Unlocking Automated Certificate Management for 5G Core Networks with ACME

IIT RTC 2025

Charles Eckel, Cisco Systems
eckelcu@cisco.com

October 7, 2025

A secure connection is not actually secure if you do not know who it is with

A secure connection is not actually secure if you do not know who it is with

Agenda

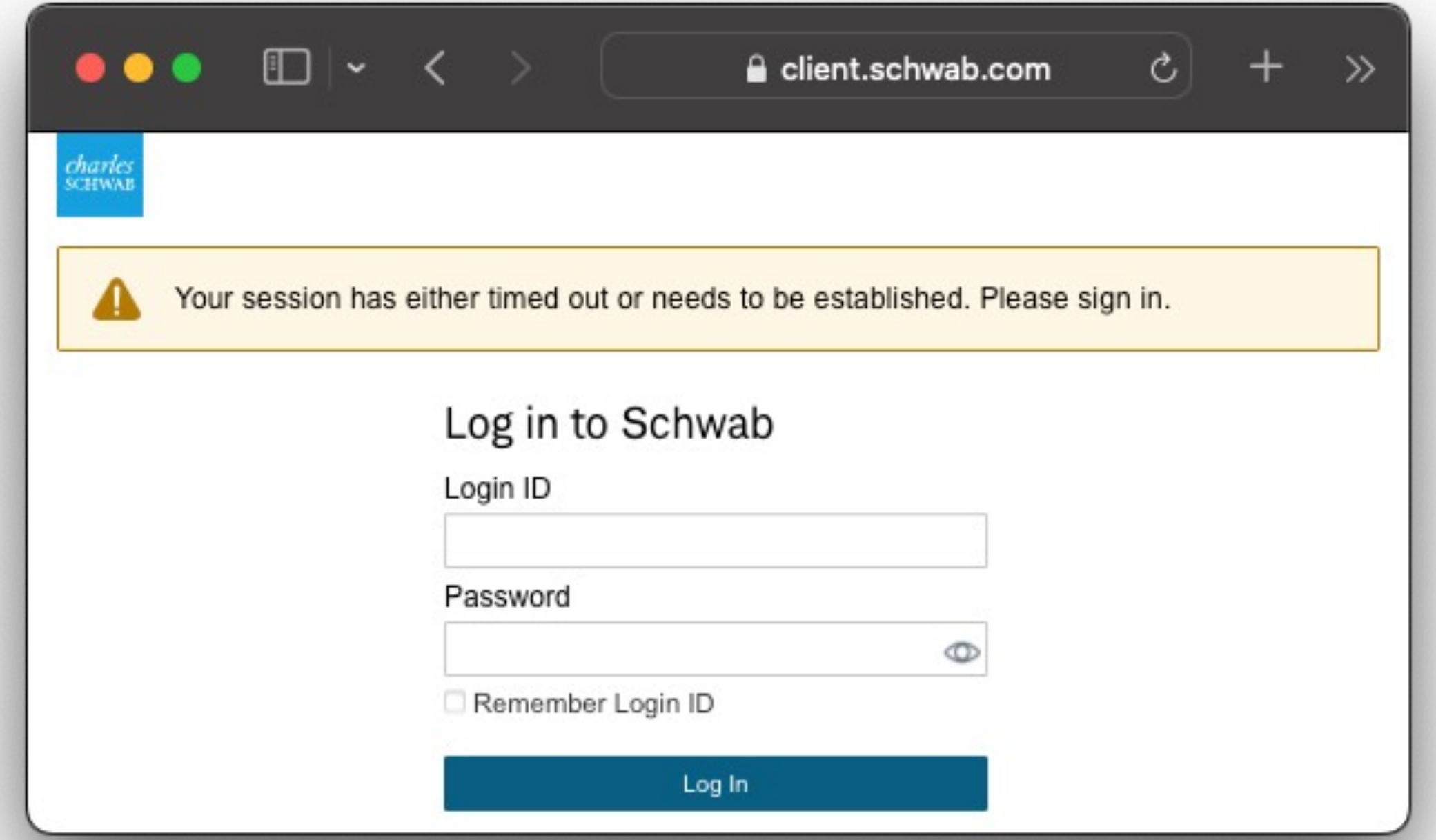
01 ACME

