Certyfikaty publiczne i ACME: Let's Encrypt z DNS-01, skracanie ważności i step-ca

Poziom: średnio

Dotyczy: Let's Encrypt, ACME (RFC 8555), certbot z wtyczką RFC 2136, Traefik na docker01, step-ca 0.30.2 na acme01 (10.20.10.128, opcja), Rocky Linux 9, harmonogram CA/Browser Forum SC081v3

Czas: ~50 min

Certyfikaty publiczne, którym ufa każda przeglądarka, od 15 marca 2026 roku żyją najwyżej 200 dni, od marca 2027 najwyżej 100, a od marca 2029 najwyżej 47. Przy takich liczbach ręczne odnawianie przestaje być możliwe, więc ta strona pokazuje automatyczne odnawianie protokołem ACME: publicznie z Let's Encrypt i walidacją przez DNS, a wewnętrznie z własnym urzędem ACME na step-ca. Wracamy tu przy każdej nowej usłudze widocznej z internetu i przed każdym progiem skracania ważności.

ZAŁOŻENIA SCENARIUSZA

strefa publiczna `zielona.example` jest u dostawcy DNS, który przyjmuje aktualizacje RFC 2136 z kluczem TSIG; gdy dostawca ma tylko własne API, używamy wtyczki certbota albo dostawcy w Traefiku dla tego API, a reszta strony się nie zmienia. Traefik na `docker01` już odnawia certyfikaty publiczne przez HTTP-01. `acme01` to propozycja: Rocky Linux 9 bez Dockera, step-ca jako urząd podrzędny `Zielona-Root-CA`, ograniczony do nazw w `zielona.example`.

Skróty użyte na tej stronie (29)

Skrót	Co znaczy
ACME	Automatic Certificate Management Environment: protokół, którym Traefik albo certbot automatycznie zamawiają i odnawiają certyfikat w Let's Encrypt, bez ręcznego wniosku

Skrót	Co znaczy
AD CS	Active Directory Certificate Services: rola Windows Server pełniąca funkcję firmowego urzędu certyfikacji
API	Application Programming Interface: sposób, w jaki programy rozmawiają ze sobą
ARI	ACME Renewal Information: rozszerzenie protokołu ACME, w którym urząd podpowiada klientowi, kiedy odnowić certyfikat, także wcześniej po masowym unieważnieniu
B2B	business to business: sprzedaż i systemy skierowane do innych firm, nie do klientów indywidualnych
CA	Certificate Authority: urząd certyfikacji, który podpisuje certyfikaty
DNS	Domain Name System: książka telefoniczna internetu, zamienia nazwy na adresy IP
DNS-01	sposób potwierdzenia domeny w ACME: urząd certyfikacji każe opublikować jednorazowy rekord TXT _acme-challenge i sprawdza go w DNS; działa bez portu 80 i pozwala na certyfikat wieloznaczny
DNS-PERSIST-01	nowa walidacja ACME ze stałym rekordem TXT _validation-persist, który nie zmienia się przy każdym odnowieniu (głosowanie CA/Browser Forum SC-088v3)
EAP-TLS	EAP z TLS: uwierzytelnianie certyfikatem użytkownika albo urządzenia zamiast hasła
EJBCA	Enterprise JavaBeans Certificate Authority: otwarty urząd certyfikacji firmy Keyfactor z interfejsem WWW i obsługą wielu protokołów rejestracji, wersja Community bezpłatna
EPEL	Extra Packages for Enterprise Linux: dodatkowe repozytorium pakietów dla RHEL i klonów
HTTP-01	sposób potwierdzenia domeny w ACME: urząd certyfikacji pobiera jednorazowy plik z adresu /.well-known/acme-challenge/ na porcie 80 naszego serwera
IIS	Internet Information Services: serwer WWW wbudowany w Windows Server
NPS	Network Policy Server: rola Windows Server obsługująca RADIUS, pośredniczy w sprawdzaniu hasła domenowego przy logowaniu do VPN albo Wi-Fi
NTAuth	magazyn NTAuth (NT Authentication): lista urzędów w konfiguracji lasu AD, których certyfikaty mogą służyć do logowania do domeny
OCSP	Online Certificate Status Protocol: sprawdzanie na żywo, czy certyfikat nie został unieważniony
RFC	Request for Comments: oficjalny dokument opisujący standard internetowy, na przykład sposób działania danego protokołu; w ITIL także Request for Change, czyli wniosek o zmianę
RHEL	Red Hat Enterprise Linux: komercyjna dystrybucja Linuksa
RPM	RPM Package Manager: format paczek i narzędzie instalujące oprogramowanie w RHEL/Rocky, dnf korzysta z niego pod spodem
SCEP	Simple Certificate Enrollment Protocol: protokół, którym urządzenie samo prosi urząd certyfikacji o wystawienie certyfikatu
SSL	Secure Sockets Layer: dawna nazwa szyfrowania, dziś formalnie TLS
TLS	Transport Layer Security: szyfrowanie połączeń, np. w HTTPS

