tiny-acme-server: aufwandsarmes Ausstellen automatisierter Zertifikate per http-01 Challenge

25. März 2025

Simon Ruderich

Regionales Rechenzentrum Erlangen (RRZE)
Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU)

Inhalt

ACME

ACME

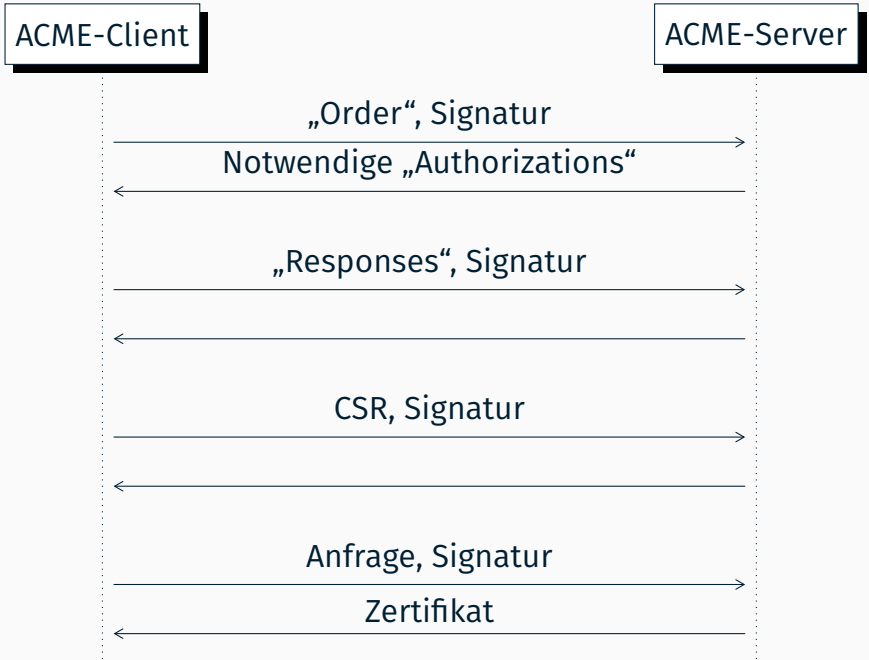
- Automatic Certificate Management Environment
- RFC 8555 (2019)
- Automatisiertes Ausstellen von Zertifikaten

ACME

- Automatic Certificate Management Environment
- RFC 8555 (2019)
- Automatisiertes Ausstellen von Zertifikaten
- Nutzt JSON Web Signatures (RFC 7515)
  - Alle Nachrichten vom Client sind signiert
  - Zertifikate an jeweils einen Account gebunden
- Für alle angefragten Domains muss der Besitz per Challenge bestätigt werden

Ablauf: Accounterstellung

Ablauf: Zertifikatsausstellung



http-01 Challenge

- Challenge wird für jede Domain durchgeführt
- Client schreibt Token plus Account-key „Thumbprint“ nach `/.well-known/acme-challenge/<token>`
- Server verbindet sich zur Domain per HTTP (!) und prüft Challenge (Redirects werden verfolgt, ungültiges HTTPS ist okay)

Hintergründe

Situation

- Migration Sectigo $\rightarrow$ HARICA

Situation

- Migration Sectigo $\rightarrow$ HARICA
- Relativ große Uni mit „historisch gewachsenen“ Strukturen: keine automatisierbare Zuordnung Einrichtung zu Domains
- DNS zentral verwaltet
- Viele Lehrstühle nutzen Let’s Encrypt

Situation

- Migration Sectigo → HARICA
- Relativ große Uni mit „historisch gewachsenen“ Strukturen:
  keine automatisierbare Zuordnung Einrichtung zu Domains
- DNS zentral verwaltet
- Viele Lehrstühle nutzen Let’s Encrypt
- Bisher: Manuelle Bestätigung der Sectigo Zertifikate (via Tool und API)
- Wenig Automatisierung auf der Serverseite (Ausnahme Webauftritte)

Anforderungen

- Minimaler Aufwand bei weiteren Anbieterwechseln
- Vollständige Automatisierung; kurze Zertifikatslaufzeiten
- Zertifikate für private IP-Adressen
- Kein extra Administrationsaufwand
- Wenige Konfigurationsaufwand für Nutzer

Let’s Encrypt?

