Engineer

ACME

protokollen, der udsteder et certifikat uden et menneske

Hvad det er

ACME, Automatic Certificate Management Environment, er den protokol, et program bruger til at bede en certifikatudsteder om et certifikat, bevise at det kontrollerer de navne, det har bedt om, og komme tilbage efter et nyt, før det gamle udløber. Undervejs er der hverken formularer at udfylde eller bekræftelsesmails at besvare. RFC 8555 gjorde protokollen til standard i marts 2019, og Let's Encrypt er den udsteder, de fleste møder den gennem.

Klienten starter med sin egen kontonøgle, holdt adskilt fra nøglen i ethvert certifikat, og signerer hver forespørgsel med den. Vejen til et certifikat går derefter gennem fem ressourcer på serveren:

```
newAccount   registrér kontonøglen én gang
newOrder     angiv de navne, certifikatet skal dække
authz        én autorisation pr. navn, hver med udfordringer
finalize     send en certifikatanmodning (CSR), når alle er gyldige
certificate  hent den udstedte kæde, og installér den
```

En udfordring (challenge) er måden, klienten viser, at den kontrollerer et navn. Serveren udleverer et token; klienten sætter det sammen med et fingeraftryk af kontonøglen, så beviset hører til netop den konto, og lægger resultatet et sted, som kun den, der styrer navnet, kan skrive til. Tre typer er værd at kende:

- **http-01** lægger værdien som en fil på `/.well-known/acme-challenge/<token>` under selve navnet. Udstederens første forespørgsel går altid over almindelig HTTP til port 80.
- **dns-01** lægger en hash af værdien i en TXT-post på `_acme-challenge` under navnet. Den er den eneste af de tre, der kan bevise kontrol over et wildcard som `*.example.com`.

- **tls-alpn-01**, som RFC 8737 tilføjede i februar 2020, svarer på port 443 under et TLS-håndtryk med en særlig ALPN-protokol. Den er lavet til en reverse proxy, der ejer port 443 og ikke kan bruge port 80.

RFC 9773 fra juni 2025 tilføjede ACME Renewal Information, kaldet ARI. Udstederen offentliggør et foreslået tidsrum for fornyelse af hvert certifikat, den har udstedt, og klienten spørger efter det i stedet for at gætte. Så kan udstederen rykke fornyelser frem, for eksempel før en massetilbagekaldelse, uden at nogen skal rette i en konfigurationsfil.

Brug begrebet i en beslutning

Brug »Hvad er ACME?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Konti, adgangskoder og rettigheder sat op, så folk har præcis den adgang, jobbet kræver, og fratrådte mister den samme dag.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har skrevet 1.000 RBAC-regler til Linux-servere — 1,000 adgangsregler skrevet for at lukke et sikkerhedshul

Har skrevet 1.000 RBAC-regler til Linux-servere Kontakt os →

Hvorfor det betyder noget

Om et menneske stadig kan forny certifikater i hånden, er ikke længere et spørgsmål om smag. Svaret har fået datoer på.

CA/Browser Forum fastsætter de regler, som alle offentligt betroede udstedere følger. Med afstemningen SC-081v3 fra april 2025 må et certifikat, der er udstedt fra 15. marts 2026, højst gælde i 200 dage. Grænsen falder til 100 dage fra 15. marts 2027 og til 47 dage fra 15. marts 2029, og den tid, et bevis for kontrol over et domæne må genbruges, falder i samme takt: fra 200 dage nu til 100 og til sidst kun 10. Med de endelige regler skal et navn, der fornyes ved to tredjedele af levetiden, have nyt certifikat cirka hver måned, og næsten alle de fornyelser kræver en frisk udfordring.

Let’s Encrypt er foran den plan. Deres standardcertifikater gælder i 90 dage i dag. Fra 10. februar 2027 gælder de i 64 dage, og en autorisation kan genbruges i 10 dage; fra 16. februar 2028 gælder de i 45 dage, og genbrug er skåret ned til 7 timer. Deres egen meddelelse fraråder manuel fornyelse. De holdt også op med at sende påmindelser om udløb pr. e-mail den 4. juni 2025, så selv det sikkerhedsnet, der fangede en glemt fornyelse, er væk.

Manuel fornyelse er altså ude, og ACME er udgangspunktet. Tilbage står at beslutte, hvor de bevægelige dele skal bo:

- **Udfordringen.** http-01 er den lette start for en enkelt server, der svarer på port 80. dns-01 er det ekstra opsætningsarbejde værd til wildcards, til flere servere bag samme navn og til værter, internettet ikke kan nå. tls-alpn-01 hører hjemme i proxyer, der har port 443, men mangler port 80.
- **Klienten.** Den kan bo inde i webserveren, køre ved siden af med en hook, der installerer hvert nyt certifikat, eller køre på en separat maskine, der validerer centralt og kopierer certifikaterne ud.
- **Vagten.** Noget uden for klienten må opdage, når en fornyelse udebliver, for det er klienten, der lige har svigtet.

