Mapping the Attack Surface of Domain Control Validation in TLS Server Certificate Issuance

By
Derek R. Greene

A THESIS

submitted to

Oregon State University

Honors College

in partial fulfillment of
the requirements for the
degree of

Honors Baccalaureate of Science in Computer Science
(Honors Associate)

Presented March 3, 2025
Commencement June 2026

AN ABSTRACT OF THE THESIS OF

Derek R. Greene for the degree of Honors Baccalaureate of Science in Computer Science presented on March 3, 2025. Title: Mapping the Attack Surface of Domain Control Validation in TLS Server Certificate Issuance

Abstract approved:

____

Zane Ma

In the Web PKI, certificates serve as the mechanism linking domains to public keys. Domain Control Validation (DCV) is a critical component of the Web PKI, ensuring domain ownership before certificate issuance occurs. This step prevents individuals from obtaining certificates for domains they do not control. If DCV fails, attackers could obtain certificates for domains they do not own, resulting in an erosion of trust and security on the internet. To our knowledge, this work is the first to comprehensively review the security of all existing DCV methods, and map the entire attack surface. We analyze the domain control validation (DCV) methods and their dependencies. In our analysis, we map the attack surface and identify sections for debloating. We also investigate the diversity of DCV methods used by Certificate Authorities (CAs) in the Chrome, Mozilla (NSS), Microsoft, and Apple root stores, identifying BR section 3.2.2.4.12 to be a possible candidate for debloating. In our work addressing threats to specific protocols, we find the potential scope of some scenarios to be quite wide. For example, we found the email address contact@privacyprotect.org used as a contact email in WHOIS records for 7,581 (2.24%) of the domains we collected.

Key Words: Transport Layer Security, CA Browser Forum, Domain Control Validation, Certificate Authority, Web PKI

Corresponding e-mail address: derek@derekrgreene.com

Mapping the Attack Surface of Domain Control Validation in TLS Server Certificate Issuance

By
Derek R. Greene

A THESIS

submitted to

Oregon State University

Honors College

in partial fulfillment of
the requirements for the
degree of

Honors Baccalaureate of Science in Computer Science
(Honors Associate)

Presented March 3, 2025
Commencement June 2026

Honors Baccalaureate of Science in Computer Science project of Derek R. Greene presented on March 3, 2025.

APPROVED:

Zane Ma, Mentor, representing College of Engineering; School of Electrical Engineering and Computer Science

Renato Figueiredo, Committee Member, representing College of Engineering; School of Electrical Engineering and Computer Science

Dave Nevin, Committee Member, representing College of Engineering; School of Electrical Engineering and Computer Science

Toni Doolen, Dean, Oregon State University Honors College

I understand that my project will become part of the permanent collection of Oregon State University Honors College. My signature below authorizes release of my project to any reader upon request.

Derek R. Greene, Author

Contents

1 Introduction

The Web Public Key Infrastructure (PKI) is the backbone of security on the internet, providing server authentication through Transport Layer Security (TLS) for millions of websites, applications, and services. Web clients request TLS certificates from Certificate Authorities (CAs) who perform Domain Control Validation (DCV). A process in which DNS names are linked to public keys before certificate issuance. DCV is crucial to the entire framework of the web PKI as any exploitation at this stage has the potential to undermine the security of downstream nodes.

As more organizations adopt HTTPS, the importance of a secure Web PKI grows. The motivation of this research is to enhance the overall security of the Web PKI by enumerating the attack surface and comprehensively reviewing existing DCV methods. If an attacker obtains an SSL certificate for a fully qualified domain name (FQDN), they can impersonate the domain and compromise security. Given the ever-increasing reliance and trust users place on the web, it becomes increasingly important that the process for obtaining certificates is secure.

While prior work has found multiple security vulnerabilities in DCV [1, 2, 3], this study is, to our knowledge, the first to comprehensively review the security of all existing DCV methods, and to map the entire attack surface for DCV. In addition, our analysis provides context to help guide web PKI policy, including the potential impacts of vulnerability for all existing DCV methods. By analyzing the methods allowed for DCV and factors such as popularity, threats, and redundancy, we are able to further enhance the overall security and trust of the ecosystem.

In this paper, we first compile all current DCV methods and outline their dependencies to visualize the full attack surface. Next, we compiled a collection of known attacks amongst each protocol. We identified the use of outdated and insecure methods such as SMS, fax, and postal mail. Given these communication channels lack encryption or authentication, they are vulnerable to interception and spoofing.