Skrót	Co znaczy
TLS-ALPN-01	walidacja ACME, w której serwer pokazuje na porcie 443 specjalny certyfikat z tokenem urzędu (ALPN: Application-Layer Protocol Negotiation)
TSIG	Transaction Signature: podpis kluczem wspólnym, którym serwer DNS potwierdza, że aktualizację rekordu wysłał uprawniony klient, na przykład FortiGate w genericDDNS według RFC 2136
TXT	typ rekordu DNS przechowującego dowolny tekst, wykorzystywany między innymi do publikowania reguł SPF, DKIM i DMARC
VPN	Virtual Private Network: szyfrowany tunel do sieci firmowej
Wi-Fi	Sieć bezprzewodowa: wygodna w biurze, ale w serwerowni dokłada kolejną możliwą przyczynę awarii
WMS	Warehouse Management System: system zarządzania magazynem

Fiszka

Polecenia wykonujemy w bash jako użytkownik z sudo: certbot na serwerze, który potrzebuje certyfikatu, step-ca na `acme01`. Zamiast `NazwaDomeny`, `NazwaSerwera`, `AdresSerweraDNS` i nazw kluczy wpisujemy własne wartości.

□□ ZMIENIA SYSTEM

Instalacja certbota z wtyczką RFC 2136: dnf install

```
sudo dnf install -y epel-release
sudo dnf install -y certbot python3-certbot-dns-rfc2136
```

Komendy można wykonać jako blok. Certbot i wtyczka pochodzą z repozytorium EPEL; nazwę pakietu wtyczki sprawdzamy w `dnf search certbot-dns`, gdy repozytorium ją zmieni.

□□ **RFC** = Request for Comments: oficjalny dokument opisujący standard internetowy, na przykład sposób działania danego protokołu; w ITIL także Request for Change, czyli wniosek o zmianę ·**EPEL** = Extra Packages for Enterprise Linux: dodatkowe repozytorium pakietów dla RHEL i klonów ·**dnf** = Dandified YUM: menedżer pakietów RHEL/Rocky

□□ PLIK KONFIGURACYJNY

Plik z kluczem TSIG dla wtyczki: /etc/letsencrypt/rfc2136.ini

```
dns_rfc2136_server = AdresSerweraDNS
dns_rfc2136_port = 53
dns_rfc2136_name = NazwaKluczaTsig
dns_rfc2136_secret = SekretKluczaTsig
dns_rfc2136_algorithm = HMAC-SHA512
```

Plik daje prawo zmiany rekordów w strefie, więc ma prawa `600` i właściciela `root` (`sudo chmod 600 /etc/letsencrypt/rfc2136.ini`). Klucz TSIG u dostawcy ograniczamy, jeśli się da, do rekordów `_acme-challenge`.

TSIG = Transaction Signature: podpis kluczem wspólnym, którym serwer DNS potwierdza, że aktualizację rekordu wysłał uprawniony klient, na przykład FortiGate w genericDDNS według RFC 2136

ZMIENIA SYSTEM

Wystawienie certyfikatu z walidacją DNS-01: certbot certonly --dns-rfc2136

```
sudo certbot certonly --dns-rfc2136 --dns-rfc2136-credentials
/etc/letsencrypt/rfc2136.ini --dns-rfc2136-propagation-seconds 60 -d "NazwaDomeny" -d
"*.NazwaDomeny"
```

Certbot dopisuje w strefie rekord TXT `_acme-challenge`, czeka minutę na rozejście się zmiany, Let's Encrypt sprawdza rekord, a certbot go usuwa. Walidacja DNS-01 jest jedyną, która daje certyfikat z gwiazdką (`*.zielona.example`) i certyfikat dla nazwy, która nie jest osiągalna z internetu.

Przykład (skrót):

```
Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/zielona.example/fullchain.pem
Key is saved at:         /etc/letsencrypt/live/zielona.example/privkey.pem
This certificate expires on 2027-01-24.
```

DNS-01 = sposób potwierdzenia domeny w ACME: urząd certyfikacji każe opublikować jednorazowy rekord TXT _acme-challenge i sprawdza go w DNS; działa bez portu 80 i pozwala na certyfikat wieloznaczny · **TXT** = typ rekordu DNS przechowującego dowolny tekst, wykorzystywany między innymi do publikowania reguł SPF, DKIM i DMARC · **certbot** = automatyczne certyfikaty Let's Encrypt

□□ TYLKO ODCZYT

Próba odnowienia bez wydawania: certbot renew --dry-run

```
sudo certbot renew --dry-run
```

Przechodzi całe odnowienie na serwerze testowym Let's Encrypt (staging), bez nowych certyfikatów i bez limitów. Robimy to po każdej zmianie w DNS, kluczu TSIG albo zaporze.

