the middleboxes that deploy transit censorship are more likely to be middleboxes that are TCP noncompliant. This is because middleboxes that interfere with all traffic transiting their networks, whether the traffic originates from inside or outside of their network, may not observe all packets of a connection, and therefore may resort to TCP noncompliance when conducting interference. As such, a threat to the validity of our results is that the interference attributed to a specific network or country may actually be due to another network or country.

Are we triggering IDSes? Recall from §5.4 that we omitted "overblockers": networks that interfered with more than 95% of our test domains or with our control domains. We posited that this behavior could be caused by IDSes taking action against all packet sequences after a certain number of probes. If this were the case, then we would expect that the network would interfere with a contiguous sequence of probes—whereas if they had simply been applying a large blocklist, then we would expect the tampering to be randomly distributed throughout our sequence of probes.

We calculated the number of contiguous domains that we received injections from for each /24 and /48 that we scanned. There were 156,256 /24s for our HTTP/IPv4 scans and 160,080 for our HTTPS/IPv4 scans that interfered with 100 or more contiguous domains. Similarly, there were 27,422 /48s for our HTTP/IPv6 scans and 33,300 /48s for HTTPS/IPv6 scans that interfered with 100 or more contiguous domains. Compared to the number of /24 prefixes and /48 prefixes that we can trigger in total with Mint, as shown in Table 4, we believe that we are not triggering a significant amount of IDSes.

9. Conclusion

To best inform policy-makers and help guide censorship evasion efforts, it is important that network interference be measured *broadly* and *deeply*. Unfortunately, methods that rely on active endpoints within the countries are limited by availability, resources, and Internet penetration. In this paper, we introduce an alternative approach to global measurement of network interference. Our system, Mint, sends measurement packet sequences only to non-responsive IPs.

We evaluated Mint globally and through several case studies, showing that it is applicable in millions of networks and hundreds of countries. Of particular note, Mint is the first tool to permit global measurement of network interference over IPv6. The greatest challenge in IPv6 measurement—that it is difficult to find responsive end-hosts—turns out to be Mint's greatest benefit—it wants only non-responsive end-hosts.

We also demonstrated that Mint is able to measure thousands of ASes that popular tools cannot, and thus can be used to perform studies that were not possible before, such as assessing the centralization of network interference mechanism and policy (blocklists), and comparing IPv4 and IPv6 tampering between networks. That said, we view Mint as being purely complementary to existing approaches; it cannot measure in hundreds of ASes that other techniques can, and other approaches—especially those that operate inside-out—provide a ground truth perspective. We hope Mint will join other popular tools to help fill in the gaps where volunteers and responsive end-hosts cannot be found.

Looking forward, we anticipate other possible ways for researchers to apply Mint. Mint can also help researchers to measure specific networks that are difficult to measure with existing censorship measurement platforms, revealing changes in AS-specific censorship policy. Furthermore, we envision that Mint's ability to measure both IPv4 and IPv6 networks will help in censorship evasion—if Mint discovers that a domain is blocked over IPv4 but not IPv6, users can simply switch protocols to evade censorship.

To support future research, we make our code publicly available at https://censorship.ai

Acknowledgments

We thank the members of the Breakerspace lab for their feedback throughout the course of this project. We also thank the Censored Planet and OONI teams for providing contemporaneous data to ours so that we could perform our comparison evaluations. We are also grateful to the FOCI community for their feedback and support on an earlier version of this work [29]. Finally, we thank the anonymous IEEE S&P reviewers for their extensive and thoughtful feedback and suggestions. This work was supported in part by NSF grants CNS-1943240 and CNS-2323193 as well as an NSF Graduate Research Fellowship under Grant No. DGE2236417 and an Internet Society Pulse Fellowship.