We then measure the frequency of use for each DCV method to determine which nodes of the attack surface are most viable for debloating. As DCV utilization data is not publicly accessible nor apparent from certificates, we approximate DCV utilization frequency across CAs. We automate the parsing and extraction of DCV methods from CA audit and policy statements.

Our analysis found multiple DCV methods with duplicative and redundant paths, serving as potential candidates for debloating. We also find methods with a high reliance on single points of failure and low utilization. Reducing the DCV attack surface will minimize potential vulnerabilities and enhance the security of the web PKI.

In summary, this paper's contributions are:

- a comprehensive enumeration and mapping of the DCV attack surface,
- a novel approach to approximate DCV method frequency across CAs,

- a debloating analysis outlining potential DCV methods for elimination.

2 Background

In the web PKI, applicants generate a cryptographic key pair which is linked to a fully qualified domain name (FQDN) and sent to a CA in a certificate signing request (CSR). To ensure applicants control the domain for which a certificate is requested, CAs perform Domain Control Validation (DCV) which involves the applicant successfully passing a challenge specified by the CA.

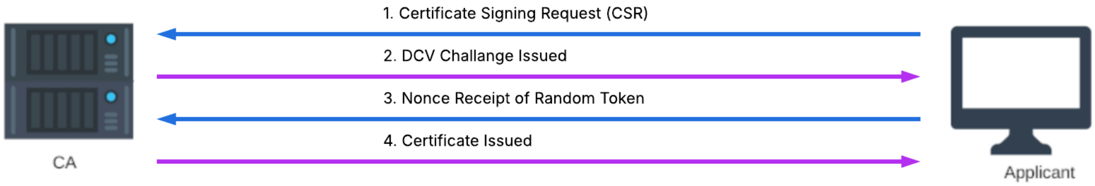
1. Certificate Signing Request (CSR)
2. DCV Challange Issued
3. Nonce Receipt of Random Token
4. Certificate Issued
CA
Applicant

Figure 1: Domain Control Validation

These challenges are defined in the Baseline requirements (BR) which define the set of standards CAs are required to adhere to in the issuance and management of SSL/TLS certificates. The Baseline Requirements are defined and revised by the CA/Browser Forum. There are currently 13 different DCV methods outlined in the BRs that CAs are authorized to utilize, however each CA may offer only a subset of these 13 methods to their subscribers. The DCV methods vary in terms of nonce-transmission and nonce-receipt methods such as email, phone, fax, DNS based.

Figure 2 shows the permitted DCV methods, which all follow a general four-stage process: 1) the subscriber requests a certificate for one or more FQDNs, 2) the CA looks up some form of contact information for the domain(s), 3) the CA communicates a random value/token to the subscriber, and 4) CA retrieves/receives the random value from the subscriber. We group DCV paths according to their BR definitions and enumerate the different options at each stage. Some DCV paths traverse multiple chains of contact information (e.g., DNS A record, followed by reverse IP lookup, then another A record resolution), creating a more convoluted process. Table 1 below provides a high level description of each DCV method, grouped by communication channel.

BR Section	Description	Communication Channel
3.2.2.4.20	Random token sent during TLS handshake using Application-Layer Protocol Negotiation (ALPN).	TLS
3.2.2.4.19	Random value placed in the ”/.well-known/acme-challenge/[token]” directory and retrieved via HTTP or HTTPS.	HTTP
3.2.2.4.18	Random value placed in the ”/.well-known/pki-validation” directory and retrieved via HTTP or HTTPS.	HTTP
3.2.2.4.17	Random value provided by phone to contact listed in DNS CAA record	Phone
3.2.2.4.16	Random value provided by phone to contact listed in DNS TXT record.	Phone
3.2.2.4.15	Random value provided by phone to contact listed in WHOIS.	Phone
3.2.2.4.14	Random Value sent to DNS TXT Email Contact.	Email
3.2.2.4.13	Random value sent to DNS CAA Email contact.	Email
3.2.2.4.12	CA confirms subscriber is ’domain contact’ through WHOIS or DNS SOA record.	N/A
3.2.2.4.8	Reverse IP address lookup.	DNS
3.2.2.4.7	Random value placed in DNS CNAME, TXT, or CAA record.	DNS
3.2.2.4.4	Random value sent to ’admin’, ’administrator’, ’webmaster’, ’hostmaster’, or ’postmaster’ followed by ’@’ and the FQDN.	Email
3.2.2.4.2	Random value sent via email, SMS, fax, or postal mail contact listed in WHOIS contact information.	Email, SMS, Fax, Postal

