Bamboozling Certificate Authorities with BGP

Henry Birge-Lee
Princeton University

Yixin Sun
Princeton University

Anne Edmundson
Princeton University

Jennifer Rexford
Princeton University

Prateek Mittal
Princeton University

Abstract

The Public Key Infrastructure (PKI) protects users from malicious man-in-the-middle attacks by having trusted Certificate Authorities (CAs) vouch for the domain names of servers on the Internet through digitally signed certificates. Ironically, the mechanism CAs use to issue certificates is itself vulnerable to man-in-the-middle attacks by network-level adversaries. Autonomous Systems (ASes) can exploit vulnerabilities in the Border Gateway Protocol (BGP) to hijack traffic destined to a victim's domain. In this paper, we rigorously analyze attacks that an adversary can use to obtain a bogus certificate. We perform the first real-world demonstration of BGP attacks to obtain bogus certificates from top CAs in an ethical manner. To assess the vulnerability of the PKI, we collect a dataset of 1.8 million certificates and find that an adversary would be capable of gaining a bogus certificate for the vast majority of domains. Finally, we propose and evaluate two countermeasures to secure the PKI: 1) CAs verifying domains from multiple vantage points to make it harder to launch a successful attack, and 2) a BGP monitoring system for CAs to detect suspicious BGP routes and delay certificate issuance to give network operators time to react to BGP attacks.

1 Introduction

Digital certificates serve as the foundation of trust in encrypted communication. When a Certificate Authority (CA) is asked to sign a certificate, the CA must establish that the client requesting the certificate is the legitimate owner of the domain name in question. An adversary that obtains a trusted certificate can pose as the victim's domain and intercept/modify sensitive HTTPS traffic like bank logins and credit card information [24]. The mechanism used by CAs to verify domain ownership, known as *domain control verification*, is critical to preventing adversaries from obtaining trusted certificates for domains they do not control. Domain control verification is performed through a standardized set of methods including http-based and email-based verification [18].

Recently, researchers have exposed several flaws in existing domain control verification mechanisms. WoSign was found issuing certificates to users that could demonstrate control of *any* TCP port at a domain (including those above 50,000) as opposed to strictly requiring control of traditional mail, HTTP, and TLS ports [3]. In addition, researchers have found instances of CAs sending domain control verification requests to email addresses that belong to ordinary users at a domain as opposed to bona fide administrators [1]. In response, countermeasures are being developed such as standardizing which URLs on a domain's web server can serve to verify control of that domain [11].

While these advances can defend against some attacks, none of them help to secure domain control verification against *network-level* adversaries, i.e., Autonomous System (AS), that can manipulate the Border Gateway Protocol (BGP). Such adversaries can launch active BGP hijack and interception attacks to steal traffic away from victims or CAs, and spoof the domain control verification process to obtain bogus certificates.

In this paper, we first analyze and compare BGP attacks on the domain verification process to develop a taxonomy and present a highly effective use of the "AS-path poisoning" attack originally performed in [39]. Next, we launch all the BGP attacks against our own domain and decrypt seemingly "secure" HTTPS traffic within seconds. To avoid harming real users, these attacks were done in an ethical manner on domains that resolve into our own IP prefix and were registered solely for the purpose of the experiments. We then quantify the vulnerability of domain verification to these attacks. Finally, we propose countermeasures against these attacks. Our main contributions are as follows:

Active BGP Attacks on Domain Verification Pro-

cess: We performed five types of real-world BGP attacks (against a domain we owned running on an IP prefix we controlled) during the domain verification process: 1) a traditional BGP sub-prefix attack, 2) a traditional BGP equally-specific-prefix attack (like the attack theorized in [22]), 3) a prepended BGP sub-prefix attack, 4) a prepended BGP equally-specific-prefix attack, and 5) a BGP AS-path poisoning attack (see section 2.2 for details about these attacks).

We are the first to demonstrate the use of the prepended and AS-path poisoning attacks on the PKI, and the first to perform any of these attacks during the domain verification process in the wild. We successfully obtained bogus certificates from all of the top five CAs (Let's Encrypt, GoDaddy, Comodo, Symantec, GlobalSign) [8] in our real-world attacks. Our results were a major factor in Let's Encrypt's decision to start deploying the multiple-vantage-point countermeasure [37].

Quantify vulnerability of domains: We collected a dataset of 1.8 million certificates from Google's Certificate Transparency project logs [32] and studied the domains requesting those certificates. By observing the number of domains run out of IP prefixes shorter than 24 bits long (/24), we found that 72% of the domains were vulnerable to BGP sub-prefix hijack attacks and BGP AS-path poisoning attacks, which could allow *any* AS to get a certificate for these domains. Furthermore, the domains were vulnerable to BGP equally-specific-prefix attacks from an average of 70% of ASes.

Countermeasures against BGP attacks: We proposed and developed two countermeasures to mitigate the threat of BGP attacks: multiple vantage point verification and a live BGP monitoring system.

- **Multiple Vantage Point Verification:** We propose to perform domain control verification from multiple locations on the Internet (vantage points) to prevent localized BGP attacks. We calculate the best locations for vantage points and quantify the resulting security benefit.
- **Live BGP Monitoring System:** We design and implement (in the Let's Encrypt's CA) a monitoring system with a novel route age heuristic to prevent short-lived BGP attacks [19] that can quickly lead to a bogus certificate before the attack is noticed. Our heuristic is designed for CAs and forces adversaries to keep attacks active for several hours, giving network operators time to react.

Some of the BGP attacks were briefly discussed in a short abstract [16]. In this paper, we go further by analyzing the complete attack surface of BGP attacks on PKI and performing all the attacks in the wild — with success. We also measure the vulnerability of the current PKI to these attacks, and propose/evaluate two effective countermeasures to defend against the attacks.

2 BGP Attacks on the PKI

The Public Key Infrastructure (PKI) requires that all certificates be signed by a trusted certificate authority (CA). Browsers and any other TLS clients maintain lists of publicly trusted CAs. 135 organizations were recognized as commercial CAs (other CAs, such as the government of France, will not accept certificate signing requests from the general public) [20]. Any CA is capable of signing a certificate for any domain.

Domain Control Verification. In order to verify that an applicant requesting a certificate has control of the domain in question, the CA must perform domain control verification through a set of methods. Each method bootstraps trust by forcing a user to demonstrate control of an important network resource (e.g., a website or email address) associated with the domain. Figure 1 illustrates the domain control verification process with HTTP verification, which requires the user to make an agreed upon change to the root directory of the website running at the domain. Another commonly used method is email verification, by which an email is sent to an administrator's email address at the domain, requiring the administrator to visit a randomly generated URL before continuing. Other methods include DNS TXT verification or methods that do not rely on communication via the Internet (e.g., official letters of authorization).

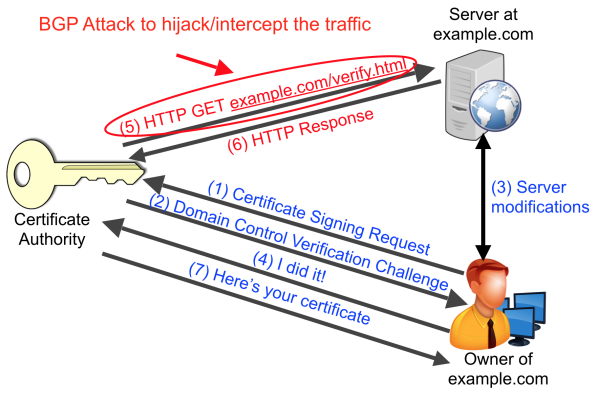


Figure 1: HTTP domain control verification.

BGP Attacks on Domain Control Verification. The domain control verification process creates a vulnerability to network-level adversaries who can fake control of the network resources in step (5) and (6) in Figure 1. An adversary can send a certificate signing request for a victim's domain to a CA. When the CA verifies the network resources via an HTTP GET request in step (5), the adversary can use BGP attacks to hijack/intercept the traffic to the victim's domain such that the CA's request will be routed to the adversary instead. The adversary can then answer the CA's HTTP request in step (6) and present the

document required for domain control verification.

Our key contribution in this section is to explore the broad BGP attack surface that can be used to obtain a bogus TLS certificate in the above process. We first develop an adversary model, and then explore five types of BGP attacks. In particular, we propose and analyze an advanced and stealthy AS-path poisoning attack, that can target **any** trusted CA that is not on the route between the adversary and the victim. We present an in depth analysis of how the intricacies of these BGP attacks affect the current PKI.

2.1 Adversary Model

Adversary Objectives: We consider an adversary that aims to obtain a bogus certificate for a victim's domain and then decrypt sensitive TLS traffic for as long as possible without being detected. Thus, the slower a defense system detects a BGP attack, the more effective the man-in-the-middle attack is.

Because intercepting a TLS stream can cause significant damage in a couple of hours [24], detection systems that require manual investigation to confirm that an attack has occurred or systems that have a significant delay before detection is possible are not effective at preventing these attacks . However, the adversary is incentivized to avoid major reachability problems (that will cause a service interruption alerting the victim to the attack) and highly suspicious BGP announcements that might get automatically filtered or immediately trigger alerts. Given this adversary model, we aim to assess the current degree of vulnerability of the PKI.