- Rate-Limits sind problematisch
- Private IP-Adressen braucht dns-01
- Damit Zuordnung der Domains nötig

tiny-acme-server

Software

- `https://gitos.rrze.fau.de/noc/tiny-acme-server`
- Sprache Go, Lizenz GPLv3+
- Ziele
  - Minimale aber vollständige ACME Implementierung
  - Möglichst wenig persistenter Zustand
  - Unterschiedliche „Quellen“ für Zertifikate
  - Einfache Erweiterungen und Anpassungen

Software

- `https://gitos.rrze.fau.de/noc/tiny-acme-server`
- Sprache Go, Lizenz GPLv3+
- Ziele
  - Minimale aber vollständige ACME Implementierung
  - Möglichst wenig persistenter Zustand
  - Unterschiedliche „Quellen“ für Zertifikate
  - Einfache Erweiterungen und Anpassungen
- Funktionsweise
  - http-01 Challenge analog zu Let’s Encrypt
  - Danach Zertifikat von Quelle anfordern

Konfiguration

```
Listen = "[::]:443"
# Pfade werden nicht unterstützt
URL = "https://acme.example.org"

# Mit HTTPS Reverse-Proxy optional
CertPath = "web-cert.pem"
KeyPath = "web-key.pem"
# Bei Reverse-Proxy für hilfreiche Logs
# RemoteAddrHeader = "X-Forwarded-For"

# Speicherung der Accounts
StatePath = "state.json"
# Ablegen von ausgestellten Zertifikaten (optional)
CertStorePath = "certs"
```

Konfiguration: Quelle

- Quellen
  - `cmd`: Befehl nimmt CSR von Stdin und schreibt Zertifikat nach Stdout
  - `sectigo`: Erste Implementierung als Prototyp
- HARICA aktuell nur per `cmd`:
  `https://github.com/hm-edu/harica` von Florian Ritterhoff

```
Source = "cmd"
[SourceCmd]
Cmd = ["./harica", "gen-cert", "--config", "harica.yaml"]
```

Konfiguration: harica.yaml

```
# HARICA braucht zwei Nutzer
requester_email: ""
requester_password: ""
validator_email: ""
validator_password: ""
validator_totp_seed: ""

transaction_type: "OV"  # falls gewünscht
stdin: true

debug: true
```

Konfiguration: Clients

- `acme.sh --server https://acme.example.org`
- `acme-tiny --directory-url https://acme.example.org`
- `certbot register --server https://acme.example.org`
  `certbot certonly --server https://acme.example.org`
- `/etc/dehydrated/config`:
  `CA=https://acme.example.org`
- `/etc/apache2/sites-available/example.conf` (mod_md)
  `MDCertificateAuthority https://acme.example.org`

Limitierungen

- Keine Wildcards, da nur http-01
- CSRs werden nicht validiert (das macht die CA)
- Keine Revocation
- Kein Update, Deaktivierung, Rollover von Accounts
- Details siehe README

Zusammenfassung

- Erfahrung
  - Erfüllt unser Anforderungen vollständig
  - Quasi kein Support-Aufwand
  - Bisher 1315 Zertifikate (mit 11497 SANs) ausgestellt

Zusammenfassung

- Erfahrung
  - Erfüllt unser Anforderungen vollständig
  - Quasi kein Support-Aufwand
  - Bisher 1315 Zertifikate (mit 11497 SANs) ausgestellt
- Mögliche Verbesserungen
  - Unterstützung für mehrere Quellen gleichzeitig
  - dns-01 gegenüber Let’s Encrypt