Table 1: DCV Method Descriptions

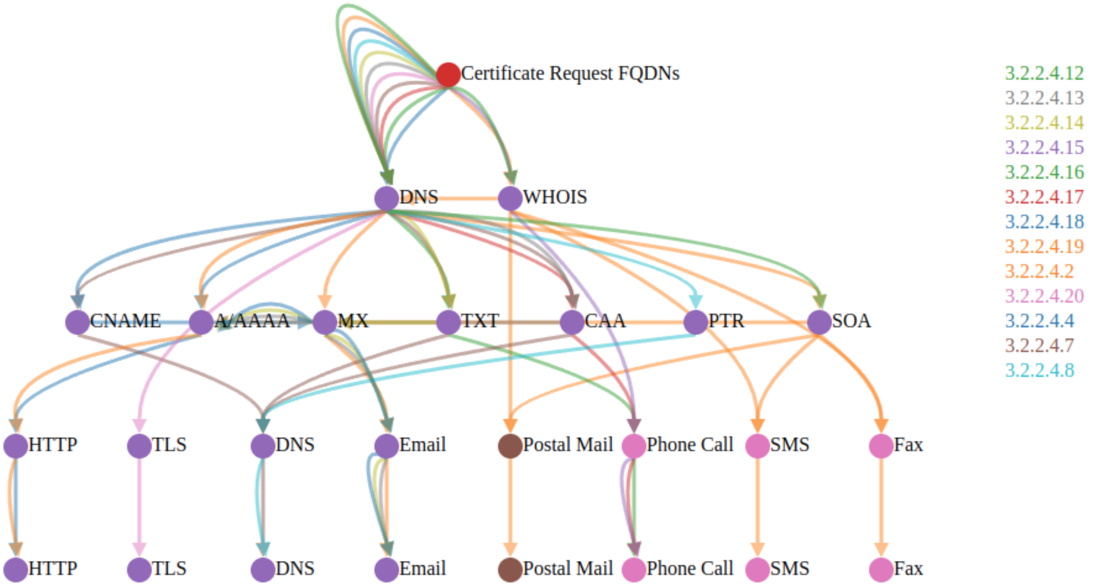

Figure 2: Domain Validation Methods Map

3 Methodology

Our goal in this work is to measure the diversity and frequency of DCV methods across certificate authorities in order to assess the theoretical impacts of compromise to any given node in the DCV process and to estimate the popularity of each DCV method. We measure diversity in two ways, by enumerating the number of CAs that support each BR method and each DCV node. We also measure the frequency of each DCV method. Ideally, we would have all certificates and their corresponding DCV method; however this is challenging as the DCV methods used are not listed in certificates nor transparency logs. Additionally, the massive scale of certificate issuance presents significant challenges in collecting data for all certificates issued across the internet. We instead collect a representative sample from certificate transparency (CT) logs and estimate associated DCV methods with information provided in each CAs certificate practice statements (CPS).

To gather the list of domains analyzed, Certstream Server Go [4] was configured in a docker container to stream CT log data via websocket connections. To collect the data from this stream, a python script was developed which continuously receives a stream of domains from CT Logs, records the data in JSON, and subsequently performs ZDNS scans (CAA and TXT records) on the domains. Each domain is prepended with the ”validation-contactemail” and ”validation-contactphone” subdomains before being piped through ZDNS [7] to retrieve the corresponding CAA and TXT records. In addition, WHOIS data for the domains analyzed was collected. A second Python script was developed to analyze the WHOIS data and return records using known disposable email domains in the registrant_email WHOIS field.

In examining root store data to determine the popularity and diversity of DCV methods, we analyzed the “All Certificate Information (root and intermediate) in CCADB” CSV from the Common CA Database [5] and additional data provided from Google outlining the DCV methods used by each root in the Chrome root store. To create an estimate of popularity, we totaled the number of CAs and Parent CAs that support each method across the Chrome root store. Our estimate does not provide an accurate representation of actual DCV use, but rather represents the number of CAs that support each DCV method. This is because the actual usage data is not made publicly available by CAs.