Realistic Constraints on Adversary Capabilities: An adversary must compromise an AS's border router or control an AS to launch the attack. Assuming the adversarial AS and victim's domain to be fixed, several variables are beyond the control of the adversary. The topological relationship between the adversary, the victim, and the CA, and the benign BGP announcement for the IP prefix that includes the victim's domain are considered beyond the control of adversary.

Despite these constraints, we assume adversaries can control exactly what BGP announcement they make and which neighboring ASes they make this announcement to. We also assume an adversary is capable of generating traffic with a source IP address that belongs to the victim. Studies show that a significant portion of ASes still allows source IP spoofing [2, 34] due to a lack of ingress filtering. Even a strictly filtered adversary can spoof packets by gaining control of a client in one of these networks that allow spoofing and use it to spoof packets on behalf of the adversary.

Another variable the adversary can control is which IP address to attack. The adversary can directly target the IP address of the victim's domain, or the IP address of any DNS server involved in resolving the victim's domain to give a bogus DNS response to the CA. This will cause the CA to request the verification webpage from the adversary as opposed to the victim.

In addition, it is possible for the adversary to attack a CA's IP address. The adversary can intercept the response of the victim (or a DNS server used to resolve the victim's IP) to the CA, modify it to contain the document specified by the CA (or an incorrect DNS response), and forward it to the CA. By man-in-the-middling the responses from the victim's domain or DNS servers, the adversary can fool the domain control verification process. These additional IP addresses an adversary can attack increase the attack surface.

BGP Attack Properties: For an attack to be effective, it must have two properties: viability and stealthiness. For a given adversary, victim, and BGP attack type, viability is a binary indication of whether the adversary is capable of launching the attack. On the other hand, the stealthiness of an attack is determined by several properties that we group into two categories:

1. Control-plane stealthiness: this is measured through the properties of a BGP announcement like the IP prefix announced and the AS path.
2. Data-plane stealthiness: this is measured through the number of ASes whose connectivity to a victim's domain is disrupted during an attack.

2.2 Taxonomy of BGP Attacks

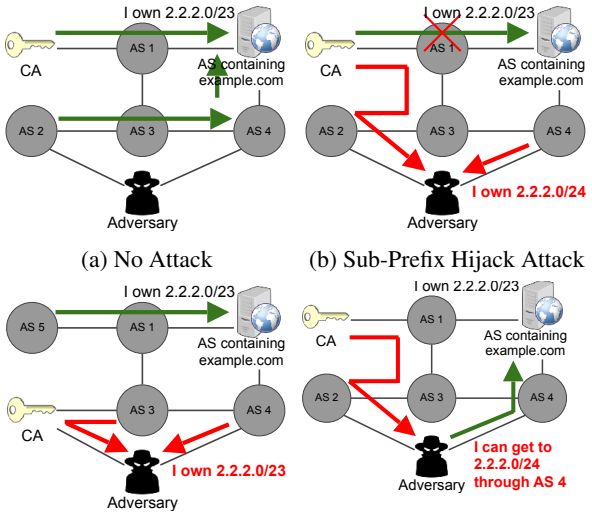
We present the details of the following five attacks, and discuss the tradeoff between attack stealthiness and viability for each attack:

- **Traditional sub-prefix attack:** An adversary makes a BGP announcement originating a more-specific IP prefix than the victim's prefix.
- **Traditional equally-specific-prefix attack:** An adversary announces an equal-length prefix as the victim's prefix.
- **Prepended sub-prefix attack:** An adversary claims reachability to a more-specific IP prefix via a non-existent connection to the victim.
- **Prepended equally-specific-prefix attack:** An adversary claims reachability to the victim's prefix via a non-existent connection.
- **AS-path poisoning attack:** An adversary announces a valid route to a more-specific prefix than the victim's prefix to intercept Internet traffic en route to the victim.

Figure 2 illustrates the effects of these BGP attacks on Internet routing, and we summarize the unique properties and implementation details of these BGP attacks in

Attack Name	Prefix Length Announced	AS-Path Effect	Effect on Victim
Traditional Sub-Prefix Hijack	Sub-Prefix	Entire Path Differs	Global Traffic Blackholed
Traditional Equally-Specific Prefix Hijack	Equal-Length	Entire Path Differs	Selective Traffic Blackholed
Prepended Sub-Prefix Hijack	Sub-Prefix	ASes After Origin Differ	Global Traffic Blackholed
Prepended Equally-Specific Prefix Hijack	Equal-Length	ASes After Origin Differ	Selective Traffic Blackholed
AS-Path Poisoning Attack	Sub-Prefix	Valid Route to Victim	Global Traffic Intercepted

Table 1: BGP attacks and their associated properties.



(a) No Attack (b) Sub-Prefix Hijack Attack

(c) Equally-Specific Hijack (d) AS-Path Poisoning Attack

Figure 2: Attack illustration.

Table 1. At a high level, each attack in the lower table is more preferable to an adversary because it is stealthier and less detectable by existing BGP security measures and data-plane measurements. However, these stealthier attacks are less likely to be viable for a given adversary. The viability and stealthiness of each attack is shown in Table 2. We later use these observations to asses the vulnerability of the PKI to BGP attacks of varying levels of stealthiness in Section 4.

2.2.1 Traditional Sub-Prefix Hijack

Attack Methodology: The adversary makes a BGP announcement to a sub-prefix that includes the victim domain's IP. For example, to attack a victim domain on the IP address X.Y.Z.1 of prefix X.Y.Z.0/23, an adversary could launch a sub-prefix attack announcing the prefix X.Y.Z.0/24 to capture the victim's traffic. Figure 2a shows the default routing of traffic when no attack is active, and Figure 2b shows the effects of a sub-prefix hijack attack. Because routers prefer more-specific IP prefixes over less-specific ones, this announcement will capture all traffic to the victim's domain, as demonstrated in Figure 2b. This attack is highly effective and can be launched by any AS on the Internet.

Attack Viability: This attack is highly viable. The majority of domains use IP prefixes shorter than the maximum /24 (shown in Section 4.2), which allows an attacker to announce IP sub-prefixes without being filtered (many ASes filter announcements longer than /24 [9]). Additionally, the attack has a global effect and the adversary's location does not influence the attack viability.

Attack Stealthiness: Although effective, this attack is very visible in both the control and data planes. As seen in Figure 2b, *all* traffic from any AS on the Internet is routed to the adversary. In the data plane, this causes a nearly global loss of connectivity to the victim's domain. In addition, from a control-plane viewpoint, the announcement is highly suspicious. The adversary's AS has likely never announced the victim's prefix before. When the adversary originates the victim's prefix (an event known as a Multiple Origin AS, MOAS, conflict [49]), many BGP monitoring systems [30, 42, 29, 26] will flag this announcement because of the suspicious change in origin AS. Furthermore, if the victim has an RPKI entry for their IP prefix, this announcement will be filtered by ASes that perform Route Origin Validation (ROV) [17]. Thus, although an adversary could easily get a certificate before the attack is detected (as we will show in Section 3, several CAs will sign a certificate seconds after domain control verification and these attacks can last for several hours), the rapid detection of this announcement would reduce the damage the bogus certificate could do.

2.2.2 Traditional Equally-Specific-Prefix Hijack

Attack Methodology: An adversary aiming to increase stealthiness (or attack a domain running in a /24 prefix so a sub-prefix attack is not viable) may launch an equally-specific-prefix hijack [22]. In this attack, an adversary announces the exact same prefix that the victim is announcing. Each AS will then pick the preferred route

Attack Name	Effective Against /24 Prefixes	Evades Origin Change Detection	Internet Topology Location Required
Traditional Sub-Prefix Hijack	No	No	Any location
Traditional Equally-Specific Prefix Hijack	Yes	No	Many locations
Prepended Sub-Prefix Hijack	No	Yes	Any location
Prepended Equally-Specific Prefix Hijack	Yes	Yes	Few locations
AS-Path Poisoning Attack	No	Yes	Any multi-homed location

Table 2: The stealthiness and viability of BGP attacks.

between the adversary's false announcement and the victim's original announcement, based on local preferences and path length, etc.. As shown in Figure 2c, this type of attack causes only part of the Internet to prefer the adversary's announcement. In parts of the Internet that do not prefer the adversary's route, this attack is unnoticeable in the data plane (connectivity is unaffected). Also, in the control plane, many ASes will not learn (let alone choose) the adversary's route.

Attack Viability: The viability of this attack is determined by the topological relationship between the CA, the victim, and the adversary. The Internet topology must cause the adversary's route to be preferred by the CA over the victim's route. Thus, this attack is less viable than a traditional sub-prefix hijack. We will further quantify the viability of this attack in Section 4.3.1.

Attack Stealthiness: In the control plane, this attack is more stealthy than a traditional sub-prefix hijack because parts of the Internet will not hear the adversary's announcement. However, this attack still involves a change in origin AS that can be detected by RPKI and BGP monitoring systems. In the data plane, this attack will not cause a global loss of connectivity to the victim's domain like the traditional sub-prefix hijack.