□□ **DNS** = Domain Name System: książka telefoniczna internetu, zamienia nazwy na adresy IP · **TSIG** = Transaction Signature: podpis kluczem wspólnym, którym serwer DNS potwierdza, że aktualizację rekordu wysłał uprawniony klient, na przykład FortiGate w genericDDNS według RFC 2136 · **certbot** = automatyczne certyfikaty Let's Encrypt

□□ TYLKO ODCZYT

Sprawdzenie harmonogramu odnawiania: systemctl list-timers

```
systemctl list-timers | grep -i certbot
```

Pakiet z EPEL instaluje timer systemd, który dwa razy dziennie uruchamia `certbot renew`. Certbot odnawia certyfikat, gdy zostaje mniej niż jedna trzecia ważności, a przy kliencie z obsługą ARI wtedy, gdy wskaże to urząd (do sprawdzenia w naszej wersji certbota).

□□ **EPEL** = Extra Packages for Enterprise Linux: dodatkowe repozytorium pakietów dla RHEL i klonów · **ARI** = ACME Renewal Information: rozszerzenie protokołu ACME, w którym urząd podpowiada klientowi, kiedy odnowić certyfikat, także wcześniej po masowym unieważnieniu ·**systemctl** = system control: sterowanie usługami systemd

□□ ZMIENIA SYSTEM

Przeładowanie usługi po odnowieniu: --deploy-hook

```
sudo certbot reconfigure --cert-name NazwaDomeny --deploy-hook "systemctl reload nginx"
```

Zapisuje w konfiguracji odnowienia polecenie, które certbot wykona po każdym udanym odnowieniu. Bez tego usługa trzyma stary certyfikat w pamięci do restartu, a certyfikat na dysku jest już nowy. Polecenie `reconfigure` jest w nowszych wersjach certbota; w starszych dopisujemy `renew_hook` w pliku `/etc/letsencrypt/renewal/NazwaDomeny.conf`.

□□ **certbot** = automatyczne certyfikaty Let's Encrypt

PLIK KONFIGURACYJNY

Walidacja DNS-01 w Traefiku: certificatesresolvers.dnschallenge

```
command:
  - "--certificatesresolvers.le-dns.acme.email=it@zielona.example"
  - "--certificatesresolvers.le-dns.acme.storage=/acme/acme-dns.json"
  - "--certificatesresolvers.le-dns.acme.dnschallenge=true"
  - "--certificatesresolvers.le-dns.acme.dnschallenge.provider=rfc2136"
environment:
  RFC2136_NAMESERVER: "AdresSerweraDNS:53"
  RFC2136_TSIG_KEY: "NazwaKluczaTsig"
  RFC2136_TSIG_ALGORITHM: "hmac-sha512."
  RFC2136_TSIG_SECRET_FILE: "/run/secrets/tsig_secret"
```

Fragment usługi `traefik` w `compose.yaml` ze strony Reverse proxy w Dockerze: Traefik przed wszystkimi aplikacjami na docker01: drugi resolver obok istniejącego HTTP-01. Nazwy zmiennych pochodzą z biblioteki lego, której używa Traefik; przed wdrożeniem sprawdzamy je w dokumentacji naszej wersji. Sekret trzymamy w pliku z Docker secrets albo `.env`, nie w `compose.yaml`.

DNS-01 = sposób potwierdzenia domeny w ACME: urząd certyfikacji każe opublikować jednorazowy rekord TXT _acme-challenge i sprawdza go w DNS; działa bez portu 80 i pozwala na certyfikat wieloznaczny · **HTTP-01** = sposób potwierdzenia domeny w ACME: urząd certyfikacji pobiera jednorazowy plik z adresu /.well-known/acme-challenge/ na porcie 80 naszego serwera

ZMIENIA SYSTEM

Instalacja step-ca na acme01: dnf install step-ca

```
sudo tee /etc/yum.repos.d/smallstep.repo >/dev/null <<'EOT'
[smallstep]
name=Smallstep
baseurl=https://packages.smallstep.com/stable/fedora/
enabled=1
repo_gpgcheck=0
gpgcheck=1
gpgkey=https://packages.smallstep.com/keys/smallstep-0x889B19391F774443.gpg
EOT
sudo dnf makecache && sudo dnf install -y step-cli step-ca
```

```
step-ca version
```

Komendy można wykonać jako blok. Repozytorium producenta podaje pakiety dla rodziny Fedora i RHEL. Najnowsza wersja, którą znaleźliśmy w październiku 2026, to 0.30.2 z 23 marca 2026; wersje starsze niż 0.30.0 mają krytyczną lukę w SCEP (CVE-2026-30836), więc nie instalujemy ich.

Przykład:

```
Smallstep CA/0.30.2 (linux/amd64)
Release Date: 2026-03-23 ...
```

RHEL = Red Hat Enterprise Linux: komercyjna dystrybucja Linuksa · **SCEP** = Simple Certificate Enrollment Protocol: protokół, którym urządzenie samo prosi urząd certyfikacji o wystawienie certyfikatu · **tee** = zapisz strumień wejścia do pliku i jednocześnie wypisz go na ekran · **dnf** = Dandified YUM: menedżer pakietów RHEL/Rocky · **step-ca** = step certificate authority: otwarty urząd certyfikacji Smallstep z ACME, SCEP i własnymi provisionerami

ZMIENIA SYSTEM

Utworzenie urzędu step-ca: step ca init

```
step ca init --deployment-type standalone --name "NazwaUrzeduAcme" --dns "NazwaSerwera"
--address ":443" --provisioner "admin"
```