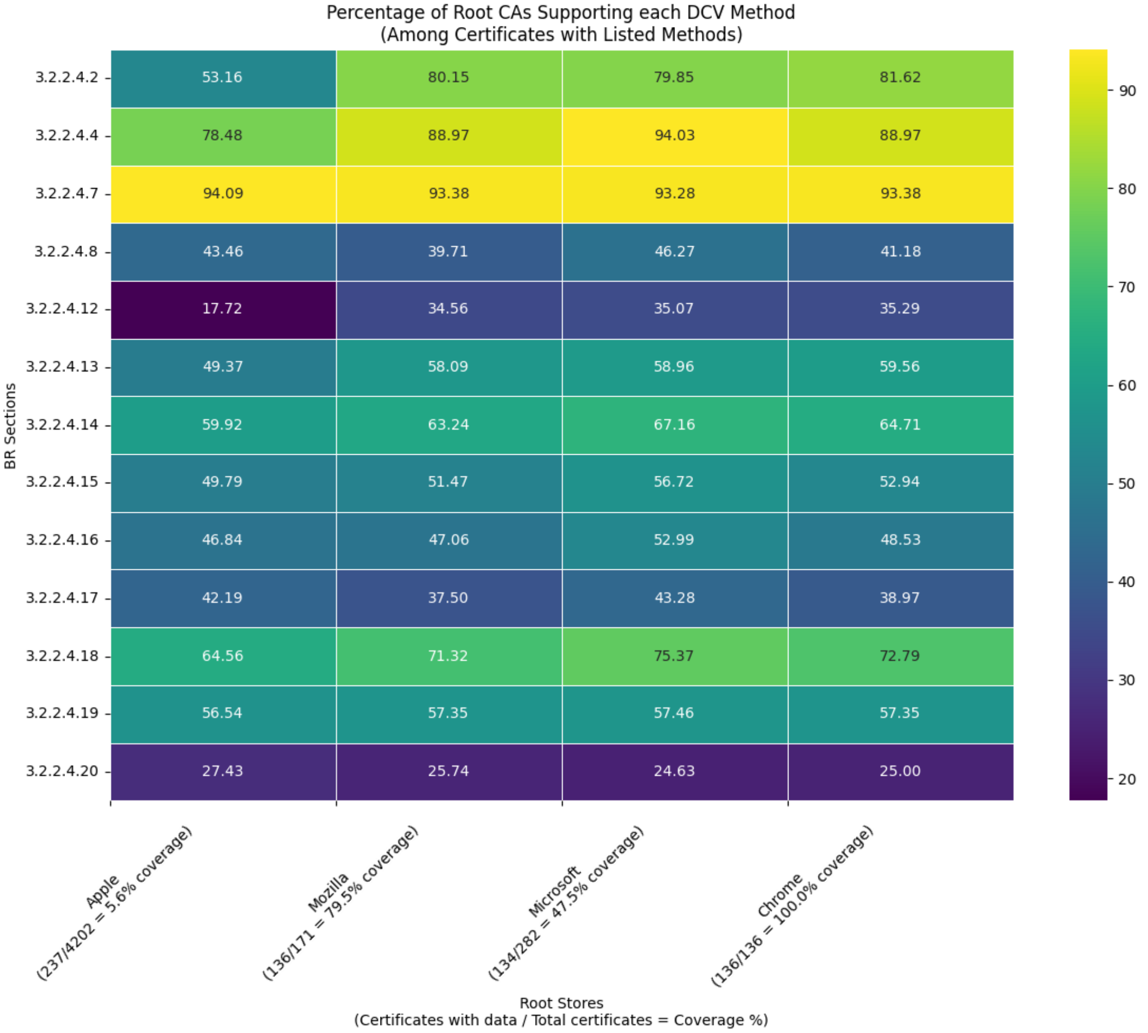

Figure 3: Percentage of Root/Parent CAs Supporting each DCV Method

To determine the maximum impact of compromise to specific protocols, we generated estimates based on the certificate records collected over the course of this research. The real-world maximum impact of compromise for any of the methods addressed is much higher given that this study only collected data for a small fraction of domains on the internet.

3.1 Limitations

There are several limitations to the data collected that should be acknowledged. The limited set of domains extracted from certificate transparency logs only represent a fraction of the FQDNs on the internet. As actual DCV method utilization data is not publicly available, our measurements relied upon compiling supported methods across root CAs. Therefore, our measurement does not represent frequency of utilization but rather the frequencies of which DCV methods each root CA allows. In the process of compiling the list of supported DCV methods for each root CA (excluding those in the chrome root store), OpenAI ChatGPT 4 model was used to automate the parsing of certificate practice statement PDF files and may have returned incorrect information. To mitigate the potential for this occurring, multiple manual checks were performed to ensure the information returned from the AI model was accurate in those instances. The analysis of WHOIS data for identifying disposable relies on the data being up-to-date which is not always the case for WHOIS records. Additionally, the list of known disposable email domains utilized for these studies measurements was compiled from multiple sources and some email domains listed may not actually be disposable email services.