2.2.3 Prepended Sub-Prefix Hijack

Attack Methodology: An adversary can increase the stealthiness of a sub-prefix hijack attack by prepending the victim's Autonomous System Number (ASN) in the malicious announcement's AS path. Thus, the AS path will begin with the victim's ASN followed by the adversary's ASN. Importantly, the adversary's AS is no longer claiming to be the origin AS for the prefix. Instead the adversary is simply claiming a topological connection to the victim (that does not in fact exist).

Attack Viability: The viability of this attack is identical to that of the traditional sub-prefix hijack attack because routers always prefer a more specific BGP announcement over a less-specific one regardless of the AS-path field. Thus, all victims that have an IP prefix shorter than /24 are vulnerable.

Attack Stealthiness: This attack is significantly more stealthy than a traditional sub-prefix hijack, particularly in the control plane. The origin ASN in the adversary's announcement is identical to the victim's ASN in the original announcement. BGP monitoring systems that only perform origin AS check will not be able to detect this attack. More advanced techniques such as data-plane measurements [42, 26] are needed to detect the attack. However, these advanced systems often require human intervention to take action on a flagged route, which may take hours [9].

On the data plane, this attack has a similar global effect to traditional sub-prefix attack. However, due to control-plane stealthiness, an adversary will likely launch this attack (instead of a traditional sub-prefix hijack attack) to increase stealthiness with no effect on viability.

2.2.4 Prepended Equally-Specific-Prefix Hijack

Attack Methodology: Similar to the prepended sub-prefix attack, an adversary can prepend the victim's ASN to an equally-specific-prefix hijack. Because the adversary is now announcing the same prefix as the victim with the same origin ASN, this attack is has a significant increase in stealthiness over all previously listed attacks.

Attack Viability: This attack is even less viable than a traditional equally-specific prefix hijack. AS-path length is an important factor in route selection. Because the adversary's route is made one hop longer by prepending the victim's ASN, the adversary's announcement will attract less traffic than it does in the traditional equally-specific prefix hijack. In many other applications, this can significantly limit the use of such an attack, but when attacking the PKI, the adversary only needs to intercept traffic from one of many trusted CAs. Thus, this attack can still be viable even with the reduced area of effect.

Attack Stealthiness: This attack has similar control plane properties to the prepended sub-prefix hijack. The prepended victim origin AS makes the attack less likely to be detected by BGP monitoring systems. Thus, the attack is very stealthy. On the data plane, it is similar to the traditional equally-specific prefix hijack which does

not cause global loss of connectivity.

2.2.5 Sub-Prefix-Interception With Path Poisoning

Attack Methodology: While all previous attacks have involved breaking data-plane connectivity to a victim's domain (either global or partial), we here present an attack that uses AS-path poisoning to maintain a valid route to the victim's domain. Our attack allows an adversary to fully man-in-the-middle encrypted TLS traffic (as opposed to only attacking unencrypted traffic [39]). In our attack, an adversary announces a sub-prefix of the victim's original announcement similar to the sub-prefix hijack attack. The crucial difference is that the adversary will append a legitimate route R to the victim following the adversary's own ASN in the announced path. *This causes the ASes along route R between the adversary and the victim to ignore the adversary's announcement because of loop prevention.* These ASes would still prefer the victim's original announcement, and thus route R is still a valid route to the victim. All of the ASes not on route R would prefer the adversary's announcement because of the adversary's more-specific prefix announcement. Thus, the entire Internet (with the exception of the ASes on route R) routes traffic destined to the victim's domain to the adversary, and the adversary can still forward all the traffic through to the victim via a valid route without breaking data-plane connectivity.

Attack Viability: This attack can be performed by any multi-homed AS against a domain on a prefix shorter than /24. It is crucial that the adversary's AS be multi-homed (have more than one provider) so at least one provider can deliver the victim's traffic to the adversary while another provider forwards the traffic to the victim.

Attack Stealthiness: This attack is completely stealthy in the data plane in terms of connectivity. Once the adversary makes the announcement, it can continue forwarding traffic to the victim via the valid route to maintain data connectivity. In addition, the adversary can use the bogus certificate gained in this attack to not only fake a victim's website but to fully man-in-the-middle all TLS connections. The adversary can decrypt TLS traffic by posing as the victim's domain to users. It can then forward the user traffic to the victim's domain to hide the attack. This ensures that there is no connectivity issue from the victim's perspective while a full man-in-the-middle attack is under way on TLS connections.

This attack also has a high degree of stealthiness in the control plane. Many networks will announce sub-prefixes on occasion for traffic engineering. Because the adversary's announcement has the victim as the origin AS of the prefix and a valid path to the victim, this announcement will look similar to a legitimate route. In addition, because of BGP loop prevention, the ASes along route R may never notice this malicious announcement.

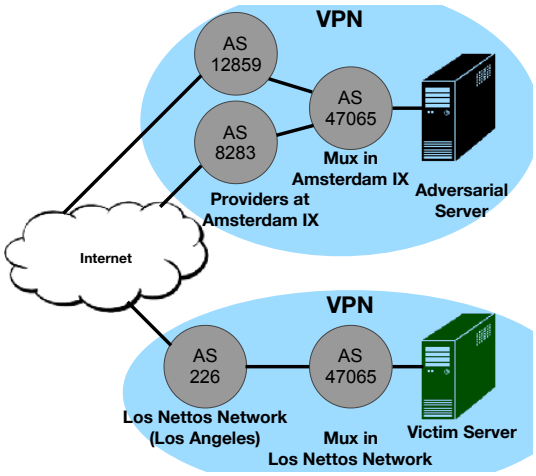


Figure 3: Experimental setup to launch BGP attacks.

3 Launching Ethical Attacks in the Wild

We successfully performed all the attacks in Section 2 in an ethical manner on the real Internet using trusted CAs.

3.1 Experimental Setup

Our experimental setup consisted of an adversarial server and a victim server. Each server was configured to make BGP announcements and forward packets through the muxes in the PEERING testbed [40]. In this experiement, we will consider a victim server in Ohio that is connected to a mux in the Los Nettos Regional Network in Los Angeles over a VPN tunnel, and an adversarial server sited in London that is connected to a mux at the Amsterdam Internet Exchange over another VPN tunnel (shown in Figure 3). Note that the adversary has two different upstream providers, making it multi-homed and capable of launching AS-path poisoning attacks.

3.2 Real-World BGP Attacks

Control Setup. We start by announcing a /23 IP prefix we controlled to the Los Nettos Regional Network. Internet traffic to the victim's domain came through the Los Nettos Regional Network to the victim's server.

3.2.1 Sub-Prefix Hijack Execution

We left the victim's network configuration untouched, and then used the adversarial server in London to make malicious BGP announcements for a more specific /24 prefix containing the victim's domain through the mux at the Amsterdam Internet Exchange. We then waited several minutes for the announcement to propagate. We

subsequently approached leading certificate authorities and requested a certificate for the victim's domain. Because the domain resolved to an IP in the hijacked sub-prefix, we were able to complete the domain control verification process without any access to the victim's server. We also successfully repeated this process using a prepended sub-prefix hijack attack where the victim's ASN was prepended to the adversary's announcement.

3.2.2 Equally-Specific-Prefix Hijack Execution

Using a similar configuration to the sub-prefix attacks, we announced the same /23 prefix as the victim from the mux at the Amsterdam Internet Exchange. Because these attacks do not affect traffic globally, we used ICMP Ping to determine which ASes had been hijacked by our announcement. We then made sure to request a certificate from a CA located in the hijacked section of the Internet. We repeated this process with and without origin AS prepending. Similar to the case above, we obtained a certificate without needing access to the victim's server.

3.2.3 AS-Path Poisoning and Traffic Interception

We launched an AS-path poisoning attack and tested the capability of these attacks to perform interception of encrypted traffic. We first observed the AS path and next hop of the route used by the mux at the Amsterdam Internet Exchange for the victim's prefix. Next, we set up a static route to forward all traffic destined to the victim's prefix to the next hop we had recorded (the only traffic that did not match this rule was traffic from the IP used by a CA for domain control verification).

We then made a route announcement for a sub-prefix (that contained the victim's domain) with every AS between the adversary and the victim prepended to the AS path. Because the announcement was for a sub-prefix, all ASes routed traffic to the adversary with the exception of the ASes between the adversary and the victim (which did not adopt the announcement because of loop prevention). Since the ASes between the victim and the adversary did not adopt the malicious announcement, the static route we configured to the victim allowed the adversary to properly forward all of the traffic to the victim and cause **no** effect on global connectivity.

With traffic forwarding in place, we approached a CA and requested a certificate. The traffic from the CA's server was not forwarded to the victim and was instead answered by the adversary's server, allowing us to obtain a trusted TLS certificate with no impact on the victim's connectivity. We then deployed this certificate to a web server run by the adversary. Finally, we removed the routing rule for traffic forwarding to the victim and answered HTTPS requests using the adversary's web server and trusted certificate. To measure the effect of this attack on real users, we simulated an innocuous user of the victim's domain by continually running HTTPS AJAX calls to the victim's domain. We observed that with no interruption in connectivity, the AJAX calls went from being securely sent to the victim's server to being read by the adversary. We were able to execute this attack in as little as 35 seconds (from BGP announcement to HTTPS traffic decryption).

