MMLAB Main Seminar

Revocation Speedrun: How the WebPKI Copes with Fraudulent Certificates

Jens Friess[1,2,3] , Haya Schulmann[1,3,4] , Michael Waidner[1,2,3]
1 ATHENE, Germany 2 TU Darmstadt, Germany
3 Fraunhofer SIT, Germany 4 Goethe-Universität Frankfurt, Germany

CoNEXT 2023

Hyunsoo Kim (hskim@mmlab.snu.ac.kr)

2024. 01. 25

Why Do We Need to Speed Up Revocation?

- The security of the Public Key Infrastructure (PKI) relies on the trusted operations of Certificate Authorities (CAs)
- Unfortunately, real-world CA operations often fall short of ideal, perfectly-managed certificate issuance
  - Downgrade attacks on Let's Encrypt [CCS 2021]
  - CAs operational issue, bugs in automated software
- Revocation as Damage control → Time is critical
  - Mitigating Man-in-the-Middle attacks
  - Efforts in dismantling phishing sites
- Assessing the revocation system's efficacy begins with measuring reaction delays

In this paper

1. Detection of Fraudulent Certificates

➔ Assess the detection speed for fraudulent certificates

2. Certificate Revocation by CAs

➔ Evaluate CAs' response time to administrative revocation requests from domain owners

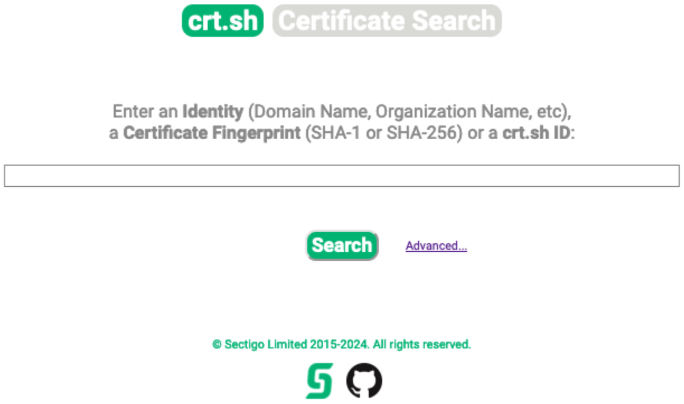
3. Client-Side Revocation Checks

➔ Conduct initial real-world measurements of revocation checks and compare them to lab results

- First comprehensive end-to-end analysis of the revocation system's performance

Detection of Fraudulent Certificates

- **Certificate Transparency (CT) logs**
  - Domain owners monitor certificate issuances via public APIs
  - However, CT logs lack domain name indexing, necessitating comprehensive scans, which demand significant storage and bandwidth
- **Third-party CT Monitors**
  - Index certificates by domain after scanning CT logs
  - Offer search capabilities and email notifications



Enter an **Identity** (Domain Name, Organization Name, etc),
a **Certificate Fingerprint** (SHA-1 or SHA-256) or a **crt.sh ID**:

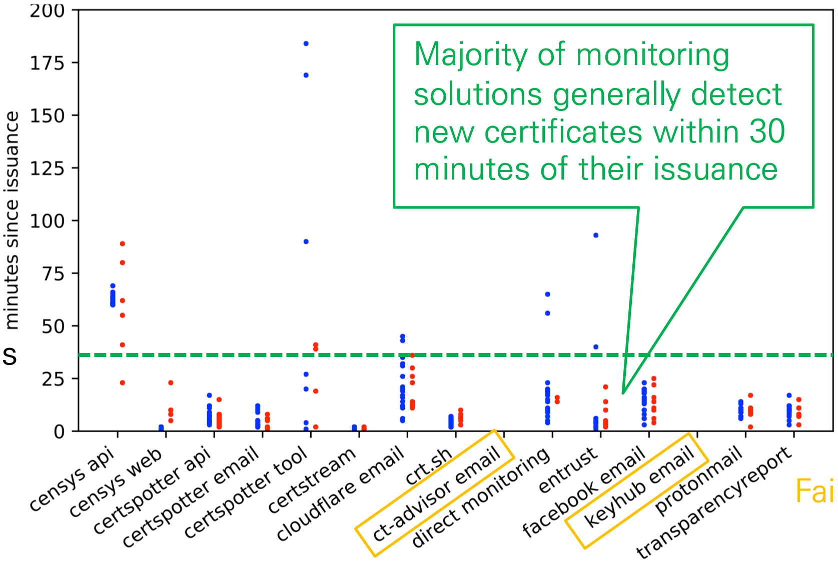
© Sectigo Limited 2015-2024. All rights reserved.