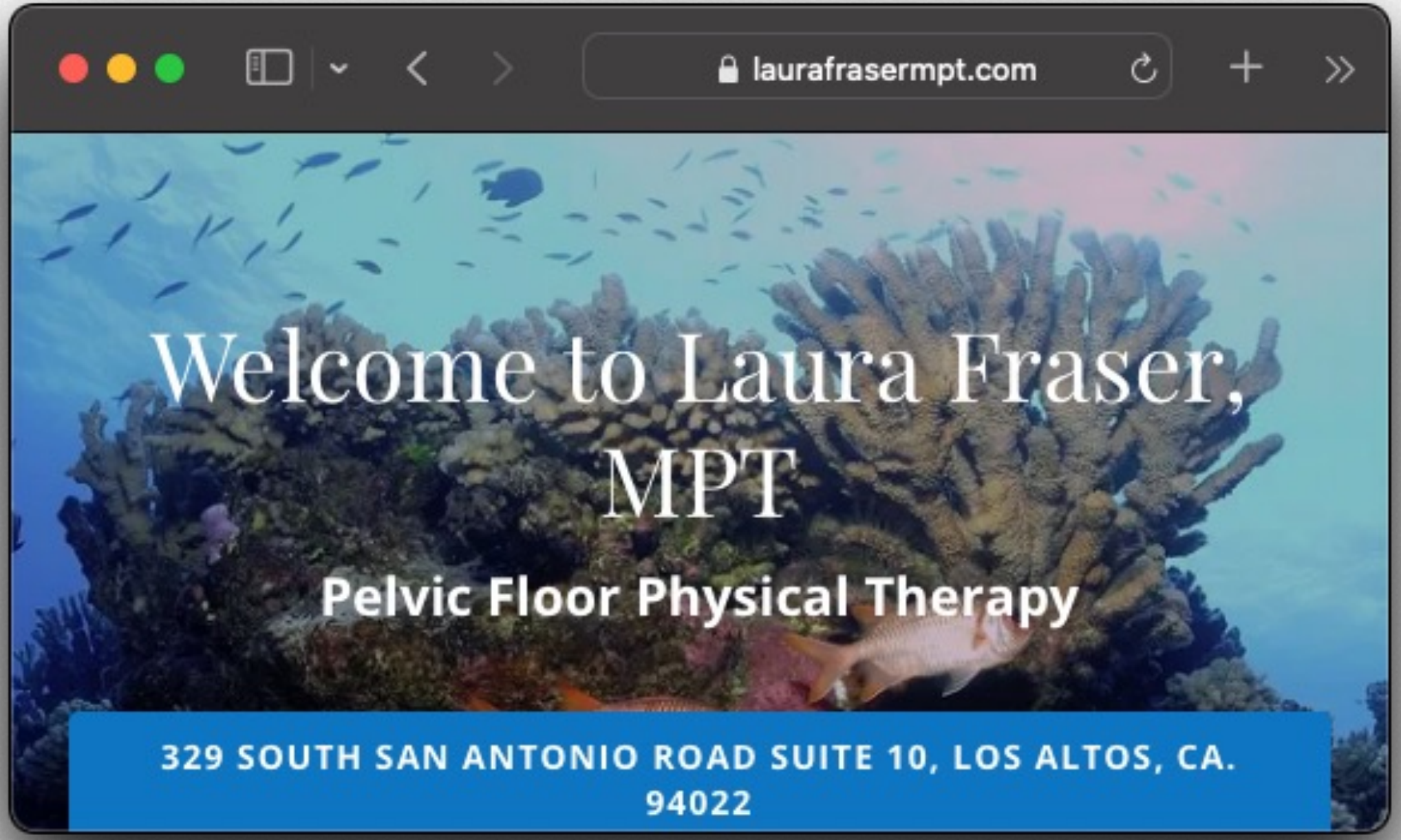
02 Certificate Management for the 5G Core Networks

03 ACME for the 5G Core Networks

04 Key takeaways and next steps

ACME





The world's largest certificate authority serving 280 million websites with free, automated TLS certificates. Let's Encrypt has dramatically changed the privacy and security of the Web for practically everyone using it.