	Let's Encrypt	GoDaddy	Comodo	Symantec	GlobalSign
Time to issue certificate	35s	<10min	51s	6min	4min
Human Interaction	No	No	No	No	No
Multiple Vantage Points	No[3]	No	No	No	No
Validation Method Attacked	HTTP	HTTP	Email	Email	Email

Table 3: The 5 CAs we attacked and obtained certificates from. We found that all CAs were automated and none had any defenses against BGP attacks.

3.3 Certificate Authorities Attacked

In addition to the variety of BGP attacks used, we also assessed the vulnerability of various CAs to the use of these BGP attacks to obtain bogus certificates. Table 3 lists the CAs we approached for certificates. For each CA, we launched a sub-prefix hijack attack against a victim's HTTP server (for HTTP verification) or Email server (for email verification) depending on the verification method preferred by the CA. Since the sub-prefix hijack attack is the most detectable attack, if a CA does not notice such an attack and signs a certificate, it must have no BGP defense in place and thus will not be able to detect any more advanced attacks.[1] We also recorded the relevant server logs to see if CAs had fetched the relevant resources on our servers from multiple IP addresses (indicating deployment of multiple vantage points). No CAs had such a countermeasure in place. We also noted the speed that each CA issued a certificate. All CAs signed our requests with no direct human interaction,[2] allowing for an adversary to obtain a certificate very rapidly. Since our experiment, Let's Encrypt has deployed one of our suggested countermeasures.

[1] As noted in Section 3.2.2 and Section 3.2.3, we also performed BGP equally-specific-prefix attacks and AS-Path poisoning attacks against a *chosen* CA (and not against all CAs).

[2] The longer delay from several CAs is due to the time it took us to manually request certificates from those CAs through web interfaces.

[3] No vantage points were deployed at time of attack. Let's Encrypt has since implemented multiple vantage point verification in their staging environment, where it is being tested before full release.

3.4 Attacks on Victim DNS

In addition to spoofing HTTP/Email domain verification by hijacking the victim's HTTP/Email servers, we launched attacks targeting the victim's DNS server. Once we had captured traffic to the victim's authoritative DNS server, we ran an adversarial DNS server configured to give a fake response for the A records associated with the victim's domain. When the CA performed a DNS lookup required for HTTP/Email verification, our adversarial DNS server responded with the IP of the adversary's server. The CA then sent the HTTP request/Email to the adversary's server instead of the victim's server.

3.5 Ethical Considerations

While performing these experiments, we made sure to not harm or interfere with the operations of real users or real web sites by following three important guidelines: 1) We only requested certificates for domains we registered strictly for the purpose of this experiment. Thus, these domains had no real users, and no users were affected when we obtained certificates for these domains. 2) We only made BGP announcements for IP prefixes that were allocated to us through the PEERING testbed, and all BGP announcements were originated by an AS belonging to the PEERING testbed. Thus, our experiment did not affect any other Internet traffic. 3) We did not generate any network traffic with a source address that we did not control (source IP spoofing). By following these guidelines, our experiments used real Internet infrastructure but did not affect any real users.

In this section, we demonstrate real-world BGP attacks that successfully obtain bogus certificates from the five largest CAs. We show that network-level adversaries can undermine the security properties offered by HTTPS by targeting domain validation protocols and attack users that are seemingly visiting a "secure" site. This motivates our work in Section 5 on developing countermeasures to prevent these attacks from ever harming real users. We have also reached out to Let's Encrypt to discuss the deployment of countermeasures.

4 Quantifying Vulnerability of Domains and CAs

The degree of vulnerability of the PKI to the various attacks outlined above depends on several factors like the topological relationship between the adversary and the victim and the length of the victim's prefix. We aim to measure these factors and quantitatively assess the viability of the attacks. Specifically, we aim to analyze what fraction of certificate signings could have been spoofed using one of the attacks above. Our measurement of domains reveals that 72% of domains are vulnerable to sub-prefix attacks (that can be launched by *any* AS on the Internet). All of the domains are vulnerable to an equally-specific-prefix attack, from an average of 70% of ASes on the Internet (specific to any given victim domain).

4.1 Data Collection

To gather data about TLS domains, we scraped the Certificate Transparency logs through crt.sh [4] and resolved the domain names in the common name field of certificates to an IP address. For each certificate, we resolve the common name to an IP address using our local DNS resolver.[4] We then map the IP address to the IP prefix and origin AS using Level3's routing table from the time the certificate was issued (see Section 5.2.1 for an explanation of our use of historical BGP data). We chose 10 of the 14 top CAs listed on W3Techs CA usage survey from 17th November 2017 [8] for our study. The 10 CAs were selected because of their consistent logging of Domain Validated (DV) certificates to Certificate Transparency. We performed filtering to exclude domains that fail to resolve to an IP address. Also, because of the large volume of certificates being signed, we were forced to rate limit our certificate scraping.[5] Over the period between 3/11/17 and 8/7/17, we generated a dataset of 1.8 million certificates after filtering.

4.2 Vulnerability to Sub-Prefix Attacks

We first evaluate the vulnerability to sub-prefix attacks, where the adversary AS announces a longer prefix than the original prefix. We evaluate vulnerability of both domains and CAs to such attacks.

4.2.1 Vulnerability of Domains

Because the majority of ASes filter BGP announcements to prefixes longer than /24, only domains running on prefixes shorter than /24 are vulnerable to sub-prefix attacks. That said, *our data shows that 72% of domains (1.3 million in our dataset) requesting certificates ran on prefixes shorter than /24 at the time of requesting certificate.* Figure 4 shows the complete distribution of domains over different IP prefix length. Thus, a sub-prefix hijack/interception attack is very viable on the PKI.

[4] Wildcard certificates were ignored because some CAs require DNS verification for wildcard certificates [5] and thus do not contact the server running at the domain's A record.

[5] To ensure our sample was representative, we obtained another sample of certificates directly from Let's Encrypt's logs (the CA most affected by the rate limiting) and compared the distribution of prefix lengths and originating ASes. We found these distributions to be similar implying that our research findings were not significantly impacted by the rate limiting.

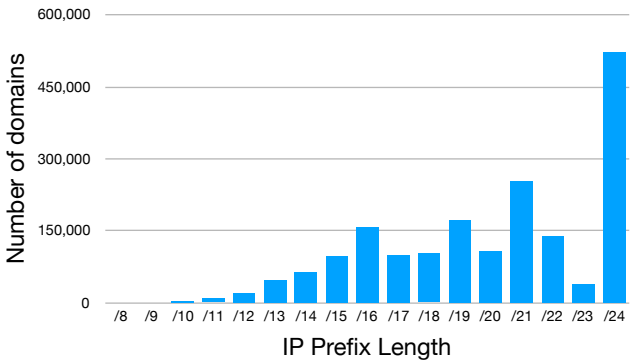


Figure 4: Number of domains hosted in an IP prefix of a given length. Only 28% of domains are on /24 prefixes.

Remark: While works on BGP attacks in other applications have recommended that ASes announce /24s to prevent sub-prefix attacks [44, 45], this is not feasible for domain owners. Owing to the very large number of domains with TLS certificates, running every domain on a /24 would cause a sizable increase in BGP routing table. Thus, in the absence of feasible countermeasures, 72% of domains are vulnerable to sub-prefix attacks. This motivates our work on designing new countermeasures for PKI in Section 5.

4.2.2 Vulnerability of CAs

CAs are also a target for attacks. Of the five CAs we performed attacks on, only one (Comodo) ran the IP used for verification out of a /24 prefix. Table 4 shows the IPs we observed CAs using for verification and the prefix length for each IP. We also show the originating AS and the number of providers (including tier 1 networks) of the originating AS. Unlike the large number of domains, there is a fairly small number of CAs, and it would be reasonable for CAs to run the IPs used for domain control verification on a /24 prefix to avoid sub-prefix hijacks. In addition, Comodo and GoDaddy operate their own ASes, meaning that running the verification servers on a /24 IP prefix would require only an update in routing policy. For CAs that do not control their own BGP announcements, we recommend negotiations with the relevant ISPs because running domain control verification servers on /24 IP prefixes has a sizable security benefit with little additional cost as explained in Section 2.2.1.

4.3 Vulnerability to Equally-Specific-Prefix Hijacking

To assess the vulnerability of domains and CAs to equally-specific-prefix attacks, we used the notion of *resilience* [31]. An AS of a CA v is *resilient* to an attack launched by a false origin AS a on a victim domain AS t, if v is *not deceived* by a and still sends its traffic to t. For a given (v, a, t) pair, resilience is calculated by:

$$\bar{\beta}(t,v,a) = \frac{p(v,t)}{p(v,t)+p(v,a)}$$

where $p(v,a)$ is the number of equally preferred paths from CA v to false origin a and $p(v,t)$ is the number of equally preferred paths from CA v to victim domain t. We perform the path inference based on (1) local preference of customer routes over peer routes over provider routes and (2) shortest AS path as outlined by Gao et al. [21].

Then, for a given CA v and victim domain t, we will consider all other ASes as possible attackers a and aggregate the above values to obtain a resilience for pair (v,t). We computed such resilience values for all pairs of the top ten CAs and the 12992 victim domain ASes in our dataset using the AS topology published by CAIDA in October of 2017.