Services: 3.4B IPv4 Hosts: 254.4M IPv6 Hosts: 175.8M Virtual Hosts: 1.2B

Detection of Fraudulent Certificates

1. **Issue various certificates from multiple CAs and track the notification speed of each monitor**
   - Measure the interval from domain validation (DV) completion to each monitor's notification
2. **Issue a rogue certificate for each domain using the same respective CA**
   - Utilize distinct accounts to purchase certificates and complete DV from various IP addresses

Fraudulent and legitimate certificates are not fundamentally differentiable based on detection speed



Majority of monitoring solutions generally detect new certificates within 30 minutes of their issuance

Failed to notify at all

Certificate Revocation by CAs

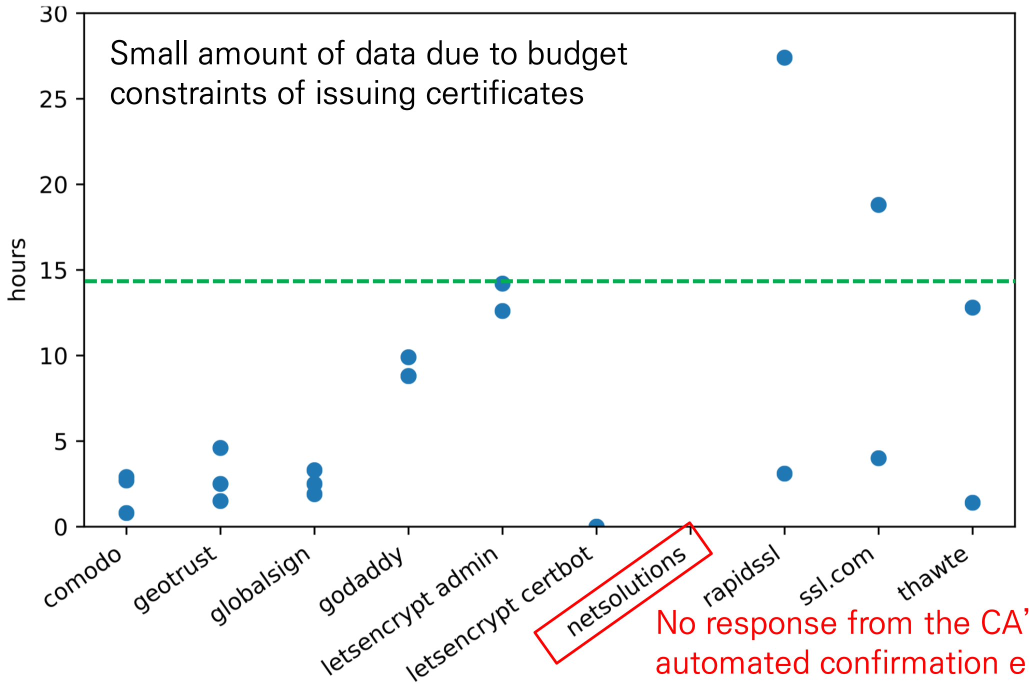
- **Domain owners must contact the CA to revoke detected fraudulent certificates**
  - Without the account or private key of the fraudulent certificate, the domain owner needs to request an administrative revocation