Tworzy konfigurację, własny urząd główny i urząd pośredni w `$(step path)`. Własny urząd główny zaraz usuniemy, bo podpisem dla `acme01` będzie `Zielona-Root-CA`; kreator jest potrzebny tylko po to, żeby powstały wszystkie pliki konfiguracji.

step = step CLI (Smallstep): narzędzie do certyfikatów, kluczy i zarządzania urzędem step-ca

OSTROŻNIE · NIEODWRACALNE

Podpięcie step-ca pod Zielona-Root-CA: step certificate create --csr

```
shred -u "$(step path)/secrets/root_ca_key"
cp zielona-root-ca.crt "$(step path)/certs/root_ca.crt"
step certificate create "NazwaUrzeduAcme" intermediate.csr intermediate_ca_key --csr --
kty RSA --size 3072
```

Komendy można wykonać jako blok. Usuwa klucz urzędu głównego wygenerowanego przez kreator, podmienia certyfikat główny na `Zielona-Root-CA` i tworzy klucz z żądaniem dla

urzędu pośredniego. Plik `intermediate.csr` jedzie na nośniku na ceremonię `ca-root01`, a podpisany certyfikat wraca jako `$(step path)/certs/intermediate_ca.crt` (klucz trafia do `$(step path)/secrets/intermediate_ca_key`).

shred = Linux: nadpisuj plik albo cały dysk wielokrotnie, żeby utrudnić odzyskanie danych, na SSD nie daje gwarancji · **cp** = copy: skopiuj · **step** = step CLI (Smallstep): narzędzie do certyfikatów, kluczy i zarządzania urzędem step-ca · **RSA** = Rivest, Shamir, Adleman: algorytm szyfrowania kluczem publicznym, którym Vaultwarden podpisuje tokeny sesji, stąd pliki rsa_key* w danych aplikacji

PLIK KONFIGURACYJNY

Ograniczenie nazw w urzędzie: policy w ca.json

```
"policy": {
  "x509": {
    "allow": {
      "dns": ["*.zielona.example", "*.ad.zielona.example"]
    },
    "allowWildcardNames": false
  }
}
```

Fragment pliku `$(step path)/config/ca.json`, wstawiony wewnątrz obiektu `authority` (tak wymaga dokumentacja Smallstep): step-ca wyda certyfikat wyłącznie dla nazw z listy i nie wyda certyfikatu z gwiazdką. To zabezpieczenie po stronie oprogramowania; twarde ograniczenie nazw w samym certyfikacie urzędu (name constraints) wymaga przygotowania na ceremonii i sprawdzamy je osobno.

ZMIENIA SYSTEM

Dodanie provisionera ACME: step ca provisioner add --type ACME

```
step ca provisioner add acme --type ACME
sudo systemctl restart step-ca
```

Od tej chwili `acme01` ma katalog ACME pod adresem `https://acme01.ad.zielona.example/acme/acme/directory`. Domyślna ważność certyfikatów step-ca to 24 godziny; dłuższą (na przykład 7 dni) ustawiamy w opcjach provisionera, a nazwy opcji sprawdzamy w `step ca provisioner add --help`. Usługę systemd zakładamy według dokumentacji „Running step-ca as a daemon”.

□□ **ACME** = Automatic Certificate Management Environment: protokół, którym Traefik albo certbot automatycznie zamawiają i odnawiają certyfikat w Let's Encrypt, bez ręcznego wniosku ·**step** = step CLI (Smallstep): narzędzie do certyfikatów, kluczy i zarządzania urzędem step-ca ·**systemctl** = system control: sterowanie usługami systemd

□□ ZMIENIA SYSTEM

Certyfikat z wewnętrznego ACME: certbot --server

```
sudo REQUESTS_CA_BUNDLE=/etc/pki/tls/certs/ca-bundle.crt certbot certonly --standalone
--server https://acme01.ad.zielona.example/acme/acme/directory -d "NazwaSerwera"
```

Ten sam certbot, inny urząd. `REQUESTS_CA_BUNDLE` wskazuje magazyn systemowy, w którym jest już `Zielona-Root-CA`, bo inaczej certbot nie zaufa samemu `acme01`. Walidacja HTTP-01 wymaga, żeby `acme01` dotarł do portu 80 serwera.

□□ **ACME** = Automatic Certificate Management Environment: protokół, którym Traefik albo certbot automatycznie zamawiają i odnawiają certyfikat w Let's Encrypt, bez ręcznego wniosku ·**HTTP-01** = sposób potwierdzenia domeny w ACME: urząd certyfikacji pobiera jednorazowy plik z adresu /.well-known/acme-challenge/ na porcie 80 naszego serwera

Z życia wzięte

□□ Z ŻYCIA WZIĘTE

Piątek, 15:50, marzec 2026. Ola zgłasza, że nowy certyfikat dla sklepu B2B kupiony u komercyjnego urzędu „jest za krótki”: zamówiła roczny, a dostała ważny 199 dni. Tomek sprawdza i wyjaśnia, że od 15 marca 2026 żaden publiczny urząd nie wyda dłuższego niż 200 dni, a za rok będzie to 100. Pan Zbigniew pyta, czy w takim razie co trzy miesiące ktoś będzie siedział nad certyfikatami. Odpowiadamy, że nie, i jeszcze w tym miesiącu sklep przechodzi na certyfikat z Let's Encrypt odnawiany przez ACME.