Resilince is largely determined by AS interconnectivity. ASes with a larger number of neighbors tend to have higher resiliences (especially if these neighbors are tier 1 providers) because they are closer to other parts of the Internet, which makes their route more preferable. AS size (as measured by infrastructure or geographic area covered) does not directly influence resilience but is correlated, because large ASes are more likely to have a larger number of neighbors.

	Let's Encrypt	GoDaddy	Comodo	Symantec	GlobalSign
IP Used	64.78.149.164	68.178.177.122	91.199.212.132	69.58.183.55	114.179.250.1
IP Prefix	/20	/22	/24	/20	/11
Origin AS	AS13649	AS26496	AS48447	AS30060	AS4713
Num. Providers	5	4	4	4	0
# Tier 1 Providers	4	4	1	4	AS4713 is Tier 1
Resilience of CAs (section 4.3.2)	0.887	0.731	0.217	0.440	0.587

Table 4: This table shows the IPs used by various CAs to perform domain control verification.

4.3.1 Resilience of Domains

Figure 5 shows the *average resilience* of the domains averaged over the top ten CAs. We can see that 50% of the domains have resilience values lower than 57%, meaning that if an adversary selects a *random CA* to issue a certificate for these victim domains, there would be at least 43% probability that the adversary would be able to launch an equally-specific-prefix hijack and obtain the bogus certificate from that CA.

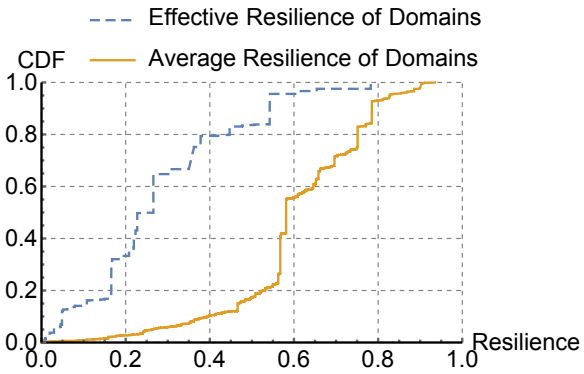


Figure 5: Average resilience and effective resilience of victim domains considering the top ten CAs.

Furthermore, an adversary can choose a target CA to exploit as opposed to choosing a random CA to increase the probability of success. Thus, we also compute the *effective resilience* of the domains by taking the minimum resilience value from the top ten CAs, also shown in Figure 5. We can see that the effective resilience is a lot lower than the average resilience. 50% of the domains have resilience values lower than 30%, meaning that if an adversary targets one of the ten CAs to issue a certificate for these victim domains, there would be at least 70% probability that the adversary would succeed. Note that there are many more CAs than the top ten CAs we considered in our dataset, so considering a larger set of CAs could further lower the effective resilience.

4.3.2 Resilience of CAs

Similarly, we compute the average resilience of CAs by averaging over all victim domains. We show the average resilience in the last row in Table 4 for the five CAs that we attacked in Section 2.

There is high variation among the resiliences of CAs. Let's Encrypt's resilience is very high (.887) because it has four direct tier 1 providers and is one hop away from much of the Internet, so its announcement will likely be preferred over the adversary's announcement. On the flip side, Comodo has a very low resilience (0.217) because it has only one direct tier 1 provider. This makes the path longer for Comodo to reach the rest of the Internet and likely less preferred over an adversary's announcement.

5 Countermeasures for CAs

At the time we performed our attacks, no CAs we studied had any countermeasures in place to prevent BGP attacks from acquiring bogus TLS certificates.[6] As a result, all attacks we launched and theorized were possible against leading CAs. In this section, we present two countermeasures that can be deployed by CAs to mitigate these attacks: multiple vantage point verification and BGP monitoring system.

To test the effectiveness of these countermeasures, we developed our own implementation of both countermeasures in the Let's Encrypt code base and relaunched the attacks in an attempt to fool our modified CA. We found that our defenses are effective in mitigating the attacks discussed in this paper.

5.1 Multiple Vantage Point Verification

As discussed in Section 2.2, equally-specific-prefix attacks and AS-path poisoning attacks do not affect the whole Internet. The former affects only a local network and the later does not affect the on-path ASes from the adversary to the CA. In other words, while the attack *successfully captures* traffic from the CA, it *will not capture* traffic from other parts of the Internet. Thus, it is important for CAs to perform domain control verification from a global perspective by repeating the verification from multiple vantage points.[7]

We propose a multiple vantage point verification method that can be deployed by CAs (with a similar motivation to the Perspectives [47] and Double Check [12] systems for trust-on-first-use protocols). The CAs will establish multiple vantage points in several different ASes. During the domain verification process, CAs will perform domain verification from all these vantage points. Our proposal in this section focuses on the HTTP verification method. We provide an adapted proposal on the Email verification method in Appendix B.

5.1.1 Vantage Point Selection

Given limited resources available for deploying vantage points, we need to strategically select the vantage points to maximize the security. Two distinct factors contribute to the quality of a set of vantage points:

1. The uneven distribution of domains. As shown in Table 5, five ASes host nearly 50% of all the domains in our dataset. Vantage points that are topologically closer to these ASes are preferable to more distant vantage points.
2. Vantage point diversity. Vantage point sets that are more spread out across the Internet topology are

[6] Since the time of our work, Let's Encrypt has deployed the multiple vantage point countermeasure presented in this section in their staging environment. We will discuss their deployment and our recommendations.

[7] Note that the multiple vantage point verification is effective against attacks that do not have a global effect. To defend against attacks that have a global effect (e.g., traditional sub-prefix attacks), we propose a BGP monitoring system in Section 5.2.

ASN	Organization	# domains	Resilience
53831	SquareSpace	260045	0.166
26496	GoDaddy	239226	0.306
14618	Amazon	155593	0.542
16276	OVH	146780	0.362
62679	Shopify	60157	0.378
37963	Alibaba	52769	0.378
16509	Amazon	36014	0.783
24940	Hetzner	33855	0.219
197695	Reg.ru	23506	0.378
32475	SingleHop	20166	0.108
All Other ASes	-	819366	-

Table 5: Top ten ASes by number of hosted domains.

more difficult to attack with a single localized routing announcement.

With these criteria in mind, we designed an algorithm to select preferred vantage points for a given CA. The algorithm requires a set of customer domains (in our case, domains from our dataset of certificates), and a list of candidate vantage points (e.g., data centers where the CA can potentially deploy vantage points). Fundamentally, the algorithm attempts to find a set of vantage points with the maximum resilience *as a set*. We calculate the resilience for a set as following. We first compute the resilience of each sample domain from each vantage point in the set, as explained in Section 4.3. Then, we take the maximum resilience of each domain from the previous step. We then average the maximum resiliences over all domains to obtain the resilience for the set.[8]

Next, our algorithm has three nested steps:

1. Vantage Point Set Improvement: The algorithm begins with an initial set of randomly-selected vantage points from the list of candidate vantage points. Then, for each vantage point in the set, the algorithm substitutes that vantage point with the potential vantage point (chosen from the list of candidate vantage points) that causes the set of vantage points to have the greatest resilience increase.
2. Finding a Local Maximum: The process of vantage point set improvement is repeated until the set of vantage points can no longer be improved. We refer to this set of vantage points as a local maximum.
3. Using Randomization to find a Global Maximum: Given a set of candidate vantage points, there exist several local maximum of which only one is a global maximum (i.e., the optimal set of vantage points). To increase the likelihood of finding a global maximum, our algorithm repeats the above steps with random initial vantage points to find as many local maximum as possible.

We found that there is a roughly 18% chance that a local maximum found by the script will be the global maximum we eventually found (when considering a set of five vantage points chosen from 1,000 candidate vantage points). Thus, the above algorithm can find global maximums with a reasonable number of repetitions.

This algorithm can also let CAs find out how best to expand while utilizing existing infrastructure. To compute *additional* vantage points given a set of already deployed vantage points, we simply consider certain vantage points in the candidate set to be fixed (e.g., CA's existing vantage points such as its own data center) and we do not consider alternatives to these vantage points.

5.1.2 Vantage Point Evaluation

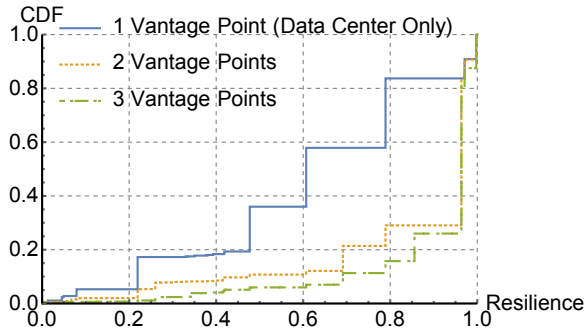


Figure 6: Resilience for Let's Encrypt with varying numbers of vantage points.

We evaluate resilience for Let's Encrypt with different numbers of vantage points, shown in Figure 6. The baseline is 1 Vantage Point, where the CA only performs domain control verification from its own existing AS/data center without any additional vantage points (in Let's Encrypt's case, the ViaWest data center AS 13649 is the fixed vantage point). This gives an average resilience of domains of 61%, meaning an attack will have a 39% chance of success. When the number of vantage points is more than one, the adversary must hijack traffic from all of the vantage points to deceive the CA. This greatly reduces the chance of success for the attacker. Note that this evaluation considers the *domains* as the target of BGP attacks, whereas resiliences shown Table 4 considers the *CAs* as the target.