References

[1] "Censored Planet Dashboard - Kuwait HTTP Measurement," 2024. [Online]. Available: https://archive.ph/6TaML

[2] "Censored Planet Dashboard - Kuwait HTTPS Measurement," 2024. [Online]. Available: https://archive.ph/OFbMN

[3] "OONI Measurement Aggregation Toolkit (MAT) - Kuwait AS6412," 2024. [Online]. Available: https://archive.ph/5YZzH

[4] A. Bhaskar and P. Pearce, "Many Roads Lead To Rome: How Packet Headers Influence DNS Censorship Measurement," in *USENIX Security Symposium*, 2022.

[5] ——, "Understanding Routing-Induced Censorship Changes Globally," in *ACM Conference on Computer and Communications Security (CCS)*, 2024.

[6] K. Bock, A. Alaraj, Y. Fax, K. Hurley, E. Wustrow, and D. Levin, "Weaponizing Middleboxes for TCP Reflected Amplification," in *USENIX Security Symposium*, 2021.

[7] K. Bock, P. Bharadwaj, J. Singh, and D. Levin, "Your Censor is My Censor: Weaponizing Censorship Infrastructure for Availability Attacks," in *USENIX Workshop on Offensive Technologies (WOOT)*, 2021.

[8] K. Bock, G. Hughey, L.-H. Merino, T. Arya, D. Liscinsky, R. Pogosian, and D. Levin, "Come as You Are: Helping Unmodified Clients Bypass Censorship with Server-side Evasion," in *ACM SIGCOMM*, 2020.

[9] K. Bock, G. Hughey, X. Qiang, and D. Levin, "Geneva: Evolving Censorship Evasion Strategies," in *ACM Conference on Computer and Communications Security (CCS)*, 2019.

[10] T. Chung, Y. Liu, D. Choffnes, D. Levin, B. M. Maggs, A. Mislove, and C. Wilson, "Measuring and Applying Invalid SSL Certificates: The Silent Majority," in *ACM Internet Measurement Conference (IMC)*, 2016.

[11] R. Clayton, S. J. Murdoch, and R. N. M. Watson, "Ignoring the Great Firewall of China," in *Privacy Enhancing Technologies Symposium (PETS)*, 2006.

[12] "DB-IP," https://db-ip.com/, 2024. [Online]. Available: https://db-ip.com/

[13] Z. Durumeric, E. Wustrow, and J. A. Halderman, "ZMap: Fast Internet-wide Scanning and its Security Applications," in *USENIX Security Symposium*, 2013.

[14] A. Filasto and J. Appelbaum, "OONI: Open Observatory of Network Interference," in *USENIX Workshop on Free and Open Communications on the Internet (FOCI)*, 2012.

[15] O. Gasser, Q. Scheitle, P. Foremski, Q. Lone, M. Korczyński, S. D. Strowes, L. Hendriks, and G. Carle, "Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists," in *ACM Internet Measurement Conference (IMC)*, 2018.

[16] O. Gasser, Q. Scheitle, S. Gebhard, and G. Carle, "Scanning the IPv6 Internet: Towards a Comprehensive Hitlist," in *Network Traffic Measurement and Analysis*, 2016.

[17] B. Haas, “Man in China sentenced to five years' jail for running VPN,” *The Guardian*, 2017, accessed 2018. [Online]. Available: https://www.theguardian.com/world/2017/dec/22/man-in-china-sentenced-to-five-years-jail-for-running-vpn

[18] N. P. Hoang, J. Dalek, M. Crete-Nishihata, N. Christin, V. Yegneswaran, M. Polychronakis, and N. Feamster, “GFWeb: Measuring the Great Firewall's Web Censorship at Scale,” in *USENIX Security Symposium*, 2024.

[19] N. P. Hoang, S. Doreen, and M. Polychronakis, “Measuring I2P Censorship at a Global Scale,” in *USENIX Workshop on Free and Open Communications on the Internet (FOCI)*, 2019.

[20] N. P. Hoang, M. Polychronakis, and P. Gill, “Measuring the Accessibility of Domain Name Encryption and its Impact on Internet Filtering,” in *Passive and Active Network Measurement Conference (PAM)*, 2022.