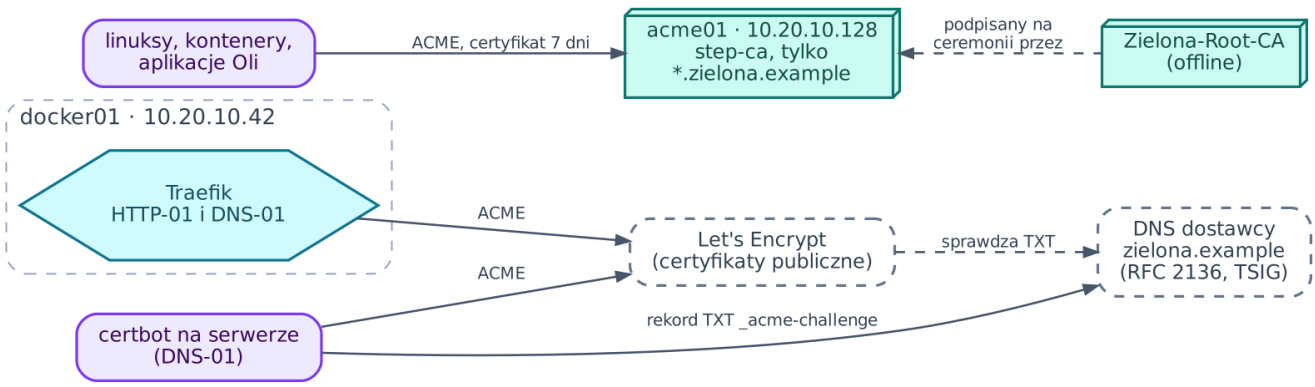
Jak to działa

□□ **Rozwiń: jak to działa pod spodem**

ACME w jednym akapicie

> **Pojęcie: ACME (Automatic Certificate Management Environment).** Protokół, którym program na serwerze sam zakłada konto w urzędzie, prosi o certyfikat, udowadnia, że kontroluje nazwę, odbiera certyfikat i odnawia go przed końcem ważności. Opisany w RFC 8555 z 2019 roku, mówią nim Let's Encrypt, wiele urzędów komercyjnych, step-ca, EJBCA i FreeIPA.

Dowód kontroli nazwy (walidacja) ma trzy popularne odmiany: HTTP-01 (plik pod adresem `http://nazwa/.well-known/acme-challenge/`), DNS-01 (rekord TXT `_acme-challenge.nazwa`) i TLS-ALPN-01 (specjalny certyfikat na porcie 443). Traefik w Zielonej używa HTTP-01, więc port 80 na `docker01` musi być osiągalny z internetu.



Rysunek 1. Dwie drogi ACME w Zielonej: certyfikaty publiczne z Let's Encrypt przez DNS-01 u dostawcy strefy zielona.example albo HTTP-01 przez Traefika na docker01, oraz certyfikaty wewnętrzne z acme01 ze step-ca podpisanego przez Zielona-Root-CA, dla serwerów linuksowych i kontenerów (kliknięcie otwiera pełny rozmiar)