4 Results

4.1 Threats to DCV dependencies

To catalog the attack surface of DCV methods, we analyzed the dependency concentration and vulnerability exposure. This section examines the attack surface of DCV methods as specified by the CA/Browser forum as of version 2.0.4 (August 2024). We first present an overview of valid DCV paths, which highlights the diversity of protocols that could be targeted by an attacker. This analysis also identifies which DCV methods, if removed, would be most effective at reducing overall attack surface. Next, we measure individual nodes in the attack surface to point out areas of concern and identify instances of vulnerability. Finally, we study the concentration of dependencies at each attack point to quantify the impact of an optimized attack.

Dependency concentration in DCV methods increases the potential attack surface. Since most CAs only require a single successful DCV method, every node results in a single point of failure, meaning if an attacker gains control over one of these, they can bypass DCV altogether. For example, if an attacker can exploit DNS, then all methods reliant upon DNS become exploitable. We categorize these dependency concentrations into two groups, individual contact information and contact infrastructure provider. Individual contact information encompasses the contact details used by DCV methods and include email addresses, phone and fax numbers, and addresses. If one of these dependencies is compromised, all FQDNs utilizing these are potentially compromised as well. The contact infrastructure providers group includes service providers such as DNS, WHOIS, HTTP, and email service providers. If an attacker compromises a node in this group, all FQDNs utilizing that service are potentially compromised. For example, if an attacker is able to compromise Gmail, they would potentially be able to obtain a certificate for any FQDN using a @gmail domain.

A total of 10 (83%) DCV methods rely on DNS, which is vulnerable to BGP attacks, as shown in the paper Bamboozling Certificate Authorities with BGP [1]. This attack specifically targets the HTTP GET request made by a CA when performing DNS lookups for HTTP and email-based DCV methods.

A total of 2 (0.16%) DCV methods depend on WHOIS, whose main vulnerability is the use of weak contact information, such as disposable email, outdated phone and fax numbers, and outdated addresses. Although these all present as user errors, they are still present as vulnerabilities. Notably, 52 domains of the 337,378 domains collected were identified as using disposable email accounts in WHOIS contact information. Of these, seven were found to be fully accessible. That is, any individual with the email address can access the corresponding inbox. We extrapolated these results further by performing bulk WHOIS queries for all domains using specific known disposable emails in WHOIS such as "@yopmail.com" and identified hundreds of thousands of additional domains using disposable email.

The vulnerability that arises from allowing postal mail as a DCV method is the chance for postal addresses to be compromised or fall under the control of another individual, such as with P.O. boxes. We found only 1 (0.07%) DCV method relies on postal mail.

A similar vulnerability arises with DCV methods that utilize phone contact, as with postal mail. Namely, the opportunity for a phone number to become under the control of a different individual. We found 3 (0.23%) DCV methods rely on phone contact.

A total of 4 (0.30%) DCV methods depend on email and are susceptible to a plethora of vulnerabilities due to the current design. As these methods rely on WHOIS or DNS records to obtain contact emails, they are vulnerable to both compromised DNS and the aforementioned use of disposable email addresses.

4.2 DCV Method Redundancy

During stage two of the general DCV process, in which the CA looks up the contact information for the associated domain, the contact information may be obtained from several DNS records including CAA, TXT, and SOA. To determine potential methods for debloating, we listed the methods with the highest number of connected and dependent nodes and factored in practical considerations such as popularity and potential impact to domain owners.

We found BR section 3.2.2.4.20 is an easy candidate for debloating as it has the least support among CAs and in its current form, is under-specified and unclear in terms of implementation.

Additionally, we find sections 3.2.2.4.7, 3.2.2.4.13, 3.2.2.4.14, 3.2.2.4.16, and 3.2.2.4.17 all rely upon a subscriber proving control over either a CAA, TXT, or CNAME record. A subscriber would require the same access to place contact information in any of these DNS records. Allowing multiple sources bloats the certificate transparency process and does not provide any fundamental benefit to subscribers.

While BR definitions for nonce transmission and nonce receiving are clear in some instances, such as section 3.2.2.4.2, other DCV methods remain ambiguous in their guidelines. Section 3.2.2.4.12 is unclear in terms of both nonce transmission and receiving. Email based DCV methods are of particular interest as they only require a user to prove the ability to receive but not write, which is a much weaker standard for proof of control over a FQDN.