1. **Revocation is requested through email or an online portal**
   - Emails from administrative addresses (e.g., admin, postmaster) typically influence the process. However, all CAs except GoDaddy were unaffected, allowing room for spoofed requests
2. **CAs mandate a domain control challenge**
   - DV certificates involve a DNS TXT-based challenge; successful verification leads to revocation

➔ Track the time from the initial revocation request to the OCSP revocation timestamp

Certificate Revocation by CAs

- **Median: 3.18 hours / Average: 6.5 hours**
- **Possible reasons for high variability of these delays**
  - Propagation of the DNS TXT records created to complete the domain control challenges
  - Workload of the employee at the time of each measurement and the CA's prioritization of incoming revocation requests



Small amount of data due to budget constraints of issuing certificates

90% of revocations were completed within 14.2 hours

No response from the CA's support address, beyond automated confirmation emails → HUGE PROBLEM

Client-Side Revocation Checks in the Lab

- Assess how popular OS and browser combinations respond to revoked certificates
- Discovered OS-level caching of revocation information
  1. Accessed a revoked-certificate site on Windows via Edge or Internet Explorer, using OCSP/CRL
  2. Accessed the site for the first time on Firefox and Chrome without OCSP/CRL access
  3. Firefox and Chrome displayed a warning sign

→ Used a VM to isolate browsers and reset the state to prevent OS-level interaction

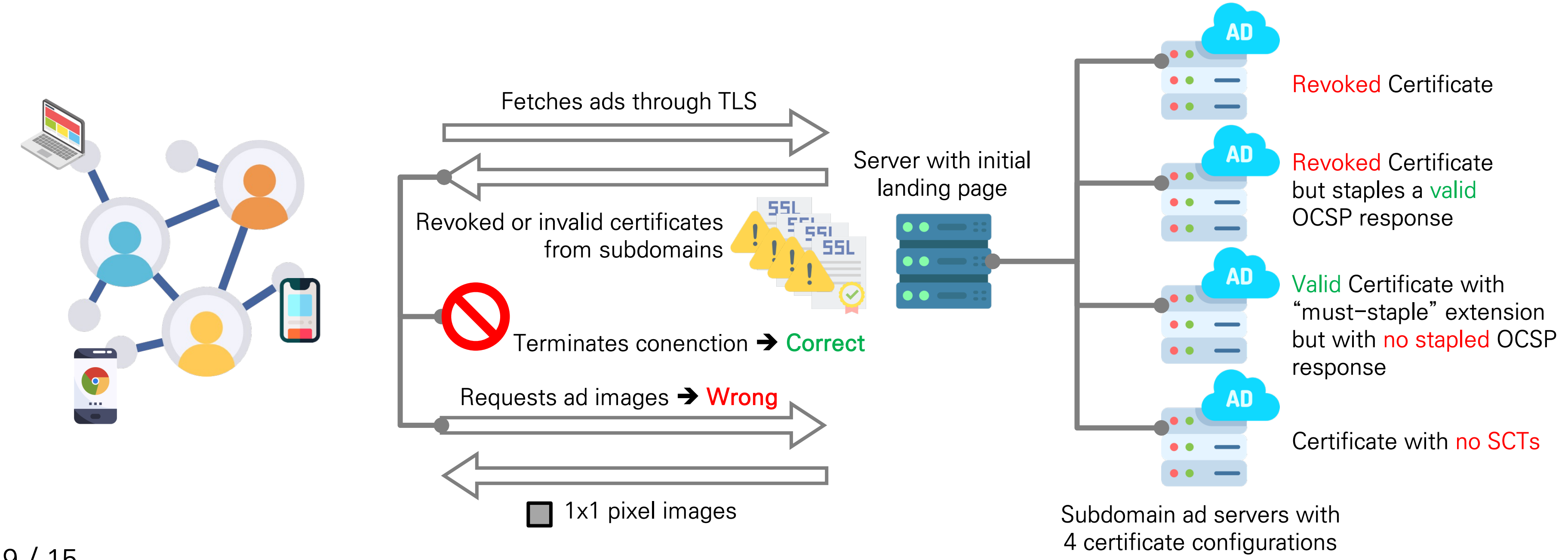
- Browsers consistently soft-fail if OCSP and CRLs are inaccessible
- Furthermore, this soft-fail caching results in certificates being accepted even after revocation endpoints become available again