Free, Automated, and Standard

Previously, certificates were either manual or via proprietary APIs

- ... and to use the API, you had to pay
- Low usage, low automation

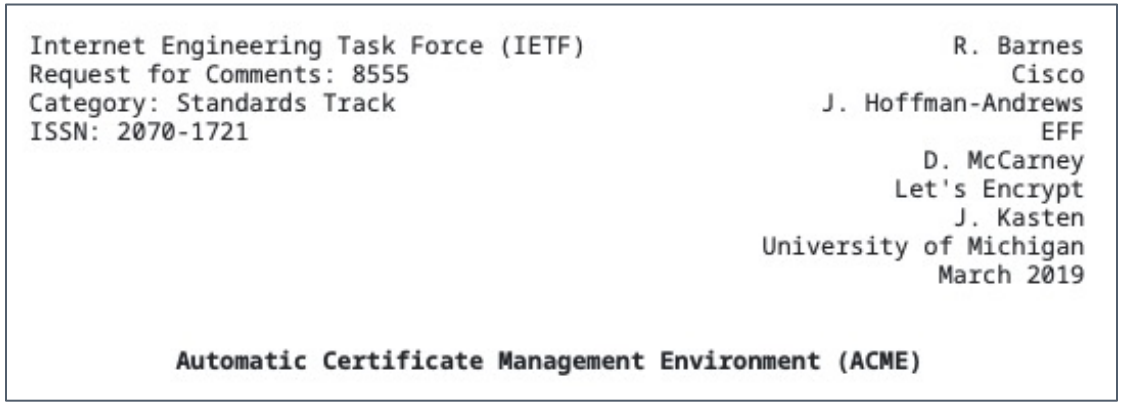
Let’s Encrypt wanted to be free, so they had to automate

- Can’t afford staff to support manual issuance
- Need to encode all the CA interactions in an API

Side goal of automation for the whole web

- Submitted IETF draft of API at the same time as launch

RFC 8555: Automatic Certificate Management Environment (ACME)



Internet Engineering Task Force (IETF)
Request for Comments: 8555
Category: Standards Track
ISSN: 2070-1721

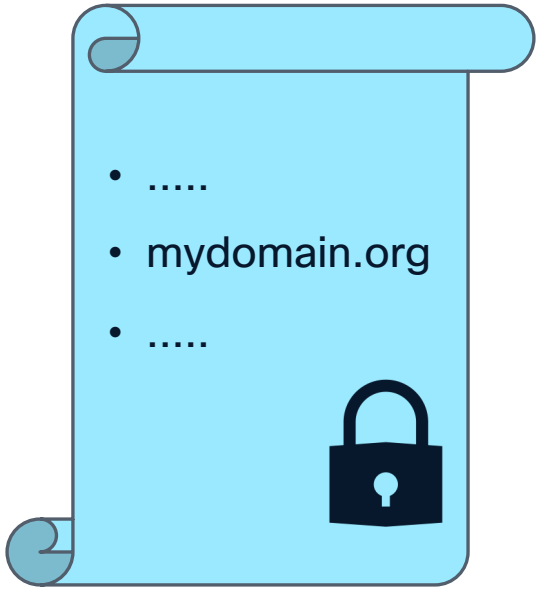
R. Barnes
Cisco
J. Hoffman-Andrews
EFF
D. McCarney
Let's Encrypt
J. Kasten
University of Michigan
March 2019

Automatic Certificate Management Environment (ACME)

- Automates process of obtaining, renewing, and revoking X.509 certificates
- Eliminates manual intervention in certificate lifecycle management
- ACME Client: Software on user's server or device (e.g., web server, mail server) that uses ACME protocol to request certificate management actions
- ACME Server: Operated by a Certificate Authority (e.g., Let's Encrypt), responds to client requests, and performs requested actions (e.g., issue or revoke certificate)

Challenge types

Used by ACME Server to challenge ACME Client to prove it is authoritative for the identities associated with the requested certificate.