We can see that, with only one additional vantage point (two vantage points in total), there is already a 24% increase over the baseline (to an average resilience of 85%). With three vantage points, the resilience is at

[8] This calculation is actually a lower bound on the true resilience of a set of vantage points as an adversary must fool *all* vantage points in the set and not just the vantage point closest to the domain. However, computing the true resilience for all sets of vantage points is computationally infeasible.

least .9 for 74% of the domains, meaning that the attacker only has 10% probability to succeed (a 28% improvement over the baseline).

5.1.3 Let's Encrypt's Deployment

Our work was a key factor in Let's Encrypt's preliminary deployment of multiple vantage points in their staging environment, which is used for testing features before full release in the production environment [37]. Here we present a discussion of the current staging environment implementation and some of the changes Let's Encrypt is making in the full release.

Vantage point location. Based on our measurements in Let's Encrypt's staging environment [6], Let's Encrypt deployed two remote vantage points in addition to their original data center in AS 13649 (ViaWest). The two vantage points were located in Amazon data centers in Ohio and Frankfurt. Although these vantage points have a broad geographic distribution, they are not sufficiently diverse in terms of network topology. Both vantage points are run by Amazon and both belong to the same AS 16509, which are likely to have similar BGP routes. Thus, in the full release, the Let's Encrypt team plans to improve AS-level diversity by deploying more vantage points in distinct ASes located in different parts of the Internet topology.

Handling anomaly. Let's Encrypt's staging environment deployment permits one of the remote vantage points (although not the original data center) to time out, which allows for network/hardware failures and maintains a low false positive rate. However, this also weakens the security guarantee of the system. If one vantage point is allowed to time out, then the system will miss out on the routing information from that vantage point. Furthermore, strategic attackers can target vantage points that may be able to observe the attack, and launch DoS attacks against the target to make it time out.

Given the tradeoff between a strong security guarantee and false positives in the event of a network failure, we propose that (1) there be a limit on the total number of vantage points allowed to time out, and (2) at least one vantage point in each AS where vantage points are deployed be required to send a response. We recommend this method in order to tolerate failure while still providing strong security.

5.2 Monitoring BGP Route Age

We present a new BGP monitoring system that is specifically tailored for deployment by CAs with a novel route age detection heuristic.

Traditional general purpose BGP monitoring systems attempt to maintain a low false positive. However, some seemingly innocent BGP route updates that would normally not be labeled suspicious can be used to target the PKI. For example, the announcement of a single prefix over a peering relationship with the true origin prepended would likely not attract much attention because little traffic would be misdirected. If a traditional BGP monitoring system were to flag such an announcement, there would likely be an unreasonable number of false positives. However, such a leak could allow an adversary to obtain a bogus TLS certificate. Thus, a monitoring system for CAs needs to be more aggressive about flagging routes as suspicious than a traditional monitoring system for general security purposes.

Route Age Heuristic. We propose a new mechanism, the route age heuristic, to detect suspicious routes for CAs that would likely be missed by a traditional monitoring system. At a high level, the route age heuristic computes an *age* for each route the CA's ISP is using and flags routes that are too new. *This would force attacks to be active for a minimum amount of time before a CA would be willing to sign a certificate based on them.* In this system, legitimate users with recent BGP routes will have their certificates signed after the routes have sufficient age. However, adversaries are required to leave their attacks active, so network operators have time to react. There is a clear tradeoff between false positives (legitimate users that are unnecessarily delayed) and this minimum time threshold. A larger minimum time allows network operators more time to shutdown a potential BGP attack but will clearly cause CAs to delay signing a larger number of certificates that are coincidently based on very recent routes. Our goal is to engineer a method to compute the age of a route that allowed for a minimum time threshold that was long enough for network operators to react but also did not have an unreasonably high false positive rate.

Algorithm. Our heuristic considers the age of the last three hops of a route: the origin and the two ASes before the origin. We use a different threshold value for each hop. Our algorithm computes the age based on 1) *how long any route to a given prefix had been seen (network age)* and 2) how long each hop in the route to that prefix had been seen. To compute the age of each hop, we constructed an SQL database containing, for each prefix, the last seen AS path and a list of timestamps indicating when each AS was added to that path. To populate the database, our algorithm compares the AS path of each new update for a prefix with the previously stored AS path. Working one AS at a time in the AS path, the algorithm checks to see if each new AS differed from the stored AS. If the two ASes are the same, the algorithm keeps the stored time stamp for that hop because there has been no change in that particular hop on the route. However, if the two ASes differ, the algorithm uses the

timestamp of the new BGP update for that hop and all hops after that hop. To compute the hop ages of a prefix, the algorithm looks up a prefix in the database and computes for each hop the current timestamp subtracted by the stored timestamp for that hop. With these hop ages, a CA can make fine tuned judgements as to whether a route is considered old enough to be used in domain control verification.

5.2.1 Evaluating False Positives

We evaluated the false positive rate of our monitoring system by simulating its hypothetical deployment by the Let's Encrypt CA. We combined the 1.2 million certificates from Let's Encrypt in our dataset with historical BGP data. Using BGPStream from CAIDA [38], we replayed historical BGP updates and routing information base data (RIBs) from Level 3 (AS 3356) through routeviews2 vantage point. Level 3 was selected because it is a tier one ISP and it is a provider to Let's Encrypt.

We seeded our database by loading in a RIB from one month before our earliest certificate. We then began processing BGP updates (from after the RIB we loaded) and certificates in lockstep. If a BGP update had a timestamp greater than the timestamp of the oldest unprocessed certificate, we would look up the resolved IP address from the certificate in our database and find the longest prefix match. We then recorded the age of the route used when the signing CA performed domain control validation for this certificate. This process was continued until we had collected the age on the routes used for every certificate in the database.

We found that with a reasonable set of thresholds, we were able to obtain a false positive rate of 1 in 800 certificates. Table 6 shows the tradeoff between false positive rates and threshold values. At the 1 in 800 false positive rate, an adversary would be forced leave subprefix attacks active for 30 hours because these attacks announce new networks and would have to meet the network age threshold before being used by CAs. During this time, traditional manual means of attack detection (that network operators rely on heavily [41]) would be able to shut down the attack. Note that the certificates that would trigger false positives would not require human intervention from CAs. The CAs may automatically retry the certificate signing later once the BGP route announced by the domain's ISP becomes stable.

False Positive Rates	Network Age	Origin Age	Provider Age	3rd Hop Age
1 in 100	285	52	3.6	4.6
1 in 200	159	33	1.5	1.6
1 in 400	50	17	0.56	0.56
1 in 800	30	6	0.11	0.11

Table 6: The minimum time thresholds (in hours) for hops in the AS path with different false positive rates.

6 Related Work

BGP Attacks on Infrastructure and Applications. BGP attacks have been shown to have a sizable effect on various applications. Sun *et al.* have shown the effectiveness of BGP attacks at deanonymizing Tor users [44], and Apostolaki *et al.* demonstrated the use of BGP to attack the Bitcoin protocol [13]. Arnbak *et al.* also showed how entities such as NSA can use BGP to bypass US surveillance laws [15]. Gavrichenkov performed a preliminary exploration of BGP attacks on TLS [22], which only considered the most basic traditional subprefix and equally-specific-prefix hijacks. We are the first to consider more sophisticated attacks and perform real-world demonstrations of all the attacks, as well as develop countermeasures.

BGP Attacks and Defenses. Previous work by Pilosov and Kapela has demonstrated the use of advanced BGP attacks with strategically poisoned AS paths [39]. The vulnerability of peering links has also been explored by Madory [36]. However, no previous work has applied these BGP attacks to target encrypted communications.

BGP defenses have been studied in both general and application-specific forms. Lad *et al.* outline a well-known system to detect traditional BGP attacks using origin changes [30]. RPKI can be used to authenticate the origin ASes of BGP routes and generate route filters to prevent BGP attacks [17]. Both these systems only operate on the origin AS of a BGP announcement and can be fooled by prepended ASNs [23]. BGPsec cryptographically assures the validity of BGP paths and is immune to such prepending attacks [33]. However, BGPSec is not deployed and researchers have shown that partial BGPSec deployment does not bring significant security improvement [35]. Additionally, SCION presents a clean slate architecture that would prevent BGP hijacks [48]. SCION has been deployed in production environment of multiple ISPs but is still not used by the vast majority of the Internet. Karlin *et al.* introduced the idea of cautiously adopting new routes to avoid routing based on malicious BGP announcements [28]. We adapt this idea to the PKI by developing a more complex measurement of age and recommending CAs not use new routes during domain control verification.

Sun *et al.* developed an application-specific BGP monitoring system to protect the Tor network that includes a similar analytic using route age [43]. Our study considers a more nuanced notion of age and uses it to advise CAs in certificate signing as opposed to alerting prefix owners of an attack.

