

Security of SC-082 Redux

Henry Birge-Lee

henrybirgelee@gmail.com

Unaffiliated Interested Party Member of the CA/Browser Forum Server Certificate Working Group
USA

Keywords

Persistent DCV, Static DCV, Ballot SC-082 Redux

1 Introduction

In previous statements I have made on list I expressed support for Ballot SC-082 Redux also known as Persistent DNS Validation. I also previously posted a short security justification. The previous security justification I posted was rather brief, and I feel my position should be supported particularly given my involvement as an unaffiliated interested party. In conclusion I find compelling evidence to support this ballot for the good of the ecosystem and the security of the PKI as a whole.

I will perform analysis across two fronts to answer distinct questions:

- Does the inclusion of the proposed method “3.2.2.4.22 DNS TXT Record with Static Value” degrade the security of existing domains merely by its existence in the baseline requirements (even if domain owners choose not to use it)?
- Does the use of the proposed method “3.2.2.4.22 DNS TXT Record with Static Value” expose domain owners to a larger attack surface?

This analysis is performed as of 5/8/2025 using the text in <https://github.com/slghtr-says/servercert/pull/3/files> with latest commit of 8694f3d93dfb29146a5368ba9d3b243a53347ec3.

2 Precedent

While not a rigorous argument, it is worth noting that in today’s web ecosystem, there are several times persistent values allow for authorization of domain control in contexts beyond the PKI. Several of these examples are discussed in a current Internet Draft submitted to the IETF DNS OP working group: <https://datatracker.ietf.org/doc/draft-sheth-identifiers-dns/>. Beyond these examples some examples include:

- **Source domains for email sending** At major email sending platforms (e.g., AWS SES, Sendgrid), domain control is often required for emails to be sent from the platform using a customer’s domain after the “@” sign. This verification is typically done by checking for the presence of a static DNS configuration (e.g., DKIM or Sender Policy Framework records).

- **Routing at cloud providers** Cloud providers that serve DNS records and also manage content typically check for the presence of their name servers in the NS records for the domain in question and only activate certain features after those NS records are observed.
- **Delegated enterprise accounts** Services that provide account management to enterprises may choose to allow enterprises to obtain accounts with enterprise domains after the “@” (e.g., Google Workspace). These accounts may be used for email receiving as well as authentication to 3rd-party apps. These are typically managed via the presence of a static value in the zone of the customer’s domain.

While alone these examples do not justify inclusion into the baseline requirements, they do show that validation of subscriber domains using static DNS values has precedent.

3 Security Justification for the Inclusion of DNS TXT Record with Static Value

In the current web PKI ecosystem, the applicant has choice of domain control validation in the absence of CAA records. Thus, for the approximately 85% of domains that do not have CAA records, security reasoning should assume the adversary can pick the most preferable validation method permitted in the baseline requirements even if this method is not used by the victim domain. For this reason, the CA/Browser Forum should generally be extremely cautious when adding new methods as even if these methods present value for some domains, they pose a risk to the entire community.

However, in this section I argue that the **inclusion of “DNS TXT Record with Static Value” does not expand the attack surface of domain control validation attacks on the web PKI as this attack surface is provably a subset of the attack surface presented by several existing methods.** Specifically, existing and permitted methods 3.2.2.4.14 (Email to DNS TXT Contact) and 3.2.2.4.15 (Phone Contact with Domain Contact) have an attack surface that is a superset of the attack surface “DNS TXT Record with Static Value” under reasonable adversary assumptions. Additionally, the attack surface of 3.2.2.4.7 (DNS Change) and ACME dns-01 is also a superset of “DNS TXT Record with Static Value” under slightly more permissible adversary assumptions. With this in mind, the inclusion of “DNS TXT

Record with Static Value” into the TLS BRs does not expand the attack surface of domain control validation as it is strictly a subset of existing permitted methods.

3.1 Proof Structure

This section uses proof by contradiction. We assume there the new method “DNS TXT Record with Static Value” is compromised and show that this implies an existing method (e.g., 3.2.2.4.14 Email to DNS TXT Contact) is compromised.

IF compromise in “DNS TXT Record with Static Value” implies compromise in Email to DNS TXT Contact THEN (by the contrapositive) IF Email to DNS TXT Contact is secure, “DNS TXT Record with Static Value” is secure.