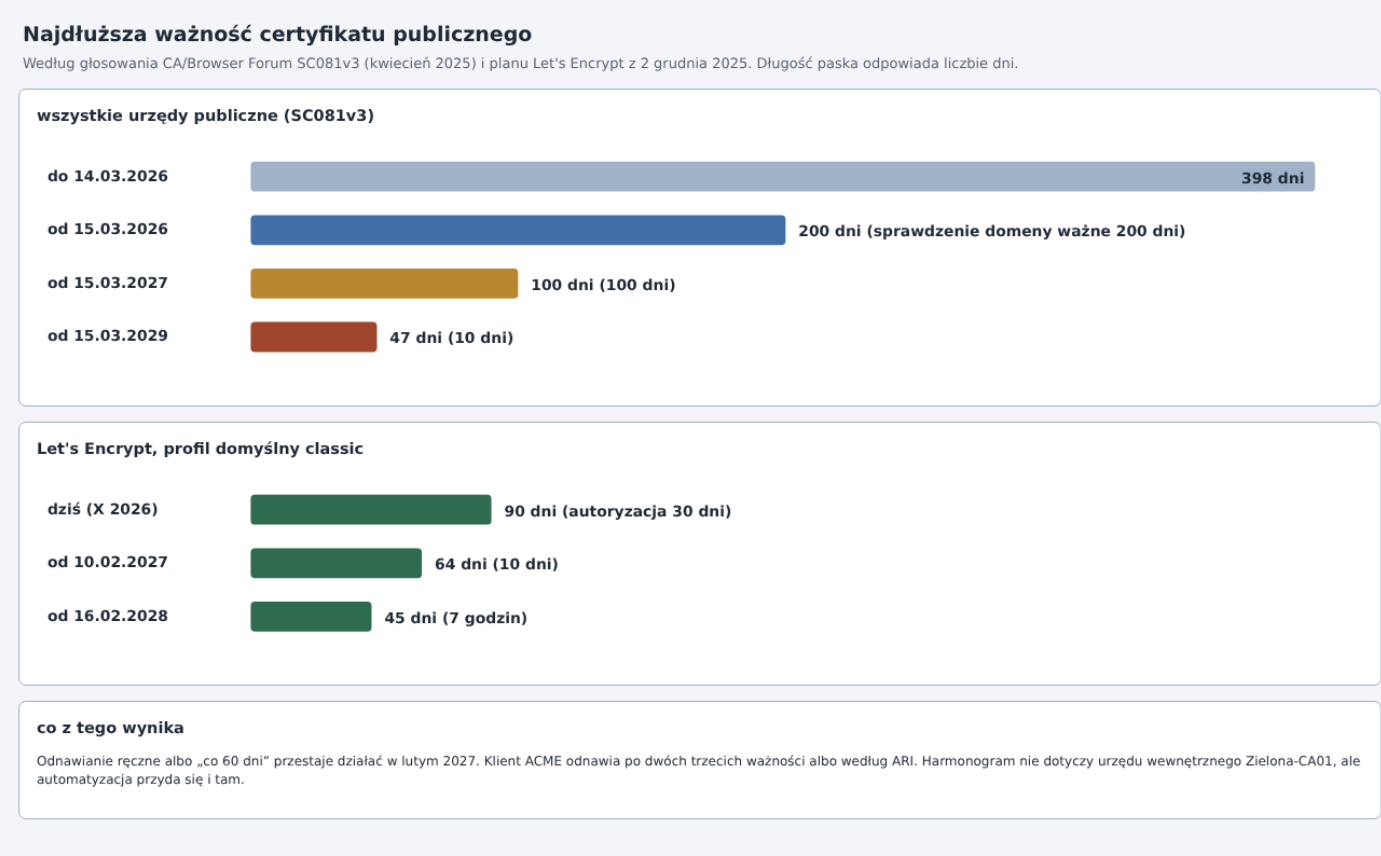
Walidacja	Kiedy wybieramy	Wady
HTTP-01	serwer jest osiągalny z internetu na porcie 80, pojedyncze nazwy	brak certyfikatów z gwiazdką, nie działa dla nazw wewnętrznych
DNS-01	certyfikaty z gwiazdką, nazwy wewnętrzne w domenie publicznej, serwery za zaporą	klucz do zmiany strefy DNS na serwerze; jego wyciek pozwala przejąć domenę
TLS-ALPN-01	serwer ma tylko port 443	mniej klientów, nie działa przez większość proxy

Walidacja	Kiedy wybieramy	Wady
DNS-PERSIST-01	jeden stały rekord `_validation-persist` zamiast zmiany przy każdym odnowieniu	nowa metoda (głosowanie SC-088v3 w październiku 2025); Let's Encrypt planował produkcję na II kwartał 2026, a obsługę w naszym kliencie sprawdzamy przed użyciem

> ⚠ Każdy certyfikat z publicznego urzędu trafia do publicznych dzienników Certificate Transparency. Certyfikat dla `git.zielona.example` z Let's Encrypt ogłasza światu, że taka nazwa istnieje. Dla nazw, których nie chcemy pokazywać, wybieramy certyfikat z gwiazdką albo urząd wewnętrzny.

Harmonogram skracania ważności

W kwietniu 2025 roku CA/Browser Forum przyjęło głosowanie SC081v3. Ustala ono najdłuższą ważność certyfikatów TLS z urzędów publicznie zaufanych i to, jak długo urząd może korzystać z wcześniejszego sprawdzenia kontroli domeny.



Rysunek 2. Harmonogram ważności certyfikatów publicznych: 398 dni do 15 marca 2026, potem 200 dni, od 15 marca 2027 100 dni, od 15 marca 2029 47 dni; obok Let's Encrypt: 90 dni dziś, 64 dni od 10 lutego 2027 i 45 dni od 16 lutego 2028 (kliknięcie otwiera pełny rozmiar)

Od dnia	Najdłuższa ważność certyfikatu	Ponowne użycie sprawdzenia domeny
do 14 marca 2026	398 dni	398 dni
15 marca 2026	200 dni	200 dni
15 marca 2027	100 dni	100 dni
15 marca 2029	47 dni	10 dni

Let's Encrypt ma własny, krótszy harmonogram, ogłoszony 2 grudnia 2025 roku:

Od dnia	Profil	Ważność	Ponowne użycie autoryzacji
dziś (październik 2026)	`classic` (domyślny)	90 dni	30 dni
13 maja 2026	`tlsserver` (do wyboru)	45 dni	według profilu
10 lutego 2027	`classic`	64 dni	10 dni
16 lutego 2028	`classic`	45 dni	7 godzin
bez zmian	`shortlived` (do wyboru)	6 dni	według profilu

Dwie rzeczy z tego wynikają dla Zielonej. Po pierwsze, skrypt albo kalendarz z odnawianiem „co 60 dni” przestanie działać w lutym 2027, więc klient ma odnawiać według ARI albo po dwóch trzecich ważności. Po drugie, od 11 lutego 2026 domyślne certyfikaty Let's Encrypt nie mają zastosowania do uwierzytelniania klienta, a profil `tlsclient` zakończono 8 lipca 2026: certyfikaty publiczne nie nadają się już do logowania serwer do serwera (na przykład wzajemnego TLS między serwerami poczty), do tego służy urząd wewnętrzny.