Client-Side Revocation Checks in the Lab

OS	Browser	OCSP/CRL endpoints Available	OCSP/CRL endpoints Blocked	OCSP/CRL endpoints Blocked -> Available	OCSP/CRL endpoints Blocked -> clear cache -> Available
Ubuntu 20.04	Chromium 90	X	X	X	X
	Firefox 88	O	X	X	O
	Brave 1.24	X	X	X	X
	Opera 76	X	X	X	X
Windows 10	Chrome 90	X	X	X	X
	Firefox 88	O	X	X	O
	Brave 1.24	X	X	X	X
	Opera 76	X	X	X	X
	Edge 90	O	X	O	O
	IE 11	O	X	O	O
Mac OS 11.3	Safari 14	O	X	X	O
	Chrome 90	O	X	X	O
	Firefox 88	O	X	X	X
	Brave 1.24	O	X	X	X
	Opera 76	O	X	X	X
Android 11	Chrome 90	X	X	X	X
	Firefox 88	X	X	X	X
	DuckDuckGo 5.80	X	X	X	X
iOS 14.5	Safari 14	O	X	X	O

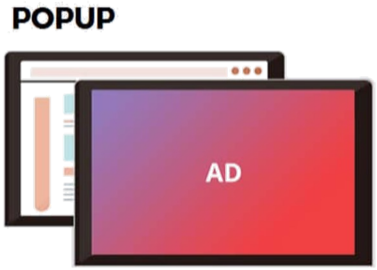
Revocation Checking in the Wild – Methodology

- **Live measurements using an advertising network to determine which actual end-users are vulnerable to revoked certificates**
  - Minimal network/storage load by using 1x1 image
  - Collect only client IP address and user agent info
- **Problematic certificates are sent during TLS handshakes for requesting ad images → percentage of successful TLS handshakes**

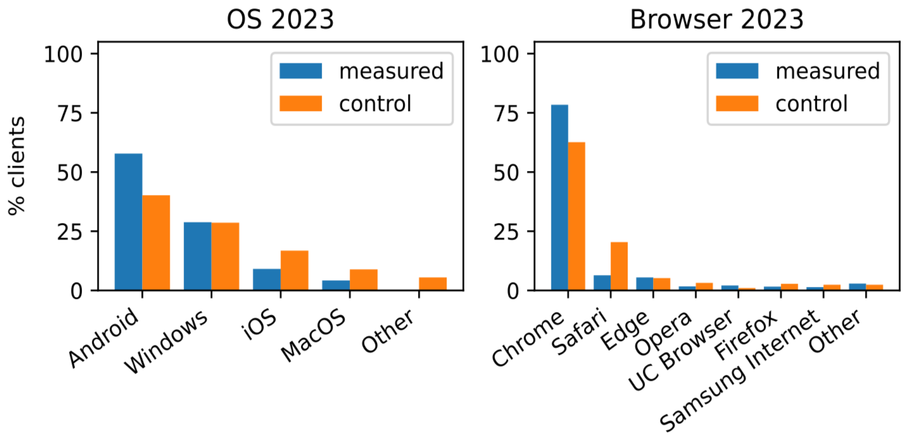

Fetches ads through TLS
Revoked or invalid certificates from subdomains
SSL
Terminates conenction → Correct
Requests ad images → Wrong
1x1 pixel images
Server with initial landing page
AD
Revoked Certificate
Revoked Certificate but staples a valid OCSP response
Valid Certificate with "must-staple" extension but with no stapled OCSP response
Certificate with no SCTs
Subdomain ad servers with 4 certificate configurations