This analysis can also be thought of in terms of the space of possible adversaries. IF there exists an adversary capable of compromising “DNS TXT Record with Static Value” THEN there exists an adversary capable of compromising Email to DNS TXT Contact. In other words, the attack surface of a domain is not expanded by the inclusion of “DNS TXT Record with Static Value” into the BRs.

It should be noted that this type of proof is often used in theoretical cryptography where the definitions of constructs are much more theoretical. Here the proofs must exist in the real web ecosystem degrading meaning they are less rigorous than the purely theoretical proofs from cryptography, but I still feel they provide value to the community.

3.2 Proof using 3.2.2.4.14 (Email to DNS TXT Contact) and 3.2.2.4.15 (Phone Contact with Domain Contact)

The most straightforward justification for “DNS TXT Record with Static Value” is to prove its security is a subset of 3.2.2.4.14 (Email to DNS TXT Contact) and 3.2.2.4.15 (Phone Contact with Domain Contact).

Using the proof-by-contradiction strategy outlined above, let us consider an agent capable of compromising “DNS TXT Record with Static Value.” This agent will be a black box (allowing the proof to generalize to any potential method “DNS TXT Record with Static Value” may be compromised) with the following capabilities: it can place a single static TXT record value at an underscore-prefixed subdomain of the victim’s domain at **one instant in time**. After placing the value in the victim’s subdomain, this agent cannot interact with any CA for an arbitrarily-large amount of time. This models an agent capable of compromising “DNS TXT Record with Static Value” with minimal capabilities.

We will assume in addition to access to this agent, the adversary has access to a personal email address. This email does not need to be related to the victim’s domain in any way and could be a Gmail address or an email at a domain registered by the adversary.

Steps showing compromise of “DNS TXT Record with Static Value” implies compromise of 3.2.2.4.14 Email to DNS TXT Contact

Assume the victim’s domain is victim.example.com, the adversary’s email is adversary@adversary.example.com and the adversary has access to an agent capable of compromising “DNS TXT Record with Static Value” as discussed above referred to as *agent*.

- Adversary requests *agent* place adversary@adversary.example.com at _validation-contactemail.victim.example.com.
- Adversary waits for *agent* to complete this task and performs no CA interactions.
- After *agent* completes its task, Adversary approaches a publicly-trusted CA and requests validation using 3.2.2.4.14 Email to DNS TXT Contact.
- The CA looks up _validation-contactemail.victim.example.com IN TXT and gets adversary@adversary.example.com.
- The CA sends the validation email to adversary@adversary.example.com which is read by the adversary.
- The adversary completes method 3.2.2.4.14 Email to DNS TXT and obtains a certificate for victim.example.com

This proof can be trivially adapted to show compromise of “DNS TXT Record with Static Value” implies compromise of 3.2.2.4.15 (Phone Contact with Domain Contact). The adversary simply needs to have a phone number it controls, register the label _validation-contactphone.victim.example.com with its phone number, and then request the CA use 3.2.2.4.15.

We have shown a compromise of “DNS TXT Record with Static Value” implies a compromise of 3.2.2.4.14 and 3.2.2.4.15. By the contrapositive, if we assume 3.2.2.4.14 and 3.2.2.4.15 are not compromised, “DNS TXT Record with Static Value” is not compromised.

3.3 Proof using 3.2.2.4.7 (DNS Change) or ACME dns-01

3.2.2.4.7 (DNS Change) or ACME dns-01 can also be used to prove the security of “DNS TXT Record with Static Value” with one variation: I will assume an agent that can upload a CNAME into the victim domain’s zone instead of a TXT. Given that many DNS providers have a single set of credentials that allows access to edit any record type within a zone, I feel this still provides a reasonable and valuable proof. For the purpose of this proof, I will consider an hypothetical method called DNS CNAME Record with Static Value which is identical to “DNS TXT Record with Static Value” but the required record type is CNAME instead of TXT.

Steps showing compromise of “DNS CNAME Record with Static Value” implies compromise of ACME dns-01 and 3.2.2.4.7 DNS Change

Assume the victim's domain is victim.example.com, the adversary controls a domain called adversary.example.com¹ and the adversary has access to an agent capable of compromising uploading a single static CNAME record into the victim's zone as referred to as *agent*. The steps of this proof are:

- Adversary requests *agent* upload adversary.example.com at _acme-challenge.victim.example.com IN type CNAME.
- Adversary waits for *agent* to complete this task and performs no CA interactions.
- After *agent* completes its task, Adversary approaches a publicly-trusted CA and requests validation using dns-01 or DNS Change.
- The CA presents the adversary with a challenge token to upload to _acme-challenge.victim.example.com.
- The adversary places this token at adversary.example.com
- The CA checks for the presence of the token at _acme-challenge.victim.example.com IN TXT and gets the CNAME response to adversary.example.com
- The CA's DNS resolver automatically follows the CNAME and resolves adversary.example.com IN TXT.
- The CA's infrastructure sees the challenge the adversary uploaded.
- The adversary completes method dns-01 or DNS Change and obtains a certificate.