> **Pojęcie: ARI (ACME Renewal Information).** Rozszerzenie ACME, w którym urząd podpowiada klientowi, kiedy najlepiej odnowić certyfikat. Gdy urząd musi masowo unieważnić certyfikaty po błędzie, przez ARI każe klientom odnowić je wcześniej, a klient z ARI zrobi to bez naszego udziału.

> Harmonogram SC081v3 nie dotyczy urzędów wewnętrznych. `Zielona-CA01` może dalej wydawać certyfikaty serwerów na rok, bo komputery ufają mu z magazynu firmowego, a nie z listy przeglądarek. Krótsza ważność i ACME

> także wewnątrz mają jednak sens: certyfikat, który odnawia się sam, nie wygasa w piątek o 16:00.

Wewnętrzny ACME: step-ca na acme01

AD CS nie mówi protokołem ACME. Serwery z Windows dostają certyfikaty przez automatyczną rejestrację, ale linuksy, kontenery, urządzenia sieciowe i aplikacje Oli dostają je dziś ręcznie, z `certreq` i OpenSSL, według strony Generowanie certyfikatów: OpenSSL i urząd certyfikacji Windows. Własny urząd ACME zamienia to na ten sam mechanizm, którego używamy z Let's Encrypt.

Pytanie	Odpowiedź dla Zielonej
czym jest `acme01`	urzędem podrzędnym `Zielona-Root-CA` ze step-ca, podpisanym na ceremonii, z listą dozwolonych nazw `*.zielona.example` i `*.ad.zielona.example`
czy może wydać certyfikat do logowania w domenie	nie: nie jest w magazynie NTAuth, więc kontroler nie przyjmie jego certyfikatów do logowania
jaka ważność	7 dni dla serwerów z klientem ACME, odnowienie automatyczne
warstwa	0, dopóki certyfikat urzędu nie ma twardego ograniczenia nazw (name constraints); z ograniczeniem do rozważenia warstwa 1
lista unieważnień	przy ważności 7 dni unieważnienie ma małe znaczenie; czy włączamy CRL w step-ca, decydujemy przy wdrożeniu (opis w dokumentacji Smallstep)
co z FreeIPA	serwery zapisane w FreeIPA na `ipa01` mogą zamiast tego używać ACME z Dogtag (`ipa-acme-manage enable`) z urzędem FreeIPA; wtedy ten urząd musi być zaufany na stacjach

> ⚠ `acme01` dostaje certyfikat urzędu, ale nigdy klucza `Zielona-Root-CA` ani `Zielona-CA01`. Klucza urzędu nie eksportujemy poza urząd: step-ca ma własny klucz urzędu pośredniego, a podpis pod nim składa urząd główny na ceremonii.

Droga bez Microsoftu

Ta strona w ogóle nie wymaga Microsoftu: Let's Encrypt, certbot, Traefik i step-ca działają tak samo w firmie z Google Workspace i serwerami na Rocky Linux. Microsoft wchodzi tylko w jednym miejscu: gdy serwer z Windows i IIS potrzebuje certyfikatu publicznego, używamy klienta ACME dla Windows (na przykład win-acme albo Posh-ACME) zamiast certbota, a wewnętrznie dalej automatycznej rejestracji z AD CS.

Częste błędy

Objaw	Przyczyna	Co robimy
`certbot renew --dry-run` kończy się `NXDOMAIN looking up TXT for _acme-challenge`	rekord nie zdążył się rozejść albo klucz TSIG nie ma praw do strefy	dłuższe `--dns-rfc2136-propagation-seconds`, test `nsupdate` z tym samym kluczem
Certyfikat odnowiony, przeglądarka pokazuje stary	usługa nie przeładowała certyfikatu	`--deploy-hook` z przeładowaniem usługi
Let's Encrypt odmawia z `too many certificates`	limit wydań dla tej samej nazwy przy testach	testy zawsze z `--dry-run` albo `--test-cert` (staging)
Klient odrzuca certyfikat z `acme01`	brak `Zielona-Root-CA` w magazynie albo serwer nie podaje certyfikatu pośredniego	`fullchain.pem` zamiast `cert.pem` w konfiguracji usługi, `update-ca-trust` na kliencie
step-ca odmawia wydania dla `sklep.inna-firma.example`	nazwa poza listą dozwolonych w `policy`	to zamierzone; certyfikat dla innej domeny bierzemy z publicznego urzędu
Po lutym 2027 certyfikaty z Let's Encrypt wygasają przed odnowieniem	zadanie odnawiające „co 60 dni” przy ważności 64 dni	timer certbota albo klient z ARI zamiast własnego harmonogramu
Serwer poczty partnera przestał przyjmować nasz certyfikat klienta	certyfikat publiczny bez zastosowania uwierzytelniania klienta (od lutego 2026)	certyfikat klienta z urzędu wewnętrznego albo uzgodnienie z partnerem innego sposobu uwierzytelnienia

□□ Ćwiczenia

1. Marek chce certyfikat dla panelu WMS dostępnego tylko w sieci magazynu pod nazwą `wms.zielona.example`. HTTP-01 czy DNS-01 i dlaczego?