Revocation Checking in the Wild – Methodology

- Three separate campaigns based on continents with equal budget
- “Pop-under” ads
  - Open in the background
  - More likely to remain open long enough to trigger all ad requests
- “Untargeted” ads to achieve random sampling of clients
  - Published sites were chosen by the adnet → possible bias
  - Measured data was close to known OSes and browsers market share



POPUNDER
AD

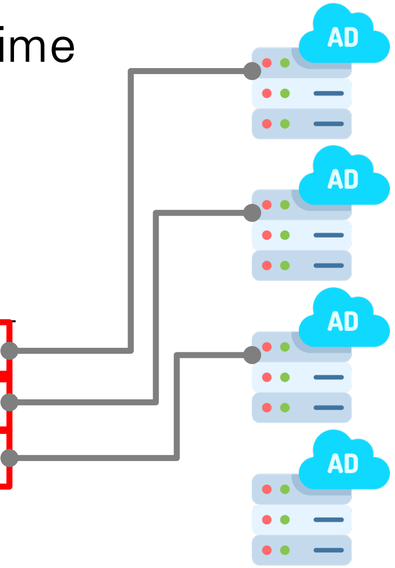


Revocation Checking in the Wild – Results

- Majority of clients do not check revocation at all
- Stapling cached valid OCSP response increases the chance of accepting a revoked certificate
    - Still, some clients ignores OCSP stapling and performs realtime revocation checking + older clients with no OCSP stapling support
- Most clients disregard the “must-staple” extension
    - “must-staple” is often discussed as a prime candidate for improving revocation

	2022	2023
clients	529575	285543
unique IPs	369302	281066
countries	226	194

	2022	2023
rev	87.4%	82.8%
sta	90.3%	85.5%
mus	89.5%	88.3%
sct	70.4%	31.2%



Revoked Certificate

Revoked Certificate but staples a valid OCSP response

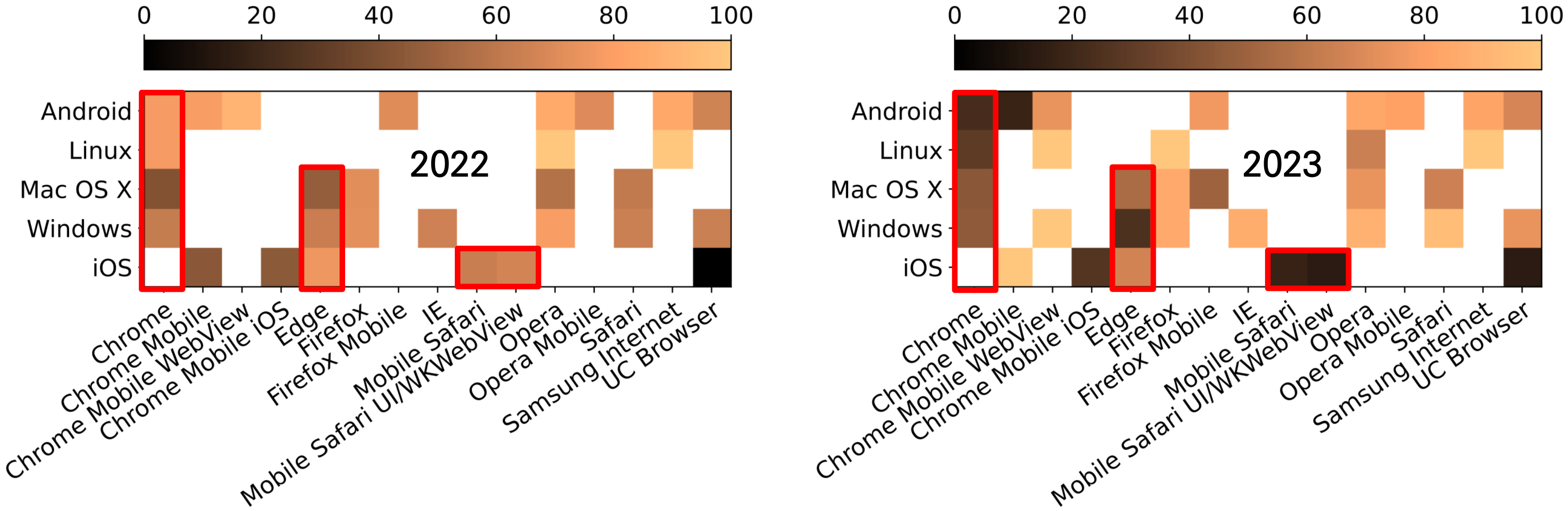
Valid Certificate with “must-staple” extension but with no stapled OCSP response