The same proof holds for ACME dns-01 and DNS Change with record type TXT. Only the specific underscore-prefixed label would be changed.

We have shown a compromise of a hypothetical method DNS CNAME Record with Static Value implies a compromise of ACME dns-01 and DNS Change 3.2.2.4.7. By the contrapositive, if we assume ACME dns-01 or DNS Change 3.2.2.4.7 is not compromised, DNS CNAME Record with Static Value is not compromised.

The only deviation of this proof from the proposed "DNS TXT Record with Static Value" is that it assumes an adversary with ability to upload a CNAME record into the victim's zone. "DNS TXT Record with Static Value" can be compromised by an adversary that could only upload a TXT record. However, I feel this proof is still quite relevant since DNS providers usually do not differentiate access based on record type.

4 Security Justification for the Use of DNS TXT Record with Static Value

While the earlier reasoning justifies the inclusion of this method, a more broad ecosystem analysis should be done to ensure that the use of this method does not put domain

¹I always use the example.com suffix as this is a reserved domain available for example purposes. In real life, it would unlikely the victim and adversary-controlled domains would have the same registered domain part.

owners at an increased risk of attack. I argue that the usage patterns this method encourages actually reduce the attack surface domain owners. Note that the security considerations in this section only apply to domain owners that choose to use "DNS TXT Record with Static Value."

4.1 CA Subscriber Account Compromise Risk

One attack vector for this method that is often brought up is that since the value in DNS is persistent, a compromise of the subscriber's CA account could lead to a misissued certificate. However, **I argue that in the existing web PKI there is very little security in place against a compromised subscriber account and there are several techniques a compromised account could lead to misissued certificate even for domains that do not use "DNS TXT Record with Static Value"**.

Validation reuse is a common behavior in the baseline requirements and among deployed CAs. Most CAs have a window where if a subscriber has already validated a domain, a subsequent certificate requests from the same subscriber does not need to perform validation on that domain. If this window is taken into account with certificate transparency, an that has compromised a subscriber account can simply wait until it sees a legitimately issued certificate for the victim's domain, then request a certificate from the same CA using the compromised account, and then bypass domain control on the victim's domain. The only difference between this attack and the compromise of the account of a domain that is using "DNS TXT Record with Static Value" is that for domains that use "DNS TXT Record with Static Value" the adversary does not need to wait until the issuance of a benign certificate. However, with the increased push to shorter certificate lifetimes, this waiting period goes down, and the adversary does not have to use the subscriber account for any actions during the waiting period likely allowing the adversary to stay stealthy during this time.

Furthermore, subscribers are responsible for protecting the credentials to their CA accounts. In the case of an ACME client, these credentials are often co-located on machines that have access to certificate private keys. In summary, compromise of a subscriber account already causes significant vulnerabilities for domains regardless of which DCV method they use.

4.2 Offering Better Security to DNS

A primary motivation for including this method is to eliminate the need for organizations to engage in two potentially problematic behaviors that are currently required to complete DNS challenges:

- (1) Putting overly powerful DNS credentials on web infrastructure
- (2) Delegating management of “_acme-challenge” to a 3rd party organization

Regarding point 1, some ACME clients can complete dns-01 challenges using credentials from a domain’s DNS providers (<https://cert-manager.io/docs/releases/release-notes/release-notes-0.3/#new-acme-dns01-providers>). This gives the ACME client access to change any of a domain’s DNS records if these credentials are not scoped correctly. Ideally these credentials would have strictly limited scope allowing for only TXT records placed at “_acme-challenge” to be uploaded. However, it is not unlikely many domain owners will simply provide an unscoped DNS credential to the ACME software. Additionally, some DNS providers may not offer the required credential scoping.

