(Probleme beim) Widerruf von Zertifikaten für RADIUS-Server

Jan-Frederik Rieckers

stud. Mitarbeiter am Zentrum für Netze der Universität Bremen

71. DFN-Betriebstagung, 24.–25. September 2019

Kontext

- Login ins eduroam geschieht meist mittels EAP-TTLS oder EAP-PEAP
- Beide Verfahren basieren auf EAP-TLS (RFC 5216)
- EAP-TLS nutzt X509-Zertifikate für Identitätsschutz und Verschlüsselung

Grundsätzliches Problem

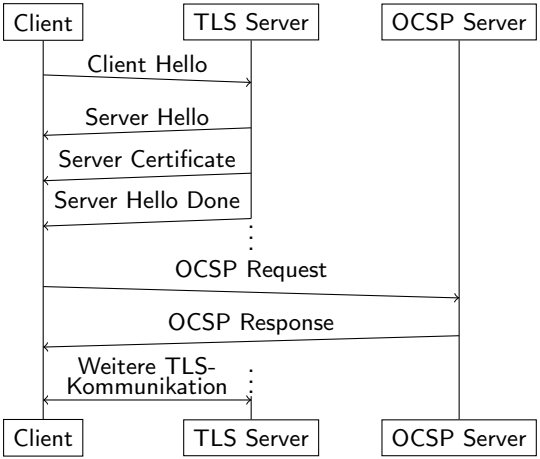
- Digitale Zertifikate sind (ohne weitere Maßnahmen) nach Ausstellung unwiderrufbar und während des Gültigkeitszeitraum uneingeschränkt gültig
- Private Keys können gestohlen/kompromittiert werden $\rightarrow$ Zertifikate müssen widerrufbar sein
- PKI stellt Verfahren zum Widerruf bereit, CAs müssen entsprechende Infrastruktur betreiben
- Worst-Case-Annahme für diesen Vortrag: Der private Schlüssel eines RADIUS-Servers wird gestohlen, Angreifer strahlt eduroam mit gestohlenem Zertifikat auf dem Campus aus, um Zugangsdaten abzugreifen

Zertifikatswiderruf

- Veröffentlichung von Zertifikatswiderruf auf zwei Arten möglich:
- Certificate Revocation Lists (CRL)
  - Von CA signierte Liste aller widerrufenen Zertifkate
  - Listen werden lang, müssen im Voraus heruntergeladen werden
  - Jede (Sub-)CA hat eigene CRL, Sub-CA bei Verbindungsaufbau aber evtl. noch gar nicht bekannt
- Online Certificate Status Protocol (OCSP)

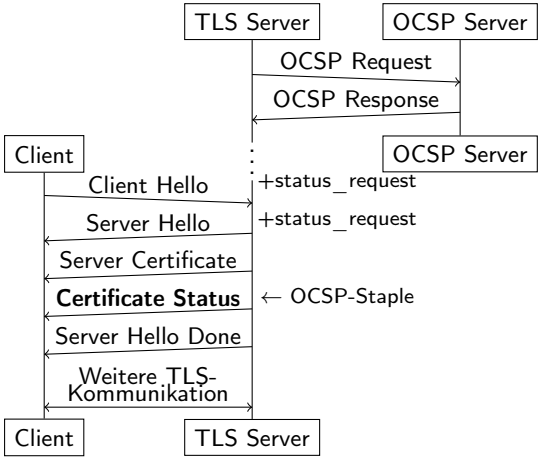
Online Certificate Status Protocol (OCSP)

- Live-Abfrage beim OCSP-Server der CA, ob ein Zertifikat gültig ist
- URL zum OCSP-Server in Zertifikat enthalten
- Verbindung zum OCSP-Server wird während des Zertifikatchecks aufgebaut
- Problem: Beim Einloggen ins eduroam besteht noch keine Internet-Verbindung



OCSP-Stapling[1]

- OCSP-Response wird vom TLS-Server im Voraus geholt (sog. Staple)
- TLS-Server schickt Staple als Gültigkeitsbeweis im TLS Server Hello mit
- Client muss im Client Hello `status_request` Extension hinzufügen
- Löst Online-Problem



[1] Beispiel basiert auf TLS $\leq$ 1.2, TLS 1.3 setzt OCSP-Stapling etwas anders um

OCSP-Stapling – Die Antwort auf alle Fragen?

- OCSP-Stapling grundsätzlich sinnvoll (auch im Web)
- Keine weitere Verbindung nötig, reduziert Roundtrips
- OCSP-Stapling soll laut RFC 5216 (EAP-TLS) implementiert sein
  Section 5.4: „`[...] EAP-TLS peers and servers SHOULD implement Certificate Status Request messages [...]`“

- Sicherheitsgewinn nur marginal:
  - Angreifer würde keinen negativen Staple mitschicken
  - Stapling ist nicht verpflichtend

OCSP MustStaple

- Staples müssen „erzwungen“ werden, ohne Backwards-Compatibility zu brechen
- Lösung: Im Zertifikat festschreiben, dass der Server OCSP-Stapling unterstützen muss (X509-Extension „TLS Feature Extension“, RFC 7633)
- Clients, die OCSP-Stapling unterstützen, werden Verbindung abbrechen, wenn Server kein OCSP-Stapling anbietet
- Keine Sicherheit bei fehlender Namensüberprüfung $\rightarrow$ Angreifer könnte mit anderem Zertifikat ohne MustStaple weiterhin Zugangsdaten abgreifen