Certificate with no SCTs

Revocation Checking in the Wild – Results

	2022	2023
rev	87.4%	82.8%
sta	90.3%	85.5%
mus	89.5%	88.3%
sct	70.4%	31.2%

- Clients ignoring the absence of SCTs 70.4% → 31.2%
  - Increased enforcement by Chrome across all platforms
  - Similarly Mobile Safari, Edge
- Due to browsers declining certificates with no SCTs
  ➔ CAs are incentivized to log all their certificates to the CT



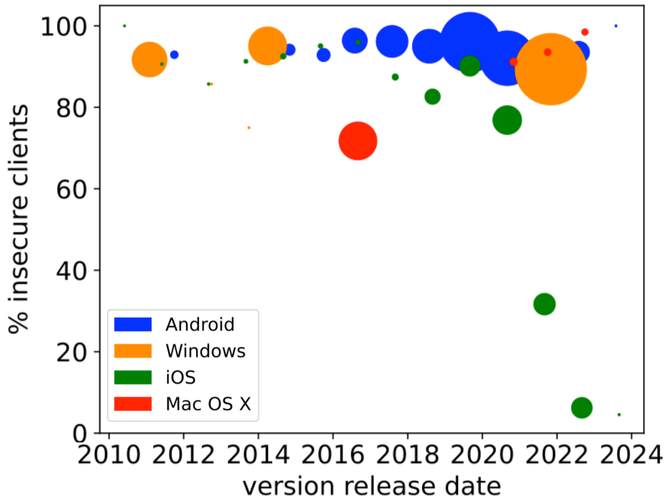
Percentage of accepting a certificate without SCTs

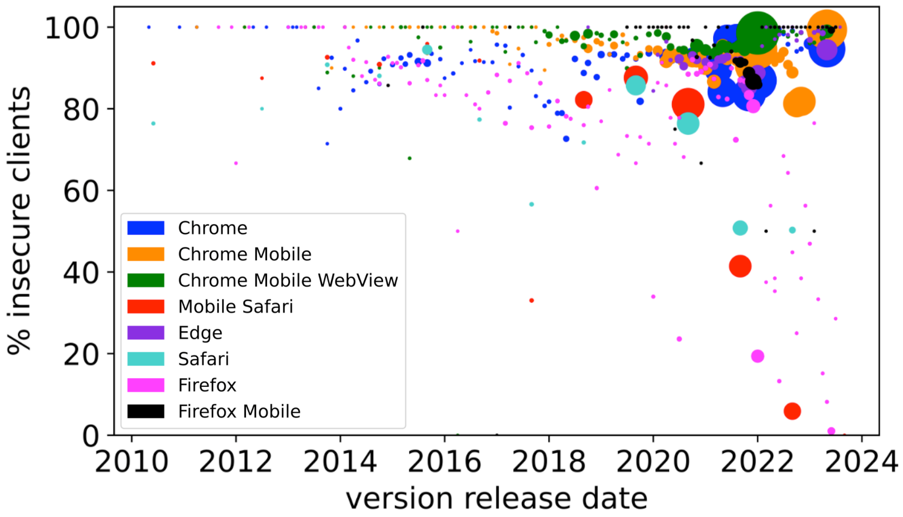
Revocation Checking in the Wild – Results