Work on Domain Control Verification. Recent work

has been making major improvements in standardizing the process of domain control verification. The security flaws in the operations of the CA WoSign highlighted the importance of port standardization during domain control verification [3] which was reflected in the CA/Browser Forum ballot 169 [10]. Ballot 169 is also the first document to rigorously enumerate which methods a CA can use for domain control verification.

Bootstrapping Trust Through DNS. Proposals like DANE [25] and RAINS [46] offer alternatives to the current PKI by including server public key information directly in the name server infrastructure, which is cryptographically verified. DNSSEC [14] provides additional security to the existing PKI by preventing network attacks on DNS-based domain control validation methods through cryptographic signatures on DNS responses.

7 Conclusion

We explore BGP attacks that can be used against the PKI and successfully demonstrate real-world BGP attacks against top CAs. We then assess the degree of vulnerability of the current PKI. Our analysis shows that the *vast majority* of domains are vulnerable to a sub-prefix or equally-specific-prefix attack that an adversary can use to obtain a bogus certificate. In addition to exploring the attack surface, we propose and implement countermeasures that can significantly reduce the vulnerability of the PKI. We recommend performing domain control verification from multiple vantage points, and develop a BGP monitoring system with a novel route age analytic that can be used by CAs. Overall, our work is the first work to develop a taxonomy of BGP attacks on on PKI (and demonstrate these attacks in the real world), and the first to propose realistic countermeasures that have already started being adopted by CAs.

8 Acknowledgments

We would like to thank Michael Bailey for shepherding this paper, Adrian Perrig for detailed feedback, Josh Aas for feedback on Let's Encrypt's deployment, and the anonymous USENIX reviewers for their suggestions and comments. We would also like to thank Let's Encrypt for their partnership, which has lead to the first implementation of multiple-vantage-point verification and has provided us with crucial data to support this research. In addition we are grateful for support from the National Science Foundation under grant CNS-1553437 and the Open Technology Fund through their Securing Domain Validation project.

References

[1] 556468 - investigate incident with RapidSSL that issued SSL certificate for portugalmail.pt. https://bugzilla.mozilla.org/show_bug.cgi?id=556468.

[2] CAIDA spoofer project. https://www.caida.org/projects/spoofer/.

[3] CA:WoSign Issues. https://wiki.mozilla.org/CA:WoSign_Issues#Issue_L:_Any_Port_.28Jan_-_Apr_2015.29.

[4] Certificate search. https://crt.sh/.

[5] Godaddy: Verify domain ownership (HTML or DNS). https://www.godaddy.com/help/verify-domain-ownership-html-or-dns-74521.

[6] Let's Encrypt staging environment. https://letsencrypt.org/docs/staging-environment/.

[7] Moscow traffic jam. https://radar.qrator.net/blog/moscow-traffic-jam.

[8] Usage of SSL certificate authorities for websites. https://w3techs.com/technologies/overview/ssl\textunderscorecertificate/all.

[9] Youtube hijacking: A RIPE NCC RIS case study. https://www.ripe.net/publications/news/industry-developments/youtube-hijacking-a-ripe-ncc-ris-case-study, Mar 2008.

[10] Ballot 169 - revised validation requirements. https://cabforum.org/2016/08/05/ballot-169-revised-validation-requirements/, Oct 2016.

[11] Ballot 190 - revised validation requirements. https://cabforum.org/2017/09/19/ballot-190-revised-validation-requirements/, Sep 2017.

[12] ALICHERRY, M., AND KEROMYTIS, A. D. Doublecheck: Multi-path verification against man-in-the-middle attacks. In *IEEE Symposium on Computers and Communications* (July 2009), pp. 557–563.

[13] APOSTOLAKI, M., ZOHAR, A., AND VANBEVER, L. Hijacking bitcoin: Routing attacks on cryptocurrencies. In *IEEE Symposium on Security and Privacy (SP)* (May 2017), pp. 375–392.

[14] ARENDS, R., AUSTEIN, R., LARSON, M., MASSEY, D., AND ROSE, S. DNS security introduction and requirements. RFC 4033, RFC Editor, March 2005. http://www.rfc-editor.org/rfc/rfc4033.txt.

[15] ARNBAK, A., AND GOLDBERG, S. Loopholes for circumventing the constitution: Unrestricted bulk surveillance on americans by collecting network traffic abroad. *Mich. Telecomm. & Tech. L. Rev. 21* (2014), 317.

[16] BIRGE-LEE, H., SUN, Y., EDMUNDSON, A., REXFORD, J., AND MITTAL, P. Using BGP to acquire bogus TLS certificates. *HotPETS'17*.

[17] BUSH, R., AND AUSTEIN, R. The resource public key infrastructure (RPKI) to router protocol. RFC 6810, RFC Editor, January 2013.

[18] CA/BROWSER FORUM. *Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates, v.1.5.4*, Oct 2017.

[19] COWIE, J. China's 18-minute mystery — Dyn blog. https://dyn.com/blog/chinas-18-minute-mystery/, Nov 2010.

[20] DURUMERIC, Z., KASTEN, J., BAILEY, M., AND HALDERMAN, J. A. Analysis of the HTTPS certificate ecosystem. In *Internet Measurement Conference* (New York, NY, USA, 2013), IMC '13, ACM, pp. 291–304.

[21] GAO, L., AND REXFORD, J. Stable Internet routing without global coordination. *IEEE/ACM Transactions on Networking (TON) 9*, 6 (2001), 681–692.

[22] GAVRICHENKOV, A. Breaking HTTPS with BGP hijacking. *Black Hat USA Briefings* (2015).

[23] GILAD, Y., COHEN, A., HERZBERG, A., SCHAPIRA, M., AND SHULMAN, H. Are we there yet? on RPKI's deployment and security.

[24] GREENBERG, A. How an unprecedented heist hijacked a bank's entire online operation. `https://www.wired.com/2017/04/hackers-hijacked-banks-entire-online-operation/`, Jun 2017.

[25] HOFFMAN, P., AND SCHLYTER, J. The DNS-based authentication of named entities (DANE) transport layer security (TLS) protocol: TLSA. RFC 6698, RFC Editor, August 2012. `http://www.rfc-editor.org/rfc/rfc6698.txt`.

[26] HU, X., AND MAO, Z. M. Accurate real-time identification of IP prefix hijacking. In *IEEE Symposium on Security and Privacy (SP)* (May 2007), pp. 3–17.

[27] HUSTON, G. Nopeer community for border gateway protocol (BGP) route scope control. RFC 3765, RFC Editor, April 2004.

[28] KARLIN, J., FORREST, S., AND REXFORD, J. Autonomous security for autonomous systems. *Computer Networks 52*, 15 (2008), 2908–2923.

[29] KRUEGEL, C., MUTZ, D., ROBERTSON, W., AND VALEUR, F. Topology-based detection of anomalous BGP messages. In *Symposium on Recent Advances in Intrusion Detection (RAID)* (2003), pp. 17–35.

[30] LAD, M., MASSEY, D., PEI, D., WU, Y., ZHANG, B., AND ZHANG, L. PHAS: A prefix hijack alert system. In *USENIX Security Symposium* (2006), vol. 1, p. 3.

[31] LAD, M., OLIVEIRA, R., ZHANG, B., AND ZHANG, L. Understanding resiliency of Internet topology against prefix hijack attacks. In *IEEE/IFIP Conference on Dependable Systems and Networks* (2007), IEEE, pp. 368–377.

[32] LANGLEY, A., KASPER, E., AND LAURIE, B. Certificate Transparency. RFC 6962, RFC Editor, June 2013.

[33] LEPINSKI, M., AND SRIRAM, K. BGPsec protocol specification. RFC 8205, RFC Editor, September 2017.

[34] LONE, Q., LUCKIE, M., KORCZYŃSKI, M., AND VAN EETEN, M. Using loops observed in traceroute to infer the ability to spoof. In *International Conference on Passive and Active Network Measurement* (2017), Springer, pp. 229–241.

[35] LYCHEV, R., GOLDBERG, S., AND SCHAPIRA, M. BGP security in partial deployment: Is the juice worth the squeeze? In *ACM SIGCOMM* (New York, NY, USA, 2013), pp. 171–182.

[36] MADORY, D. Use protection if peering promiscuously. `https://dyn.com/blog/use-protection-if-peering-promiscuously/`, Nov 2014.

[37] MCCARNEY, D. Validating challenges from multiple network vantage points. `https://community.letsencrypt.org/t/validating-challenges-from-multiple-network-vantage-points/40955`, Aug 2017.

[38] ORSINI, C., KING, A., GIORDANO, D., GIOTSAS, V., AND DAINOTTI, A. BGPStream: A software framework for live and historical BGP data analysis. In *ACM on Internet Measurement Conference* (2016), ACM, pp. 429–444.

[39] PILOSOV, A., AND KAPELA, T. Stealing the Internet: An Internet-scale man in the middle attack. *NANOG-44, Los Angeles, October* (2008), 12–15.

[40] SCHLINKER, B., ZARIFIS, K., CUNHA, I., FEAMSTER, N., AND KATZ-BASSETT, E. Peering: An AS for us. In *ACM Workshop on Hot Topics in Networks* (2014), ACM, p. 18.