4.3 Popularity of DCV Methods

The maximum impact a compromised email address presents is dependent on how frequently it is used across domains. The resultant effect is the potential compromise of all domains utilizing the compromised email; in some instances, this scope is rather significant. For example, the email address contact@privacyprotect.org is listed as a contact email in WHOIS records for 7,581 domains of the records analyzed. However, the potential for a compromised email to be effective in the domain validation control process requires that an attacker can even use this compromised email to obtain a certificate. In instances where a CAA record has been set, if the CA(s) listed do not allow email-based DCV methods, a compromised email would not directly result in an attacker obtaining a certificate for the domain. Notably, this would also prevent the individual with control over the domain from utilizing email-based DCV methods.

The maximum impact of a compromised email provider is dependent on the use of each provider. Analysis found that Gmail is the most popular domain used outside privacy-protected email domains, with 11,202 Gmail addresses used in WHOIS records for the domains analyzed. Additionally, 2,071 Outlook domains, 1,356 AOL domains, and 1,174 Yahoo domains, among other email domains. The resultant effect of a compromised email provider is that all email addresses using that domain would be potentially compromised.

The maximum impact of a compromised phone or fax number is similar to that of email, depending on the frequency of its use across domains. In cases where a phone number is only used across a single domain, the scope is limited, however if used across multiple domains an attacker would potentially be able to obtain an SSL certificate for any of those domains. However, the compromise of a phone or fax number alone will not directly allow an attacker to obtain a certificate for a domain if a CAA record is set listing CA(s) which do not utilize phone and fax based DCV methods.

The maximum impact of a compromised DNS is the complete takeover of a domain. This is because nonce transmission and nonce receiving are downstream functions of domain validation dependent on DNS. Although CAA records are still useful for securing domain validation, if DNS is compromised, CAA records can be modified by an attacker to allow any CA to issue certificates for the domain. In most instances, if an attacker has successfully compromised DNS, modification of CAA records is not needed unless the CA(s) listed do not allow DCV methods reliant upon DNS. For example, if the listed CA only allowed BR section 3.2.2.4.15 which entails a call placed to the number found in a WHOIS query.

4.4 DCV Method Infrastructure

Unfortunately, the data recording which DCV method was used to obtain a certificate is not publicly available. In order to estimate popularity, CAs and Parent CAs were compared against each DCV method that they support.Using this scoring, BR Section 3.2.2.4.7 has the highest popularity, with 149 (97%) Parent CAs and 305 (94%) CAs supporting this method. BR section 3.2.2.4.4 has the second-highest popularity, with 140 (91%) Parent CAs and 258 (80%) CAs supporting this method. Section 3.2.2.4.18 has the third-highest popularity at 111 (72%) Parent CAs and 200 (62%) supporting this method.