- Difference between lab and wild results
  - For example, iOS should decline all revoked certificates
  - However a significant fraction of iOS clients accepted revoked certificates

➔ Comparison of client versions showed increase enforcement trend starting from 2020

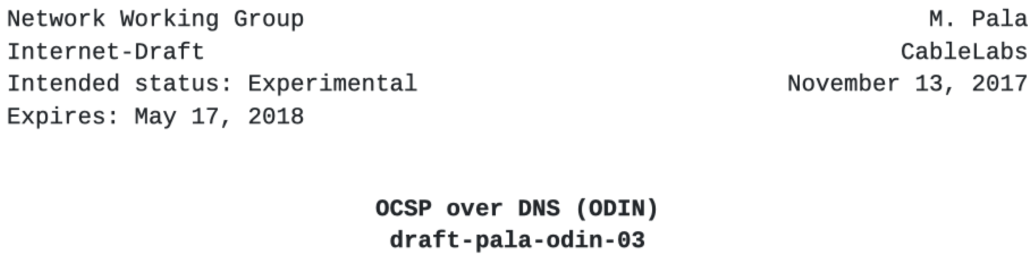
- We still see both the presence and absence of revocation checks





Discussions

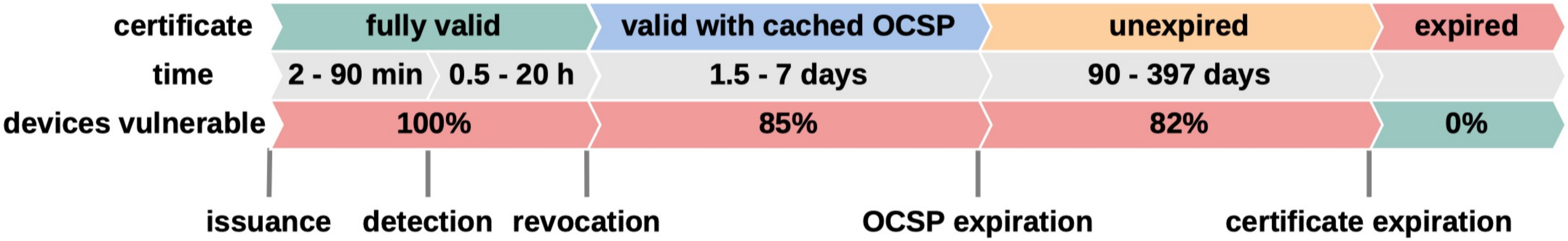
- **Advocates for shortening the validity period of stapled OCSP response**
  - Reliable and fast delivery of revocation information
    = Availability of robust, performant, DDoS-resilient OCSP responders
  - CAs need to balance responder load with the shortest viable OCSP response validity
- **Revocation checks via DNS-based delivery**
  - OCSP over DNS (ODIN): An IETF draft expired in May 2018
  - "An up-to-date certificate status is as important to a TLS-based Internet as an up-to-date IP address"



```
Network Working Group                                    M. Pala
Internet-Draft                                         CableLabs
Intended status: Experimental                  November 13, 2017
Expires: May 17, 2018


                         OCSP over DNS (ODIN)
                          draft-pala-odin-03
```

Conclusion and Critiques



certificate: fully valid | valid with cached OCSP | unexpired | expired

time: 2 - 90 min | 0.5 - 20 h | 1.5 - 7 days | 90 - 397 days

devices vulnerable: 100% | 85% | 82% | 0%

issuance, detection, revocation, OCSP expiration, certificate expiration

- Certificate revocation by CAs are already too slow; fully automated solutions are necessary
  - Ideal goal is to make detection time equal to revocation time

- CAs lack incentives for quick and reliable revocation information delivery. Domain owners must proactively disseminate revocation details via alternative channels

감사합니다

Thank you~!