- DFN-PKI stellt MustStaple-Zertifikate aus, von TN-Service auswählbar

OCSP-Stapling per Konfiguration erzwingen

- Zweite Möglichkeit, Sicherheit zu verbessern: Stapling unabhängig vom Zertifikat erzwingen
- Clients werden Verbindung abbrechen, wenn Server kein Stapling anbietet
- Auch bei fehlender/unvollständiger Namensüberprüfung noch (Teil-)Sicherheit gegeben

- Nachteil bei beiden Varianten: Abhängigkeit vom OCSP-Responder der CA

OCSP – Implementierungsstand[2]

- Realität: Server unterstützen (in aktuellen Stable-Versionen) kein OCSP-Stapling, Clients fragen z.T. kein OCSP-Stapling an, ignorieren MustStaple

[2] Bezieht sich nur auf die TLS-Implementierungen der WLAN-Treiber

Implementierungsstand - Serverseitig

Freeradius v3

- unterstützt OCSP für Client-Zertifikate
- keine Unterstützung für OCSP-Stapling

Freeradius v4 (noch in Entwicklung)

- Unterstützung für OCSP-Stapling implementiert

Cisco ISE *(nur aufgrund von Online-Recherche)*

- Wie Freeradius v3: Unterstützung für OCSP, kein OCSP-Stapling

Radiator *(nur aufgrund von Online-Recherche)*

- OCSP-Stapling seit v4.20 (Feb. '18) verfügbar

Implementierungsstand - Clientseitig

Apple-Geräte

- Implementierung von OCSP-Stapling
  - `status_request` im ClientHello vorhanden
- Keine Implementierung von OCSP MustStaple-Extension in Zertifikaten
- Verpflichtendes OCSP-Stapling auf den ersten Blick nicht konfigurierbar

Implementierungsstand - Clientseitig

wpasupplicant *(genutzt von Linux und Android)*

- Implementierung von OCSP-Stapling
- Standardmäßig abgeschaltet
- Keine Implementierung von MustStaple-Extension
- OCSP-Stapling per Konfiguration erzwingbar (in `wpa_supplicant.conf`, nicht in GUI-Frontends wie z.B. NetworkManager)

```
network {
  ssid=eduroam
  [...]
  ocsp=2
}
```

Implementierungsstand - Clientseitig

Windows-Geräte

- Augenscheinlich keine Nutzung von OCSP-Stapling

Resümee – Teil 1

- Aktuell ist Zertifikatswiderruf für Radius-Zertifikate effektiv nicht möglich
- Realer Sicherheitsgewinn würde nur existieren bei:
  - MustStaple-Zertifikate
    - Bei fehlender Namensüberprüfung Missbrauch von gestohlenen Zertifikaten der gleichen CA immer noch möglich
    - MustStaple-Extension nicht implementiert
  - Clients konfigurieren, OCSP-Stapling zu erfordern.
    - Auch bei fehlender Namensüberprüfung noch (Teil-)Sicherheit gegeben
    - Verpflichtendes OCSP-Stapling nur bei wpasupplicant konfigurierbar
- Beide Möglichkeiten bringen Abhängigkeit zu OCSP-Responder der CA
- Alle Maßnahmen undurchführbar, solange Server-Software kein OCSP-Stapling unterstützt
- → Weiteres Beispiel, weshalb PKI im eduroam-Kontext problematisch ist

Resümee – Teil 2

Kurzfristige und langfristige Gegenmaßnahmen

Mit EAP-TTLS/PEAP

- Getrennte Zugangsdaten für Account und WLAN
- Kein PAP als Phase2
  - Bei PAP ist TLS einzige Sicherheit, aber clientseitige Zertifikatsüberprüfung problematisch (q.e.d.)
  - besser MSCHAPv2 (Nachteil: NT-Passwort (MD4) in der Datenbank)

Ohne EAP-TTLS/PEAP

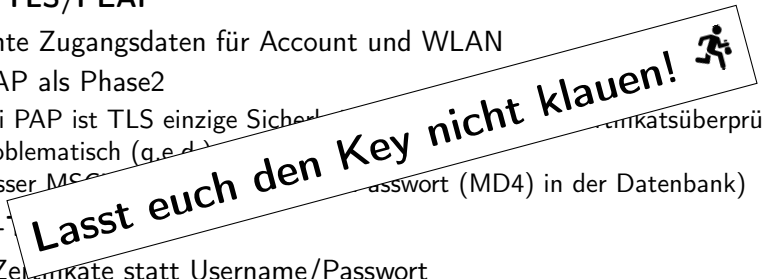
- Client-Zertifikate statt Username/Passwort
- Zero-Knowledge-Verfahren ohne PKI (z. B. EAP-PWD, Nachteil: keine anonyme Identität, aktuell noch Passwort im Klartext in der Datenbank)

Resümee – Teil 2

Kurzfristige und langfristige Gegenmaßnahmen

Mit EAP-TTLS/PEAP

- Getrennte Zugangsdaten für Account und WLAN
- Kein PAP als Phase2
  - Bei PAP ist TLS einzige Sicher[illegible] ...rtifikatsüberprüfung problematisch (q.e.d.)
  - besser MS[illegible] ...asswort (MD4) in der Datenbank)



Ohne EAP-T[illegible]

- Client-Zertifikate statt Username/Passwort
- Zero-Knowledge-Verfahren ohne PKI (z. B. EAP-PWD, Nachteil: keine anonyme Identität, aktuell noch Passwort im Klartext in der Datenbank)

Ausblick

Ausblick/Geplante weitere Arbeit:

- Weitere Tests mit Freeradius 4 und MustStaple-Zertifikaten
- Implementierung der X509-MustStaple-Extension in wpasupplicant

Fragen?

Kontakt:
rieckers@uni-bremen.de

Bildquellen: Icons by Gregor Cresnar from Flaticon