Sådan ser det ud i praksis

Som DevOps-ingeniør tog virksomheden certifikatarbejdet ud af menneskers hænder for hundrede Docker-applikationer på Ubuntu Linux-værter, hvor både Apache og Nginx tog imod trafik. ACME stod for hvert trin: udstedelse, fornyelse i god tid og udrulning på værterne. Intet trin i kredsløbet ventede på, at nogen huskede en dato. Alle hundrede applikationer blev ved med at have gyldige certifikater, og håndarbejdet forsvandt.

Udrulningen er det trin, der er lettest at undervurdere. Apache og Nginx læser et certifikat, når de starter eller genindlæser, og bliver ved med at levere den kopi, så en klient, der blot skriver en ny fil, har ikke fornyet noget, en besøgende kan se. Hvor Apache og Nginx deler en flåde, skal hver af dem genindlæses hver for sig, når et nyt certifikat lander.

Virksomhedens eget site viser den anden opstilling, hvor klienten er webserveren. Her henter Caddy selv sitets certifikat hos Let’s Encrypt og fornyer det. Caddy taler ACME direkte og kræver intet separat værktøj, så der er intet udrulningstrin at glemme.

Hvor det går galt

Valideringen løber ud i tid eller giver 404, selvom sitet åbner fint i en browser. Forespørgslen for http-01 kommer ind på port 80, og noget på vejen er forkert: en firewall eller en sikkerhedsgruppe i skyen, der kun åbner 443, en omdirigering til en anden port end 80 eller 443, eller en load balancer, der sender forespørgslen til en server, som aldrig fik tokenet. Let’s Encrypt følger op til ti

omdirigeringer, og kun til de to porte. Hold port 80 åben, lever udfordringsstien fra hver backend eller fra delt lager, og skift til dns-01, hvor port 80 er umulig at få.

DNS-nøglen på webserveren kan omskrive hele zonen. Et wildcard kræver dns-01, så klienten skal kunne oprette TXT-poster, og den letteste nøgle at få er tit en, der kan ændre alle poster på kontoen. Bliver webserveren kompromitteret, kan angriberen også omdirigere mail og alle andre navne. Brug en nøgle, der kun gælder `_acme-challenge` -posterne, hvor udbyderen tilbyder det, eller peg `_acme-challenge` med en CNAME over på en zone, der kun bruges til validering, eller valider fra en separat maskine og kopier certifikaterne over. DNS-ændringer tager tid om at sprede sig, så klienten kan være nødt til at vente op til en time, før den beder om validering.

Efter en ny udrulning starter containerne uden certifikater. Hver container beder om sit eget certifikat, når den starter, og gemmer det kun i sit eget filsystem, så hver genstart beder igen. Udrul ofte nok, eller start mange på én gang, og udstederens grænser for antal forespørgsler afviser de ekstra ordrer; RFC 8555 har sit eget navn for den fejl, `rateLimited` . Let's Encrypt advarer om, at i værste fald får ingen af instanserne et certifikat. Gem certifikater og nøgler i varigt lager, og lad en ny instans bruge et gyldigt certifikat derfra, før den bestiller et.

En fast fornyelsestimer lader certifikater udløbe, når levetiden krymper. Jobbet fornyer for eksempel hver 60. dag, et tempo afstemt efter certifikater på 90 dage. Et certifikat på 64 dage giver timeren fire dages margin, og et på 45 dage udløber cirka to uger før timeren slår til. RFC 9773 nævner faste intervaller som en af de vaner, der gør det svært for udstedere at ændre levetider. Slå ARI til, hvor klienten understøtter det. Let's Encrypt råder til at tjekke det mindst to gange om dagen og som reserve at forny, når en tredjedel af levetiden er tilbage.

Klienten melder succes, og de besøgende ser stadig det gamle certifikat. Den nye fil ligger på disken, men intet har bedt webserveren genindlæse, eller hooken genindlæste én server ud af flere. Det gamle certifikat bliver i brug, til det udløber. Genindlæs serveren fra klientens deploy-hook, og hold øje med det certifikat, der faktisk leveres, set udefra, i stedet for filen:

```
openssl s_client -connect example.com:443 -servername example.com \
    </dev/null 2>/dev/null | openssl x509 -noout -enddate
```

Læs mere

- RFC 8555: Automatic Certificate Management Environment (ACME) — konti, ordrer, autorisationer og udfordringerne http-01 og dns-01.
- RFC 9773: ACME Renewal Information (ARI) Extension — det tidsrum for fornyelse, en udsteder foreslår for hvert certifikat.
- RFC 8737: ACME TLS Application-Layer Protocol Negotiation (ALPN) Challenge Extension — udfordringen tls-alpn-01, besvaret helt på port 443.
- Challenge Types — Let's Encrypts egen sammenligning af de tre udfordringer og deres begrænsninger.
- Decreasing Certificate Lifetimes to 45 Days — Let's Encrypts datoer for kortere certifikater og deres råd om fornyelse.
- Ballot SC-081v3 — CA/Browser Forums afstemning om grænserne på 200, 100 og 47 dage.

Ordliste

Vælg ud fra jeres faktiske vilkår

Medbring udrulningsforløbet, driftsrisiciene og dem, der har vagten. Vi kan afgrænse en ændring med kontroller, en vej tilbage og tydeligt ansvar.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er ACME?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/acme/