dns-01 challenge type

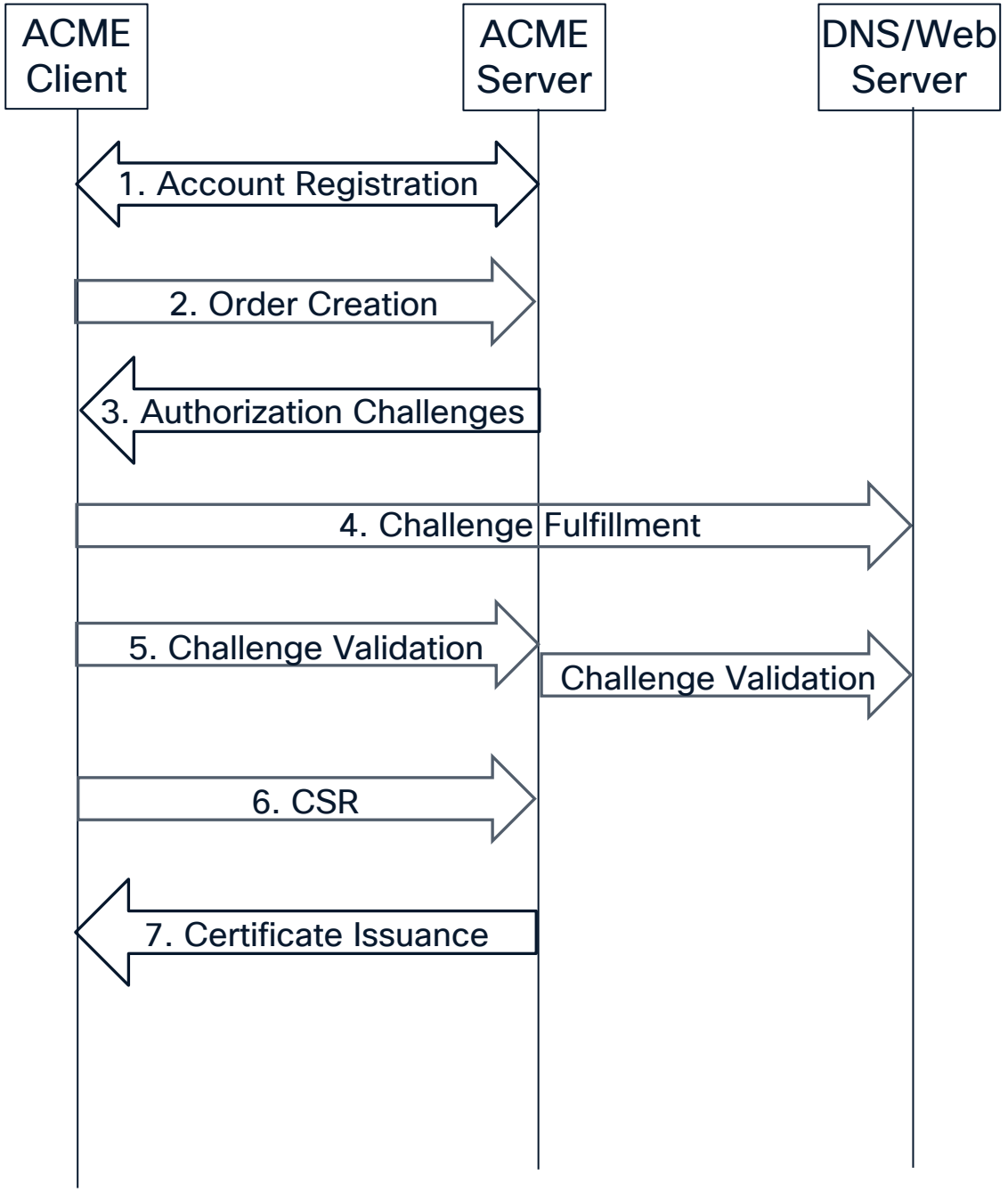
- Domain Name System (DNS is essential component for the dns-01 challenge type.
- ACME Server uses DNS to verify domain ownership by looking up specific TXT records that ACME Client is instructed to provision.

