Domain Verification Techniques

https://datatracker.ietf.org/doc/html/draft-sahib-domain-verification-techniques-02

Shivan Kaul Sahib (Brave Software)
Shumon Huque (Salesforce)

DNS Operations Working Group; IETF 111 Meeting
July 26 2021

What is domain verification?

Many providers on the internet need users to prove that they control a particular domain before granting them some sort of privilege associated with that domain.

For e.g. Let’s Encrypt has a DNS-based challenge for a user to prove that they control a particular domain (and hence should be issued a cert for it)

Survey of existing techniques

TXT-based

- “Please add this DNS TXT record with random value at the domain being verified to prove that you own this domain”
- Typically expires in a few days
- In practice, wide variation

Pattern: RDATA

```
bbc.com.  3599 IN   TXT
"atlassian-domain-verification=SQsgJ5h/FqwMTXuSG/G4Nd1Gx6uX2keREOsZSa22D5
XT46EsEuyaic8Aej4cR4Tr"

bbc.com.  3599 IN   TXT
"google-site-verification=yTRDtkD0tgHXSaJL0EtVrYGv1moNR-QkK8BAvjTv2Q8"
```

Pattern: name

Let’s Encrypt DNS TXT example

```
_acme-challenge.example.com.  IN  TXT "cE3A8qQpEzAIYq-T9DWNdLJ1_YRXamdxcjGTbzrOH5L"
```

GitHub DNS TXT example

```
_github-challenge-octocat.octocat.com  IN  TXT "9a6c10f4c4"
```

No pattern

```
bbc.com.  3599 IN   TXT  "1884df5221d841f294fd942e3e95a01f"
```

CNAME-based

- Fallback option
- Might be used if the domain name already has a CNAME
    - Since CNAMEs can’t coexist with other records (e.g. TXT) at the same domain name
- Point to a service provider property

Google Workspace CNAME example

```
3IBW7URVCRWY.example.com.  IN  CNAME
gv-LtgM1Qglw0JCE7mBVgLvM1DwuLGnuwzPCbsmXh3zjs4h6EWb8gy6domainverify.g
ooglehosted.com."
```

Recommendations

Targeted Domain Verification

1. Similar to what Let's Encrypt and GitHub do
2. Allows a service provider to get only the records they need
3. Putting all TXT records at the same name causes bloating
    a. Causing retries over TCP

Time-bound checking

1. When can the records be removed?
2. Should they exist in perpetuity?

DNSSEC

DNSSEC should be used to prevent DNS spoofing attacks from compromising domain verification.

- Domain owners should sign their zones.
- Verifiers should perform DNSSEC validation (e.g. by employing a validating DNS resolver service).

Not in draft yet...

1. Multiple vantage point checking if no DNSSEC (done by Let's Encrypt)
2. Should there be an IANA registry for _underscore prefixes?
3. Public suffix boundary
    a. Verifiers should not accept a request to verify a domain at or above a public suffix boundary.
4. Is the domain verification just for the domain or for everything underneath?
5. … do we need a new RR type? *ducks*

To consider

1. Do we want to adopt this draft?
2. Is Informational the right category for the draft?
3. Are there other topics that this draft needs to discuss?

Thanks!

Extra slides

Topics to cover

- Survey of existing domain verification techniques
- Plan to involve app folks (IETF Apps area working groups)
- Recommendations:
  - Secure Verification: DNSSEC signing & validation; (next best) multi-vantage point verification
  - Don’t collide at same name - reserved underscore names
  - Use application registry for _ names?
  - TXT vs CNAME vs new RR Type
  - Name vs Entire subtree rooted at name? Be clear about scope of verification
    - Certificate verification generally is only for the specific domain; but app verif ...
  - Use of Public Suffix List
  - Time boxing of verification records vs Long Lived records? What is our recommendation? And if both are allowed, verifications should clearly state requirements.
- Examples?
  - Let’s Encrypt, Atlassian/Google type example