Security Dispatch T. Fiebig
Internet-Draft TU Delft
Intended status: Informational K. Borgolte
Expires: March 12, 2020 Princeton University
September 09, 2019

Extended Security Considerations for the Automatic Certificate Management Environment (ESecACME)
draft-fiebig-security-acme-01

Abstract

Most Public Key Infrastructure X.509 (PKIX) certificates are issued via the ACME protocol. Recently, several attacks against domain validation (DV) have been published, including IP-use-after-free and (forced) on-path attacks. These attacks can often be mitigated by (selectively) requiring additional challenges, such as DNS validation, proof of ownership of a prior certificate, and by being more diligent in operating a certificate authority. This document provides a list of currently known attacks and describes mitigations and operational procedures to prevent issuing a certificate to an unauthorized party.

Table of Contents

1. Introduction

Today, most Public Key Infrastructure X.509 (PKIX) certificates are issued via the ACME protocol. The automated nature of ACME and its heavy use of domain validation (DV) render it susceptible to a variety of attacks. These attacks include IP-use-after-free

[CSTRIFE], (forced) on-path attacks [BAMBOO], and attacks on protocols used for validation [DVP], like DNS. In general, these attacks can be mitigated by (selectively) requiring additional challenges, e.g., validation of DNSSEC signatures, proof of ownership of a prior certificate, and by being more diligent when operating a certificate authority.

This document provides a list of known attacks and how they can be detected. It also describes mitigations and operational procedures that CAs should implement to reduce the threat of issuing a certificate to an unauthorized party. This section holds information on how these mitigations may impact the usability of CAs using ACME to issue certificates.

2. Attacks

In this section, we describe practical attacks against DV, how to detect them, and which additional verification methods should be used in case of an attack.

2.1. IP and Resource-use-after-free Attacks

IP and resource-use-after-free attacks occur if a domain owner points a DNS record to a resource, which they later vacate without deleting the corresponding DNS record. The resource, such as in cloud scenarios, could then be allocated by another party, thus, allowing an attacker to impersonate the owner.

For example, assuming that the web server for www.example.com is hosted on a virtual machine a cloud provider and the AAAA record of www.example.com points to the IPv6 address of that virtual machine, e.g., 2001:db8:1234:1234::1, and, when the operator terminates the virtual machine and frees the resource, they do not remove the DNS record. Then, it leads to a stale or dangling DNS record. If then another user of the cloud provider allocates a virtual machine, and receives the same IPv6 address (by luck or through other means), then this user could proof ownership of www.example.com to an ACME compliant CA.

These observations also hold for DNS records pointing to legacy IPv4 resources (A records), email servers in case of email-based ownership verification (MX records), SIP or other service delegations (SRV records), and even DNS delegations if DNSSEC is not being used (NS records). The attacks' feasibility is further increased by the fact that some validation challenges may validate a domain by verifying only one resource in case of multiple equivalent DNS records.

2.1.1. Detection

These attacks are difficult to detect from the CAs point of view, without domain owners taking additional precautions themselves. Techniques to detect the attack that CAs should use depend on the cooperation from domain owners.

Domain owners should use TLSA records to pin the TLS public key for a name, allowing the CA to verify the TLSA record to the key for which a certificate is requested.

A detection technique which does not require prior cooperation by domain owners leverages Certificate Transparency (CT) logs to identify certificates that were issued for the domain in the past, which can then be verified to still exist (thus, proving ownership of a previous certificate). Furthermore, CAA records can be used to restrict the number of CT logs that the processing CA needs to search. Additionally, if the processing CA is the only CA allowed to issue a certificate restricted through CAA records, then it may check that the certificate request is made by the same ACME account as prior successful certificate issuance requests.

2.1.2. Defense

If the TLSA public key and the public key used in making the certificate request do not match, then the CA must deny the requested certificate. In case of a preexisting certificate or a mismatch in the requesting ACME account, the operator must use additional validation techniques. If the domain has valid DNSSEC records, then a DNS challenge should be used. Alternatively, the CA should use a validation method that requires ownership of a previously issued certificate's key. Considering that NS and MX records may also suffer from resource-use-after-free attacks, unauthenticated DNS and email challenges must not be used.

Due to the inherent usability implications of the defense the CA may mitigate on high-risk resources only, such as known cloud providers or for operators with a high customer churn.

2.2. (Forced)-On-path Attacks

If an attacker can perform a man-in-the-middle (MitM) attack by controlling part of the network path between the CA and the resource used for validation, then they can also impersonate an operator and illegitimately obtain a certificate for a domain. Attackers may force this on-path situation, e.g., by using BGP shorter-prefix attacks [BAMBOO].