Odpowiedź

DNS-01, bo Let's Encrypt nie dotrze z internetu do serwera w sieci magazynu, a DNS-01 sprawdza tylko rekord w publicznej strefie. Trzeba jednak pamiętać, że nazwa trafi do

dzienników Certificate Transparency; jeśli to przeszkadza, certyfikat bierzemy z `acme01` albo z `Zielona-CA01`.

1. Klucz TSIG z pliku `rfc2136.ini` wyciekł z kopii serwera. Co może zrobić napastnik i co robimy?

Odpowiedź

Może zmieniać rekordy w strefie `zielona.example` w zakresie praw klucza, więc także wystawić sobie certyfikat z Let's Encrypt dla naszych nazw albo przekierować ruch. Natychmiast unieważniamy klucz u dostawcy DNS, wydajemy nowy z najwęższymi prawami (tylko `_acme-challenge`), sprawdzamy strefę i dzienniki Certificate Transparency pod kątem certyfikatów, których nie zamawialiśmy.

1. Ile razy w roku odnowi się certyfikat Let's Encrypt z profilu `classic` od 16 lutego 2028 roku, jeśli klient odnawia po dwóch trzecich ważności?

Odpowiedź

Ważność 45 dni, odnowienie po około 30 dniach, czyli około dwunastu razy w roku. Przy ręcznym odnawianiu to dwanaście okazji do pomyłki, przy ACME zero.

1. Ola pyta, czy `acme01` może wydawać certyfikaty do VPN z EAP-TLS, żeby „wszystko było w jednym miejscu”. Co odpowiadamy?

Odpowiedź

Nie: VPN i Wi-Fi z NPS logują przez domenę, a kontroler przyjmuje do logowania tylko certyfikaty z urzędów w NTAuth z silnym mapowaniem, czyli z `Zielona-CA01`. `acme01` ma ograniczenie do nazw serwerów i służy certyfikatom TLS serwerów; rozdzielenie ról jest tu zaletą, nie wadą.

Powiązane: Projekt PKI Zielonej: dlaczego nie na kontrolerze, hierarchia dwupoziomowa, nazwy, ważność, CRL i AIA · Reverse proxy w Dockerze: Traefik przed wszystkimi aplikacjami na docker01 · Certyfikaty SSL: wystawienie, konwersja, weryfikacja · MailCow: codzienna praca z domenami, skrzynkami i spamem · Inwentarz certyfikatów i częste problemy: wygasła CRL, łańcuch, SAN, alarm w Zabbixie · Instalacja serwera FreeIPA na ipa01

Źródła: (odczyt 20 do 26.10.2026) · support.dnsimple.com, „SSL Certificate Validity Changes (2026 - 2029)” (SC081v3, kwiecień 2025: 200, 100, 47 dni) · knowledge.digicert.com, „Domain

validation reuse changes in 2026” (398, 200, 100, 10 dni) · isrg.org, „From 90 to 45” (2.12.2025: profile `tlsserver`, `classic`, `shortlived`, daty 13.05.2026, 10.02.2027, 16.02.2028, ARI) · letsencrypt.org, „DNS-PERSIST-01” (18.02.2026, SC-088v3 z października 2025, plan produkcji na II kwartał 2026) · letsencrypt.org, „Ending TLS Client Authentication” (14.05.2025: 11.02.2026 i 8.07.2026) · isrg.org, „Ending OCSP Support in 2025” (adresy OCSP usunięte 7.05.2025, serwery wyłączone 6.08.2025) · RFC 8555 · eff-certbot.readthedocs.io, wtyczka `certbot-dns-rfc2136` i `certbot reconfigure` · doc.traefik.io, „ACME DNS challenge” i go-acme.github.io/lego, dostawca `rfc2136` (nazwy zmiennych do potwierdzenia w naszej wersji) · smallstep.com/docs, „Installation” (repozytorium RPM), „Run step-ca as an intermediate under an existing root” (`step certificate create --csr`, podpis szablonem SubCA w AD CS), „ACME basics” i „Certificate issuance policies” · github.com/smallstep/certificates/releases (0.30.2, 23.03.2026) · cvefeed.io, CVE-2026-30836 (SCEP w step-ca przed 0.30.0)

Revision #2
Created 2026-10-26 14:06:06 UTC by Tomasz Konieczny
Updated 2026-10-26 14:28:26 UTC by Tomasz Konieczny