[21] “IP2Location,” https://www.ip2location.com/, 2024. [Online]. Available: https://www.ip2location.com/

[22] L. Jin, S. Hao, H. Wang, and C. Cotton, “Understanding the Practices of Global Censorship through Accurate, End-to-End Measurements,” in *ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)*, 2021.

[23] B. Jones, N. Feamster, and P. Gill, “Automated Detection and Fingerprinting of Censorship Block Pages,” in *ACM Internet Measurement Conference (IMC)*, 2014.

[24] Y. Kabyshev, R. Daiyrbekov, V. Melyakov, I. Loskutov, M. Xynou, E. Yachmeneva, A. Filastò, and M. Gulati, “Kazakhstan: TLS MITM attacks and blocking of news media, human rights, and circumvention tool sites,” 2024. [Online]. Available: https://ooni.org/post/2024-kazakhstan-report

[25] F. Li and D. Freeman, “Towards A User-Level Understanding of IPv6 Behavior,” in *ACM Internet Measurement Conference (IMC)*, 2020.

[26] MaxMind, “GeoLite2,” https://dev.maxmind.com/geoip/geoip2/geolite2, 2020.

[27] T. Narten, R. Draves, and S. Krishnan, “Privacy Extensions for Stateless Address Autoconfiguration in IPv6,” RFC 4941, Sep. 2007. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4941.txt

[28] A. A. Niaki, S. Cho, Z. Weinberg, N. P. Hoang, A. Razaghpanah, N. Christin, and P. Gill, “ICLab: A Global, Longitudinal Internet Censorship Measurement Platform,” in *IEEE Symposium on Security and Privacy*, 2020.

[29] S. Nourin, K. Bock, N. P. Hoang, and D. Levin, “Detecting Network Interference Without Endpoint Participation,” in *USENIX Workshop on Free and Open Communications on the Internet (FOCI)*, 2023.

[30] S. Nourin, V. Tran, X. Jiang, K. Bock, N. Feamster, N. P. Hoang, and D. Levin, “Measuring and Evading Turkmenistan's Internet Censorship,” in *International World Wide Web Conference (WWW)*, 2023.

[31] “The NTP Pool Project,” https://www.ntppool.org/en/, 2024. [Online]. Available: https://www.ntppool.org/en/

[32] “OpenNet Initiative,” https://opennet.net, 2014.

[33] A. Ortwein, K. Bock, and D. Levin, “Towards a Comprehensive Understanding of Russian Transit Censorship,” in *USENIX Workshop on Free and Open Communications on the Internet (FOCI)*, 2023.

[34] P. Pearce, R. Ensafi, F. Li, N. Feamster, and V. Paxson, “Augur: Internet-Wide Detection of Connectivity Disruptions,” in *IEEE Symposium on Security and Privacy*, 2017.

[35] P. Pearce, B. Jones, F. Li, R. Ensafi, N. Feamster, N. Weaver, and V. Paxson, “Global Measurement of DNS Manipulation,” in *USENIX Security Symposium*, 2017.

[36] “PeeringDB,” https://www.peeringdb.com/, 2025. [Online]. Available: https://www.peeringdb.com/

[37] R. S. Raman, L. Evdokimov, E. Wustrow, J. A. Halderman, and R. Ensafi, “Investigating Large Scale HTTPS Interception in Kazakhstan,” in *ACM Internet Measurement Conference (IMC)*, 2020.

[38] R. S. Raman, L.-H. Merino, K. Bock, M. Fayed, D. Levin, N. Sullivan, and L. Valenta, “Global, Passive Detection of Connection Tampering,” in *ACM SIGCOMM*, 2023.

[39] R. S. Raman, P. Shenoy, K. Kohls, and R. Ensafi, “Censored Planet: An Internet-wide, Longitudinal Censorship Observatory,” in *ACM Conference on Computer and Communications Security (CCS)*, 2020.

[40] R. S. Raman, A. Stoll, J. Dalek, R. Ramesh, W. Scott, and R. Ensafi, “Measuring the Deployment of Network Censorship Filters at Global Scale,” in *Network and Distributed System Security Symposium (NDSS)*, 2020.