2.2.1. Detection

To detect on-path attacks, a CA should validate challenges from multiple vantage points. For this purpose, the CA must operate a geographically and topologically distributed system for verification. This system must contain at least one validator per IP region (AfriNIC, APNIC, ARIN, LACNIC, and RIPE). A CA monitor should also monitor the BGP prefix from which it the request originated, e.g., via a service similar to https://bgpmon.net [1]. However, note that, depending how close the attacker is to the victim on the network path, there may no path without malicious activity. Therefore, it generalizes the detection issue to that outlined for resource-use-after-free attacks.

2.2.2. Defense

The same defenses as for resource-use-after-free attacks apply. If an ongoing attack on a network prefix is detected, the CA must not issue certificates for the affected domains until the attack is over.

2.3. DNS Cache Poisoning Attacks

If an attacker is able to poison the DNS cache [CPOIS] of a CA while the CA validates a domain, then they may change the target of the DNS name to be authenticated. In turn, it allows the attacker to redirect the validation attempt to a host that they control. DNS cache poisoning may be successful regardless of [RFC5452] if the attacker can exploit packet fragmentation. By forcing a small on-path maximum transmission unit (MTU) between the CA's DNS resolver and a domain's authoritative DNS name server(s) using spoofed ICMP messages, an attacker may be able to fragment DNS responses. Correspondingly, by selecting the MTU so that fragmentation occurs after immediately the headers, an attacker can control the second part of a DNS packet, which then reassembles with with header of a benign packet. In an ideal scenario, it allows an attacker to overwrite the additional section of DNS responses, which the attacker could then use to change the content of an additional section for a MX, NS, CNAME, or any other type of record chain to point to a system unde the attacker's control.

2.3.1. Detection

A CA can identify that this attack takes place by measuring the MTU of inbound packets. If the MTU is smaller than 512 octets, the operator must assume that the domain is under attack.

2.3.2. Defense

To mitigate DNS cache poisoning attacks, the CA must validate DNSSEC, as already mandated by the CA browser forum [BFOR] for CAA records. If DNSSEC cannot be validated, then the CA's resolvers must ignore fragmented UDP packets with a UDP payload size of less than 512 octets.

The CA may require DNSSEC validation to succeed and TLSA records to be in place for name servers of domains that require a MTU below 1000 octets. The CA may also opt to enforce DNS-over-TCP [RFC7766], DNS-over-TLS [RFC7858], or DNS-over-HTTPS [RFC8484].

As the additional section of incoming answers at the end of a DNS response is particularly vulnerable to this attack, the CA's resolvers must not use data from the additional section, but resolve all names themselves.

2.4. DNS Cache Staleness Attacks

An attacker can execute an attack similar to resource-use-after-free attacks if a CA's DNS resolver caches a DNS record although the benign party may have updated the corresponding record. Then, the CA's resolver might serve the cached record to the validation systems. If the CA's resolver implements draft-ietf-dnsop-serve-stale, then an attacker has an even longer window of opportunity. This window can even be extended by launching a Denial-of-Service (DoS) attack on a domain's authoritative name servers, in which case the CA's resolver may serve a stale cached record with an expired TTL for up to a week.

2.4.1. Detection

For a CA, these attacks are not distinguishable from legitimate errors and downtimes.

2.4.2. Defense

To prevent DNS cache attacks, the CA's validation system's DNS recursor must not serve cached records, and it must not implement draft-ietf-dnsop-serve-stale. If an authoritative server is unreachable, a certificate must not be issued.

3. Summary of CA Operational Improvements

In this section, we summarize the operational changes and mechanisms to reduce the chance of issuing a certificate to an unauthorized party.

3.1. Hardening Against Attacks Without DNS Control

If the validation target for a challenge (A/AAAA/NS/MX) is considered at-risk or located in a network with a high resource churn, e.g., a cloud provider or a residential ISP, then the CA should require the domain for which the certificate is to be issued to be DNSSEC signed, as well as a CAA and TLSA record to be present. If the domain is not DNSSEC signed, or there is a mismatch between the TLSA record, then the CA should consider the domain under attack and must not issue a certificate.

If the CA can identify a certificate has been issued for the same name before, it may consider requiring a challenge proving ownership of the identified certificate, or a DNSSEC signed DNS challenge.

3.2. Multi-Vantage Point Validation

A CA should validate challenges from more than one network vantage point. They should validate from at least three distinct geographical and topological locations. If at least one of the CA's validation nodes does not match the results of the other nodes, then the CA must consider the requested domain to be under attack and must not issue a certificate.