[41] SERMPEZIS, P., KOTRONIS, V., DAINOTTI, A., AND DIMITROPOULOS, X. A survey among network operators on BGP prefix hijacking. *SIGCOMM Comput. Commun. Rev. 48*, 1 (Apr. 2018), 64–69.

[42] SHI, X., XIANG, Y., WANG, Z., YIN, X., AND WU, J. Detecting prefix hijackings in the Internet with Argus. In *Internet Measurement Conference* (New York, NY, USA, 2012), IMC '12, ACM, pp. 15–28.

[43] SUN, Y., EDMUNDSON, A., FEAMSTER, N., CHIANG, M., AND MITTAL, P. Counter-raptor: Safeguarding tor against active routing attacks. In *IEEE Symposium on Security and Privacy (SP)* (May 2017), pp. 977–992.

[44] SUN, Y., EDMUNDSON, A., VANBEVER, L., LI, O., REXFORD, J., CHIANG, M., AND MITTAL, P. Raptor: Routing attacks on privacy in Tor. In *USENIX Security Symposium* (2015), pp. 271–286.

[45] TODOROVIC, B. BGP spoofing in the episode: Stealing your (cc)TLD. *NANOG-45, Santo Domingo, January* (2009).

[46] TRAMMELL, B. RAINS (Another Internet Naming Service) Protocol Specification. Internet-Draft draft-trammell-rains-protocol-03, Internet Engineering Task Force, Sept. 2017. Work in Progress.

[47] WENDLANDT, D., ANDERSEN, D. G., AND PERRIG, A. Perspectives: Improving ssh-style host authentication with multi-path probing. In *USENIX Annual Technical Conference* (2008), vol. 8, pp. 321–334.

[48] ZHANG, X., HSIAO, H. C., HASKER, G., CHAN, H., PERRIG, A., AND ANDERSEN, D. G. Scion: Scalability, control, and isolation on next-generation networks. In *IEEE Symposium on Security and Privacy (SP)* (May 2011), pp. 212–227.

[49] ZHAO, X., PEI, D., WANG, L., MASSEY, D., MANKIN, A., WU, S. F., AND ZHANG, L. An analysis of BGP multiple origin AS (MOAS) conflicts. In *ACM SIGCOMM Workshop on Internet Measurement* (New York, NY, USA, 2001), IMW '01, ACM, pp. 31–35.

A Appendix: Additional Attacks

Below are attacks we were unable to perform on the PKI but could still be used by certain strategically positioned adversaries to gain bogus certificates with a high degree of stealthiness.

A.1 Intentional Route Leak

An attack that follows naturally from Table 1 is the intentional route leak, where the adversary prepends the AS path to the victim (as in the AS path poisoning attack) and announces equally-specific prefix. This attack is very stealthy because the adversary is in effect only improperly propagating a legitimate announcement it has heard from one of its neighbors. Such route leaks are relatively common because of misconfigurations [36] [7]. However, while seemingly innocuous, a route leak can route vital traffic through an adversary that could be used to gain a bogus certificate.

Intentional route leaks are not viable in many situations even when several CAs can be targeted. The adversary's route announcement must have the entire route to the victim prepended and is for the same prefix announced by the victim. Thus, many ASes will prefer the victim's original announcement to the adversary's announcement due to the long AS path in the adversary's announcement. However, these attacks are effective at capturing traffic in a localized portion of the Internet topology, and if an adversary is very topologically close to a CA (or happens to have favorable business relations) the attack is viable.

The viability of this attack increases significantly if we assume an adversary has complete administrative control of an AS (as opposed to only the technical ability to make announcements). If so, an adversary could realistically approach a victim's ISP and request to become peers with that ISP. In this way, the adversary has favorably changed the Internet topology to make the attack more viable. To illustrate this, let us consider ViaWest (Let's Encrypt's ISP). Peers of ViaWest are in a prime position to launch an intentional route leak. ViaWest would likely prefer a route from a peer over a provider route even if the AS path was longer in the peer route allowing these peers to launch an intentional route leak. In addition, this route leak would not be globally visible and would only influence ViaWest and its clients. While only 24 ASes are currently seen peering with ViaWest (peering links are also the hardest BGP relations to detect so 24 may be an underestimate), ViaWest has a Point Of Presence (POP) at the Seattle Internet Exchange (SIX) and is colocated with 283 other ASes. ViaWest also has an open peering policy, meaning that proposals to establish peering sessions with ViaWest are welcome and easily accepted. From this point of view, all 283 ASes at the Seattle Internet Exchange are in a good position to launch an intentional route leak. This trend is commonly seen with several top CAs that operate out of large data centers. Data centers often have open peering policies and POPs at many Internet exchanges to reduce latency and transit costs. However, this makes data centers prime targets for such topology manipulation. We believe this creation of peering links to change the Internet topology in an adversary's favor merits further study that uses both network analysis and studies of business practices to understand and counter this vulnerability.

We were not able to launch an intentional route leak because of guidelines imposed by the peering framework on the number ASes that can be prepended to an announcement. In addition, without administrative control of the peering framework we were not able to establish additional peering links that might make such an attack possible.

A.2 Limited Propagation Attack

Limiting the propagation of a malicious BGP announcement by announcing only to a peer AS as opposed to a provider can help an adversary to maintain as much connectivity as possible and reduce the control plane noticeability. To perform this attack we launched a sub-prefix hijack attack from the mux at the Amsterdam Internet Exchange but made the announcement only to the peer Hurricane Electric.[9]

We then ran our own non-trusted CA in a network that was a customer of Hurricane Electric. Using the NTT looking glass and our mux in the Los Nettos Regional Network, we confirmed that the adversary's announcement had not propagated globally (e.g. to NTT's network) and instead had only propagated to the customers of Hurricane Electric (e.g. the Los Nettos Regional Network). We requested a certificate from our non-trusted CA and obtained one without modifying the victim's server. We repeated a similar variation of this experiment but announced the route to peer AS 8075 (Microsoft) as opposed to Hurricane Electric (we also moved our CA into AS 8075 so it would not be affected by the hijack). While using Microsoft instead of Hurricane Electric is not a significant difference from a BGP perspective, it makes the attack significantly more stealthy for an adversary. While Hurricane Electric has many client ASes that could easily detect the attack, Microsoft has only 10 customer ASes that are all under Microsoft's administrative control. Thus, this announcement to Microsoft has such limited propagation that a vantage point within Microsoft's network is needed for the attack to be detected.

While we used a non-trusted CA for this experiment, it would still be reasonable for an adversary to launch this attack against a trusted CA given: 1) a broader selection of CAs than we explored and 2) the ability of an adversary to construct peering connections with potential target ASes. In the version of this experiment using Hurricane Electric, it would have been reasonable to find a CA with Hurricane Electric as a provider. While we did not find any CAs located in Microsoft data centers, we did find a CA that used Amazon's data centers. Had Amazon instead of Microsoft been a peer available for us to make an announcement, we would have been able to gain a trusted certificate while only propagating a route to a single organization.

A variant of this attack we did not perform is the use of BGP communities to limit propagation. It is already understood that well-known communities such as no-peer

[9] In order for this experiment to work we moved the victims announcement from the mux at Los Nettos Regional Network to the mux in the Greek Research and Technology Network because Hurricane Electric would prefer the announcement from the Los Nettos Regional Network (a customer route) over the adversary's announcement from the Amsterdam Internet Exchange (a peer route).

and no-export can make BGP attacks harder to detect by limiting propagation [27]. However, in the case of the PKI, these mechanisms for limiting propagation are more relevant as an adversary's choice of CA increases the likelihood that the CA will be topologically close to the adversary. Thus, methods for limiting propagation are more likely to be applicable in such situations.

Similar to the intentional route leak, an adversary could reasonably perform a limited propagation attack given the ability to establish peering links with target ASes.

B Appendix: Using Multiple Vantage Points for Email

The aforementioned multiple vantage point verification works well for HTTP verification and DNS TXT verification that rely on checking the existence of given data in a domain's infrastructure. However, some CAs also use email verification, which is based on proving that a user can read data sent to a domain.

Challenges in email verification. A naive implementation of the multiple vantage point verification for emails would be to have multiple locations on the Internet send emails and have the users prove that they received all of the emails. However, this is a manual form of domain control verification where a real human user is expected to read the emails from the CA and take actions accordingly. Having the users read and respond to multiple identical emails from the vantage points is not practical.

Our proposed email verification. To address the above concern, we propose a system *where a single email can be sent from multiple locations on the Internet*. We assume the CA has set up secure VPN tunnels with the vantage points. The steps are as follows.

1. The CA breaks up the secret information that needs the domain owner's action (e.g. verification URL) into several pieces so that there is at least one piece for each vantage point.

2. The CA's mail server sends the first piece of the secret via email to the domain's mail server.

3. Upon receiving the TCP ACKs from the domain's mail server, the CA reconfigures its routing policy to route the email traffic through the first vantage point via the VPN tunnel, and sends the second piece of the secret to this vantage point.

4. Upon receiving the TCP ACKs via the first vantage point, the CA repeats the above step using the next vantage point, etc., until all the pieces of secret have been sent.

In this way, the domain owner has the impression of only receiving one email from the CA, but in fact an arbitrarily large number of vantage points were used to send the email.