[41] R. S. Raman, M. Wang, J. Dalek, J. Mayer, and R. Ensaf, “Network measurement methods for locating and examining censorship devices,” in *ACM Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, 2022.

[42] R. S. Raman, M. Wang, J. Dalek, J. Mayer, and R. Ensafi, “Network Measurement Methods for Locating and Examining Censorship Devices,” in *ACM Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, 2022.

[43] R. Ramesh, R. S. Raman, M. Bernhard, V. Ongkowijaya, L. Evdokimov, A. Edmundson, S. J. Sprecher, M. Ikram, and R. Ensafi, “Decentralized control: A case study of russia,” in *NDSS*, 2020.

[44] Reporters Without Borders, “2024 World Press Freedom Index : journalism under political pressure,” https://rsf.org/en/2024-world-press-freedom-index-journalism-under-political-pressure, 2024. [Online]. Available: https://rsf.org/en/2024-world-press-freedom-index-journalism-under-political-pressure

[45] RFE/RL'S Turkmen Service, “VPNs Are Not A-OK: Turkmen Internet Users Forced To Swear On Koran They Won't Use Them,” https://www.rferl.org/a/turkmenistan-vpn-koran-ban/31402718.html, 2021-08-10.

[46] RIPE, “Best Current Operational Practice for Operators: IPv6 Prefix Assignment for End-Users - Persistent vs Non-Persistent, and What Size to Choose,” 2017, https://www.ripe.net/publications/docs/ripe-690.

[47] E. Rye and D. Levin, “IPv6 Hitlists at Scale: Be Careful What You Wish For,” in *ACM SIGCOMM*, 2023.

[48] W. Scott, T. Anderson, T. Kohno, and A. Krishnamurthy, “Satellite: Joint Analysis of CDNs and Network-Level Interference,” in *USENIX Annual Technical Conference*, 2016.

[49] Sparks, Neo, Tank, Smith, and Dozer, “The Collateral Damage of Internet Censorship by DNS Injection,” in *ACM SIGCOMM*, 2012.

[50] “TeamCymru IP to ASN Mapping Service,” https://www.team-cymru.com/ip-asn-mapping, 2024. [Online]. Available: https://www.team-cymru.com/ip-asn-mapping

[51] S. Thomson, T. Narten, and T. Jinmei, “IPv6 Stateless Address Autoconfiguration,” RFC 4862, Internet Engineering Task Force, Sep. 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4862.txt

[52] tumi8, “ZMapv6: Internet Scanner with IPv6 Capabilities,” 2024, https://github.com/tumi8/zmap.

[53] B. VanderSloot, A. McDonald, W. Scott, J. A. Halderman, and R. Ensafi, “Quack: Scalable Remote Measurement of Application-Layer Censorship,” in *USENIX Security Symposium*, 2018.

[54] “VirusTotal: URL Scanning Service,” https://www.virustotal.com/, 2024. [Online]. Available: https://www.virustotal.com/

[55] Z. Weinberg, D. Barradas, and N. Christin, “Chinese Wall or Swiss Cheese? Keyword filtering in the Great Firewall of China,” in *International World Wide Web Conference (WWW)*, 2021.

[56] G. Williams, M. Erdemir, A. Hsu, S. Bhat, A. Bhaskar, F. Li, and P. Pearce, “6Sense: Internet-Wide IPv6 Scanning and its Security Applications,” in *USENIX Security Symposium*, 2024.

[57] X. Xu, M. Mao, and J. A. Halderman, “Internet Censorship in China: Where Does the Filtering Occur?” in *Passive and Active Network Measurement Conference (PAM)*, 2011.

[58] “ZGrab2,” https://github.com/zmap/zgrab2, 2019. [Online]. Available: https://github.com/zmap/zgrab2

Appendix A.
Supplementary Plots and Tables

This appendix presents supplementary plots and tables that echo the core contributions of our paper, with either added detail or applied to IPv6 instead of IPv4.