3.3. BGP Monitoring

A CA should monitor the current state of the BGP ecosystem, e.g., by using a service similar to https://bgpmon.net [2]. If any network prefix for the A, AAAA, MX, or NS records (or intermediate names and CNAMEs) is considered to be under a BGP MitM attack, then the CA must consider the requested domain to be under attack, and must not issue a certificate.

3.4. DNS Resolver Configuration and Monitoring

To mitigate DNS fragmentation attacks, a CA's DNS resolvers should ignore fragmented packets with UDP payload below 512 octets. If a CA encounters UDP fragments of less than 1000 octets, it may require DNSSEC and TLSA records to be presented and validated for the zone before issuing a certificate. The CA's resolvers must not trust the additional section of DNS responses and resolve all names on their own.

To prevent attacks relying on stale DNS records, CAs must not utilize draft-ietf-dnsop-serve-stale on their recursors. In fact, resolvers must not serve records from their cache to the validation system. If the authoritative DNS servers of a domain are unreachable, then the CA must not issue a certificate.

3.5. DNSSEC Validation Failure and Lack of DNSSEC

If DNSSEC validation for a domain for which a certificate is requested fails, the CA must consider the domain to be under attack, and must not issue a certificate until DNSSEC validation is successful. Depending on whether the domain is considered at-risk, the CA may decide to not issue a certificate in the absence of DNSSEC or CAA records.

3.6. Recent Domain Transfer

If a domain has been transfered within the last 72 hours, a CA may consider the domain's state of ownership as insufficiently defined. It may require proof of ownership of a prior certificate, or the zone to be DNSSEC signed, and TLSA as well as CAA records to be present before issuing a certificate.

4. Additional Validation Options

4.1. Proof of Ownership of a Prior Certificate

If a CA detects an attack, it MAY require the requesting party to prove that it has access to the private key for a previously issued certificate. This can be done implicitly by requiring validation over HTTPS, using a validating prior certificate, or explicitly by using a dedicated challenge.

4.1.1. Limitations

This option has several operational challenges. An domain owner's infrastructure may not be design in a way that preserves prior private keys, for example in large container setups. Similarly, the prior key might have been lost due to data loss. Additionally, prior certificates may have expired.

An attacker may have also obtained a prior private key by compromising a system, or by having had legitimate authority over the domain before.

5. IANA Considerations

There are no IANA considerations.

6. Security Considerations

This document itself serves as a summary of additional security considerations. CA operators should carefully follow the

recommendations made in this document to prevent issuing certificates to unauthorized parties.

7. Acknowledgements

8. References

8.1. Normative References

[BAMBOO] Birge-Lee, H., Sun, Y., Edmundson, A., Rexford, J., and P. Mittal, "Bamboozling Certificate Authorities with BGP", August 2018, <https://www.usenix.org/conference/usenixsecurity18/presentation/birge-lee>.

[BFOR] CA/Browser Forum, ., "Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates", October 2018, <https://cabforum.org/wp-content/uploads/CA-Browser-Forum-BR-1.6.1.pdf>.

[CPOIS] IOActive Inc, ., "Black ops 2008: It's the end of the cache as we know it", 2008, <http://kurser.lobner.dk/dDist/DMK_BO2K8.pdf>.

[CSTRIFE] Borgolte, K., Fiebig, T., Hao, S., Kruegel, C., and G. Vigna, "Cloud Strife: Mitigating the Security Risks of Domain-Validated Certificates", February 2018, <http://dx.doi.org/10.14722/ndss.2018.23327>.

[DVP] Brandt, M., Dai, T., Klein, A., Schulman, H., and M. Waidner, "Domain Validation++ For MitM-Resilient PKI", n.d., <https://doi.org/10.1145/3243734.3243790>.

[RFC5452] Hubert, A. and R. van Mook, "Measures for Making DNS More Resilient against Forged Answers", RFC 5452, DOI 10.17487/RFC5452, January 2009, <https://www.rfc-editor.org/info/rfc5452>.

[RFC7766] Dickinson, J., Dickinson, S., Bellis, R., Mankin, A., and D. Wessels, "DNS Transport over TCP - Implementation Requirements", RFC 7766, DOI 10.17487/RFC7766, March 2016, <https://www.rfc-editor.org/info/rfc7766>.

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/info/rfc7858>.

[RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <https://www.rfc-editor.org/info/rfc8484>.

8.2. URIs

[1] https://bgpmon.net

[2] https://bgpmon.net

Authors' Addresses

Tobias Fiebig
TU Delft

Email: t.fiebig@tudelft.nl

Kevin Borgolte
Princeton University

Email: kevin@iseclab.org