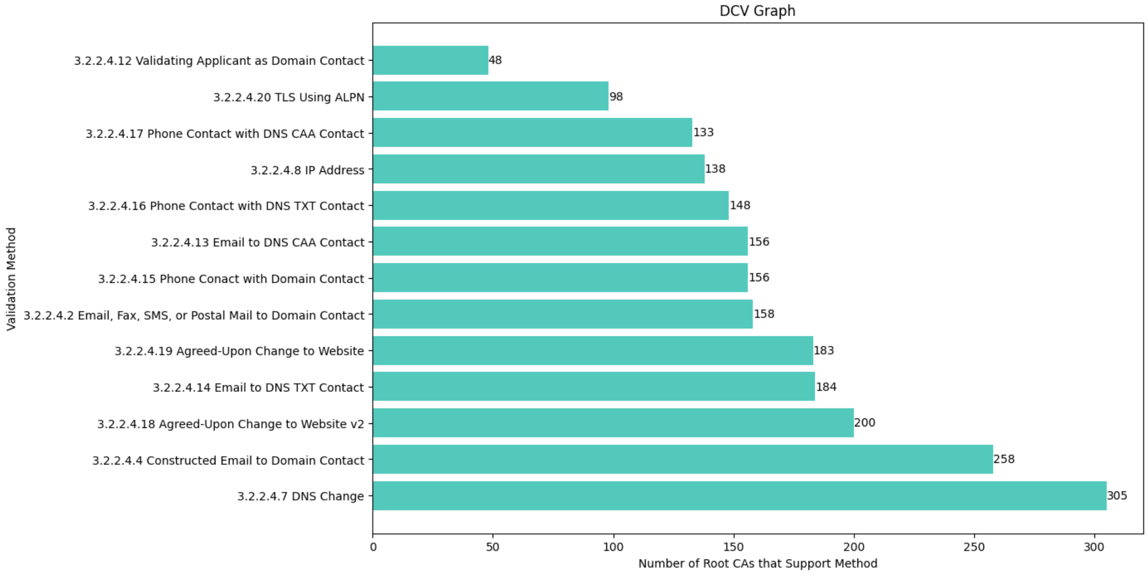

Figure 4: Domain Validation Methods Grouped by CA

The least popular DCV method is BR Section 3.2.2.4.12 with 47 (30%) Parent CAs and 48 (14%) CAs supporting this method. The second least popular method is Section 3.2.2.4.20 with 46 (30%) Parent CAs and 48 (30%) CAs supporting this method. The third-least popular is Section 3.2.2.4.17 with 62 (40%) Parent CAs and 133 (41%) CAs supporting this method.

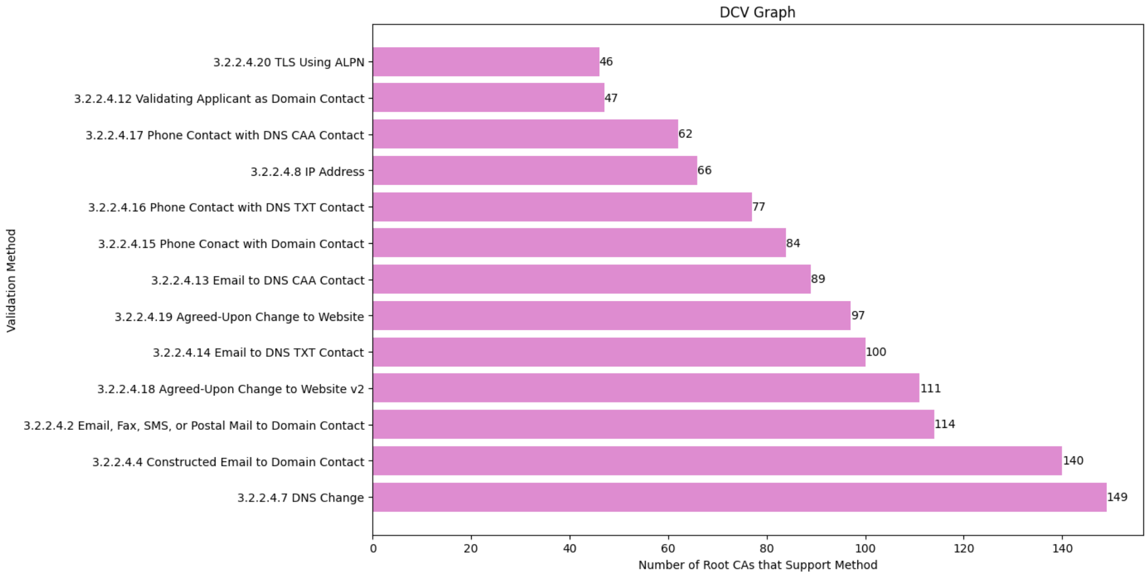

Figure 5: Domain Validation Methods Grouped by Parent CA

In terms of the three least popular methods, sections 3.2.2.4.20, and 3.2.2.4.12 are likely the least popular two because of the limitations in scope. Namely, these sections restrict CAs from issuing certificates for other FQDNs ending with the validated FQDN unless a separate validation is performed for the FQDN. Additionally, these methods do not support validating wildcard domain names. For domains with many subdomains or organizations with many domains, the requirement that each FQDN be validated separately can add significant overhead, potentially explaining why these methods are less favored. Finally, for the case of 3.2.2.4.12, this method requires the CA also be the domain registrar, which further limits the scenarios where this method may be used.

Interestingly, the two least popular DCV methods 3.2.2.4.20 and 3.2.2.4.12 do not require transmission and receipt of a special token as do the three most supported DCV methods 3.2.2.4.4, 3.2.2.4.7, and 3.2.2.4.2.

Node	**CAs**	**Percentage**
WHOIS	135	100%
A/AAAA	135	100%
MX	135	100%
DNS	134	99%
CNAME	134	99%
TXT	129	95%
CAA	129	95%
SOA	119	88%
Postal	119	88%
SMS	119	88%
Fax	119	88%
HTTP	114	84%
Phone	71	52%
TLS	31	22%

Table 2: DCV Node Support by Certificate Authorities (CAs)

5 Discussion

Ultimately, TLS certificates are the keystone of server identity in web security. If the identities of FQDNs are compromised, security essentially goes out the window. For example, if an attacker can impersonate a domain, the utility of encryption becomes almost non-existent.