Recall from §4.1 that we omitted networks from scanning and analysis unless they had non-responsive hosts in all six of our packet sequence scans for both protocols. Table 13 shows the top five organizations that were omitted, for HTTP and HTTPS, respectively. As mentioned earlier, these are mostly telecom and cloud providers.

Figure 11 shows the map of which countries Mint can trigger for HTTPS (IPv4 and IPv6), and Table 14 shows the top five countries with the highest fraction of triggerable IP addresses in IPv4 and IPv6 over HTTPS.

Tables 15 and 16 present additional results showing which packet sequences trigger HTTPS via IPv4 and HTTP(S) over IPv6.

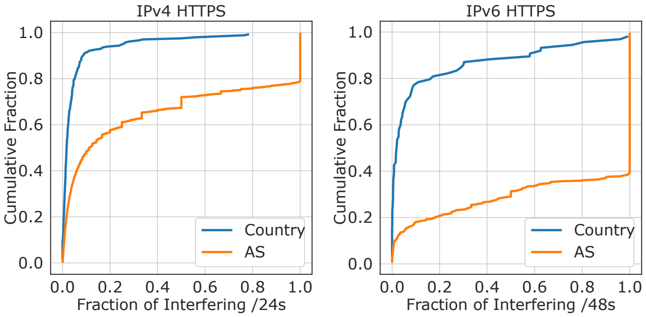


Figure 6: CDF of the fraction of interfering /24s and /48s in IPv4 and IPv6 over HTTPS

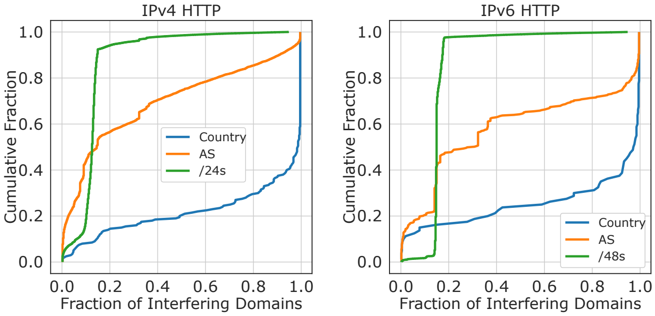


Figure 7: CDF of the fraction of interfering domains in IPv4 and IPv6 in HTTP

Figure 6 shows the variability in the fraction of interfered IP addresses for HTTPS, while Figure 7 and 8 show this for the fraction of domains over HTTP(S).

Figure 9 presents the distribution of the CoV of packet sequences over IPv6 while Figure 10 shows the distribution of the CoV of the domains for IPv6. Turkmenistan is excluded from both IPv6 figures as it does not have any allocated IPv6 prefixes.

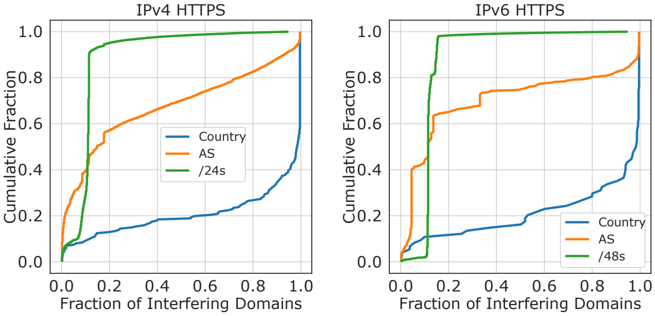


Figure 8: CDF of the fraction of interfering domains in IPv4 and IPv6 in HTTPS

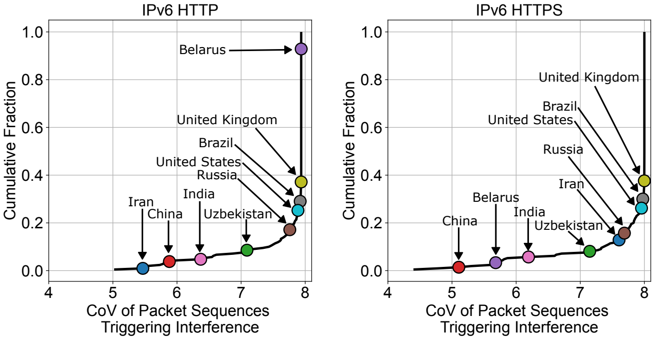


Figure 9: CDF of the CoV of packet sequences triggering interference for IPv6 over HTTP(S) with selected countries annotated