Regarding point 2, an alternative to providing scoped DNS credentials is to delegate the management of the “_acme-challenge” domain to another system (e.g., ACME DNS <https://github.com/joohoi/acme-dns>). In theory this would be run in house as whatever system manages records at “_acme-challenge” is capable of obtaining certificates for the victim’s domain. However, given the complexity of running an entire additional DNS service just for the completion of dns-01 challenges, it is not unlikely many domain owner delegate such service to a 3rd party. ACME DNS previously offered such a service at <https://acme-dns.io/>. While convenient, such a service expands a domain’s attack surface for misissued certificates as delegates certificate issuance to an additional 3rd party.

I argue persistent DNS record uploads avoid both of these complications and offer superior security to domain owners.

4.3 Shorter Certificates/Better Automation

A trend in the CA/Browser Forum (particularly among certificate consumers) has been encouraging automation and shorter certificate lifetimes. I feel static DNS change goes a long way towards achieving this as the certificate issuance process is significantly easier. ACME clients or other certificate clients do not need to continually perform extra steps to complete challenges. These clients can have the much simpler job of requesting and deploying certificates. I feel organizations that previously used manual certificate issuance may be more inclined to move towards automated issuance (and thus potentially shorter certificate lifespans) given this increased convenience.

4.4 Increased use of HTTPS on Domains that are Only Accessible from Enterprise LANs

It is not unreasonable for an enterprise to place some services behind a firewall or access control list that only permits access to those services from the enterprise LAN or VPN. These domains should still be HTTPS to protect against attacks on plaintext traffic from threats that exist on that LAN. However, there are some challenges getting certificates deployed to these domains poses some challenges. HTTP validation and tls-alpn-01 validation are not possible because of the firewall rules in place. While DNS validation is an option, it requires DNS automation and proper credential scoping which presents additional technical complexity. There is a risk that some enterprises, in the prioritization of simplicity and up-time over security, are opting to not put HTTPS on these LAN-only endpoints. IoT is another context where such challenges may occur. Permitting this validation method may improve HTTPS adoption in these contexts and other scenarios where the complexity of obtaining digital certificates was previously an adoption barrier.

4.5 Improved Transparency

Certificate Transparency (CT) and CAA records have helped the global security community gain vastly improved visibility into the intents of domain owners and the behavior of CAs. This visibility allows for detection of misissuances by 3rd parties. This can help detect CA misbehavior as well as adversarial attacks on domains (which I have done myself using CT).

Ephemeral domain control validation hinders transparency into the validation process. Existing DNS Change validation can occur at any underscore-prefixed subdomain and even with ACME dns-01 validation, clients are instructed to take the validation tokens down after validation. However, public knowledge of domain’s intent to use a particular CA or permit validations for a specified time period allows for transparency of the domain control validation process. This could significantly help 3rd parties detect misissuances. While some argue it is a domain owners responsibility to detect misissuances, many enterprises have a security team that is responsible for the behavior of several domains managed by different departments or subcontractors. In an enterprise setting, knowledge of the inner workings and certificate issuance preferences of all those domains can be difficult and the improved transparency offered by “DNS TXT Record with Static Value” could be very beneficial.

Discrepancies between listed CAA records and certificate issuance behavior has already assisted in detecting a bug at a CA (https://bugzilla.mozilla.org/show_bug.cgi?id=

1951415). CAA records are only used by about 15% of domains. Increased adoption of “DNS TXT Record with Static Value” could significantly increase the percentage of domains publicly stating their CA intent. Increased transparency increases the likelihood the general public can detect misissuance.

5 Conclusion

In conclusion I feel:

- (1) The inclusion of “DNS TXT Record with Static Value” does not expand the attacks surface of DCV as any adversary capable of compromising this method can already obtain a certificate using several other existing methods.
- (2) The use of “DNS TXT Record with Static Value” despite the slightly increased damage of a potential

account compromise, has many attributes that are beneficial to the ecosystem and outweigh these concerns.

With this in mind, I take the (somewhat counterintuitive) stance that the inclusion of “DNS TXT Record with Static Value” improves the security of the web PKI ecosystem which is why I support it as an interested party.

For members of the CA/Browser Forum inclined to vote against this method due to the increased attack surface it poses to DCV, I encourage you to reconsider several existing DCV methods. For members of the CA/Browser Forum inclined to vote against this method due to increased damage of subscriber account compromise, I encourage you to reconsider the existing policies around validation reuse. Given the existing security properties of the web PKI as outlined in the TLS Server Certificate Baseline Requirements, I feel this method is a logical and beneficial inclusion.