Future work can be done to address the security of the web PKI including collaborating with CAs to obtain and analyze DCV utilization data and performing a deeper quantification of the risks associated with legacy protocols still in use such as fax and SMS.

6 Related Works

Web Infrastructure Security Prior work has identified a critical security vulnerability exists in domain control verification. Specifically, network level attackers can employ BGP prefix hijacking to reroute traffic destined for a victim's domain, allowing them to act as a man-in-the-middle, generating certificate requests and intercepting responses. Ultimately, the vast majority of domains available on the internet were found vulnerable. As this attack exploits DNS-based domain control verification methods, all downstream methods such as email and HTTP, etc. are potentially compromised. [1]

Similarly, researchers have shown that use-after-free vulnerabilities can be implemented to obtain cloud IP addresses with stale DNS records. These stale records can then be used to impersonate ownership of a domain and thereby fraudulently obtain SSL certificates in cloud environments. This presents as a significant vulnerability to domain validation given the wide attack surface. Cloud domains hosted on both Amazon AWS and Microsoft Azure were shown to be susceptible to these vulnerabilities. In addition, use-after-free vulnerabilities were found to be both time- and cost-efficient, raising the likelihood of this vulnerability being exploited. [2]

Additionally, prior work has shown the vulnerability presented through network-level attacks such as DNS hijacking/spoofing. Researchers have demonstrated that not only is domain impersonation feasible, but there are also multiple methods by which individuals can obtain a certificate illegitimately. This work highlights the insecurity of DNS based validation methods and the domain control validation methods ultimately reliant on DNS such as email, HTTP, etc. [8]

CA Issuance Security Researchers have shown that the occurrence of certificate invalidation events leads to stale TLS certificates, which can be utilized to fraudulently obtain valid certificates. Notably, two of the three scenarios that result in state TLS certificates (domain registrant changes and managed TLS departure) occur organically and not due to any attack or exploit. This work helps to highlight the design flaws of web PKI by showing the prevalence of stale certificates and the growing gap that exists between a domain name and its cryptographic key. [6]

7 Conclusion

The current set of DCV methods permitted within the web PKI overlap and include less secure legacy communication channels such as fax and postal mail. We enumerated the dependencies for each of these DCV methods and the potential impact of vulnerability at each node. To our knowledge, we are the first to comprehensively review the security of all DCV nodes and map the attack surface. We identified multiple areas where redundancy and dependency concentration can be minimized, including BR section 3.2.2.4.20. Ultimately, we believe that reducing the attack surface by eliminating redundant and less secure DCV methods will improve the overall security of the web PKI.

References

[1] Henry Birge-Lee, Yixin Sun, Anne Edmundson, Jennifer Rexford, and Prateek Mittal. Bamboozling certificate authorities with BGP. In *USENIX Security Symposium (Sec)*, 2018.

[2] Kevin Borgolte, Tobias Fiebig, Shuang Hao, Christopher Kruegel, and Giovanni Vigna. Cloud Strife: Mitigating the Security Risks of Domain-Validated Certificates. In *Network & Distributed System Security Symposium (NDSS)*, 2018.

[3] Markus Brandt, Tianxiang Dai, Amit Klein, Haya Shulman, and Michael Waidner. Domain validation++ for mitm-resilient pki. In *ACM Conference on Computer and Communications Security (CCS)*, 2018.

[4] d Rickyy-b. Certstream server go. https://github.com/d-Rickyy-b/certstream-server-go, 2024.

[5] Common CA Database. CCADB. https://ccadb.my.salesforce-sites.com/ccadb/AllCertificateRecordsCSVFormatv2, 2024.

[6] Zane Ma, Aaron Faulkenberry, Thomas Papastergiou, Zakir Durumeric, Michael D. Bailey, Angelos D. Keromytis, Fabian Monrose, and Manos Antonakakis. Stale tls certificates: Investigating precarious third-party access to valid tls keys. In *ACM Internet Measurement Conference (IMC)*, 2023.

[7] The ZMap Project. ZDNS. https://github.com/zmap/zdns, 2024.

[8] Lorenz Schwittmann, Matthäus Wander, and Torben Weis. Domain impersonation is feasible: A study of ca domain validation vulnerabilities. In *IEEE European Symposium on Security and Privacy (Euro S&P)*, 2019.