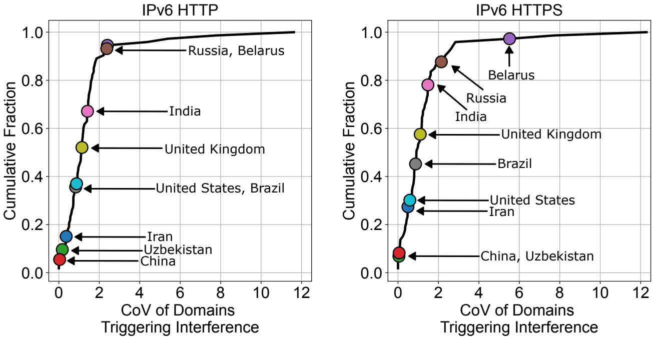
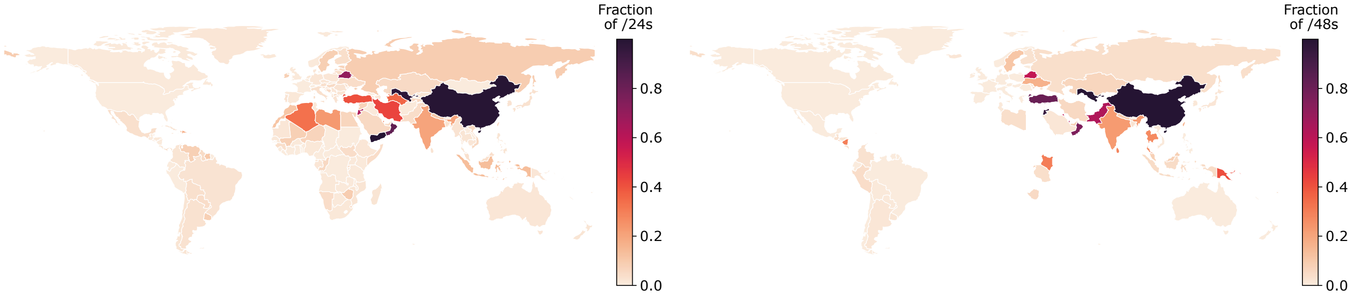


Figure 10: CDF of the CoV of domains triggering interference for IPv6 over HTTP(S) with selected countries annotated

IPv4 HTTP Scans				IPv4 HTTPS Scans			
# /24s	**ASN**	**Organization**	**Country**	**# /24s**	**ASN**	**Organization**	**Country**
91,293	4766	KIXS-AS-KR Korea Telecom	KR	117,493	4766	KIXS-AS-KR Korea Telecom	KR
88,132	16509	AMAZON-02	US	97,580	16509	AMAZON-02	US
38,106	14618	AMAZON-AES	US	41,219	14618	AMAZON-AES	US
37,089	7922	COMCAST-7922	US	39,564	9318	SKB-AS SK Broadband Co Ltd	KR
26,924	9318	SKB-AS SK Broadband Co Ltd	KR	36,760	7922	COMCAST-7922	US

TABLE 13: Top 5 ASNs and Organizations by Number of /24s Not Covered by IPv4 HTTP(S) Packet Sequence Scans



(a) HTTPS IPv4 Network Interference

(b) HTTPS IPv6 Network Interference

Figure 11: Map of where Mint can trigger HTTPS network interference over IPv4 and IPv6.

IPv4 HTTPS Network Interference				IPv6 HTTPS Network Interference			
Country	**# of Responding /24s**	**# of /24s Measured**	**Ratio**	**Country**	**# of Responding /48s**	**# of /48s Measured**	**Ratio**
Yemen	307	389	0.789	China	820,386	820,463	0.999
China	1,040,397	1,325,745	0.785	Uzbekistan	1,711	1,723	0.993
Uzbekistan	814	1,063	0.766	Jordan	2,923	3,046	0.960
Oman	2,467	3,745	0.659	Turkey	7,028	8,712	0.807
Belarus	3,618	6,490	0.557	Oman	60	66	0.758

TABLE 14: Top 5 Countries with Highest Fraction of IPs that Responded to Probes over HTTPS for IPv4 and IPv6

	Triggered			Uniquely Triggered		
Packet Sequence	**Countries**	**ASes**	**/24s**	**Countries**	**ASes**	**/24s**
⟨SYN; PSH+ACK⟩	181	4,774	1,123,020	6	945	63,878
⟨PSH⟩	97	1,376	32,060	0	81	3,720
⟨SYN; PSH⟩	173	3,918	1,081,332	5	552	32,053
⟨SYN⟩	161	2,684	66,306	1	540	22,977
⟨PSH+ACK;sleep;PSH+ACK⟩	175	4,372	114,140	3	765	29,093
⟨PSH+ACK⟩	173	4,166	91,646	1	657	23,291

TABLE 15: Number of Countries, ASes, and /24s Triggered over HTTPS for IPv4

	IPv6 HTTP Scans						IPv6 HTTPS Scans					
	Triggered			Uniquely Triggered			Triggered			Uniquely Triggered		
Packet Sequence	**Countries**	**ASes**	**/48s**	**Countries**	**ASes**	**/48s**	**Countries**	**ASes**	**/48s**	**Countries**	**ASes**	**/48s**
⟨SYN; PSH+ACK⟩	78	362	853,954	3	24	11,650	77	511	853,703	1	14	10,878
⟨PSH⟩	63	190	245,433	0	0	130	60	345	241,648	0	0	110
⟨SYN; PSH⟩	72	331	844,152	0	6	1,334	76	489	842,531	0	5	604
⟨SYN⟩	68	171	162,195	1	2	224	65	321	91,827	0	8	238
⟨PSH+ACK;sleep;PSH+ACK⟩	70	261	266,087	0	3	1,261	75	419	265,894	1	4	2,018
⟨PSH+ACK⟩	70	245	248,458	1	2	184	69	395	270,071	0	0	183

TABLE 16: Number of Countries, ASes, and /48s Triggered over HTTP and HTTPS for IPv6

Appendix B. Meta-Review

The following meta-review was prepared by the program committee for the 2025 IEEE Symposium on Security and Privacy (S&P) as part of the review process as detailed in the call for papers.

B.1. Summary

In this paper the authors evaluate a recently-proposed censorship measurement method in which a middlebox is confused into thinking an active connection exists with censored content, and its response (sending a block-page or resetting the link) can be measured. The paper shows that a generalization of this approach is able to perform Internet-wide active measurements (both for IPv4 and IPv6), and also apply it to investigate three case studies.

B.2. Scientific Contributions

- Independent Confirmation of Important Results with Limited Prior Research
- Creates a New Tool to Enable Future Science
- Provides a Valuable Step Forward in an Established Field

B.3. Reasons for Acceptance

1) The paper provides Independent Confirmation of Important Results with Limited Prior Research. While the middlebox method was previously evaluated in limited settings in previous works, this work is the first to apply it on an Internet-wide scale, both in IPv4 and in IPv6 settings.
2) The paper Creates a New Tool to Enable Future Science, and Provides a Valuable Step Forward in an Established Field. The generalized middlebox approach will be useful in future measurements of censorship, both for longitudinal and for in-depth studies.

B.4. Noteworthy Concerns

1) The reviewers had some concerns about the limited novelty of the bidirectional interference methodology, which was previously studied in two other works. In spite of these concerns, the scale of the current work, as well as the case studies and the lessons learned from them, still make the work worthy of consideration.