http-01 challenge type

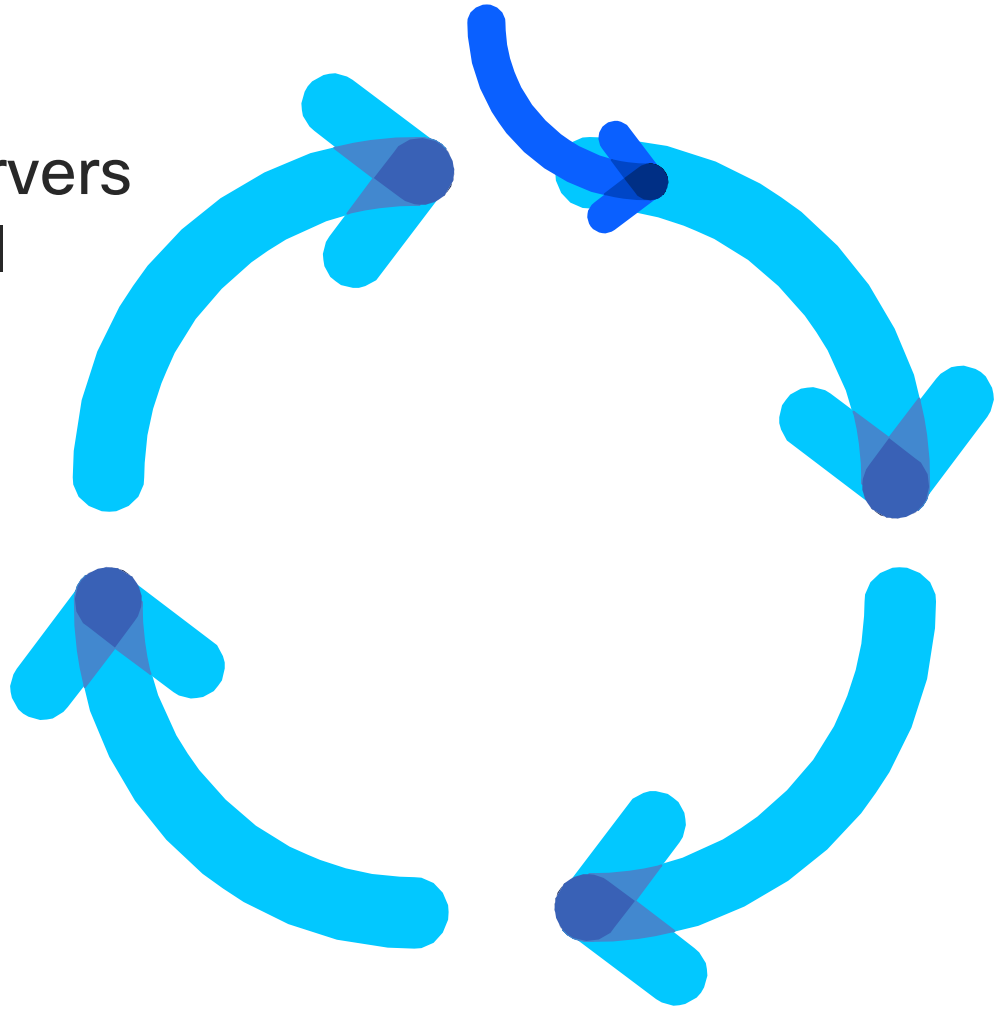
- Web Server: Used for the http-01 challenge type.
- ACME Client provisions a specific HTTP resource on the domain's web server, which ACME Server then accesses to confirm domain control.

Protocol flow

1. Account Registration: ACME Client registers account with ACME Server using a generate account key pair that is used to authenticate all subsequent requests.
2. Order Creation: ACME Client requests a certificate for one or more identifiers (e.g., domain names) via an "order" to ACME Server.
3. Authorization Challenges: ACME Server provides "authorization challenges" to ACME Client to prove control over the identifiers in the order.
4. Challenge Fulfillment: ACME Client performs the necessary actions to fulfill the chosen challenge.
5. Challenge Validation: ACME Client notifies ACME Server that the challenge is ready for verification; ACME Server performs a lookup or request to verify.
6. CSR: If ACME Server successfully validates control over all identifiers, the ACME Client can then finalize the order and send the certificate signing request (CSR).
7. Certificate Issuance: The ACME Server processes the CSR and issues the X.509 certificate to the ACME Client.
8. Certificate Deployment: The ACME Client receives the issued certificate and deploys it to the appropriate server (e.g., web server, mail server) to enable secure communication (e.g., HTTPS).



Free 🤑



Certificates available with standard API

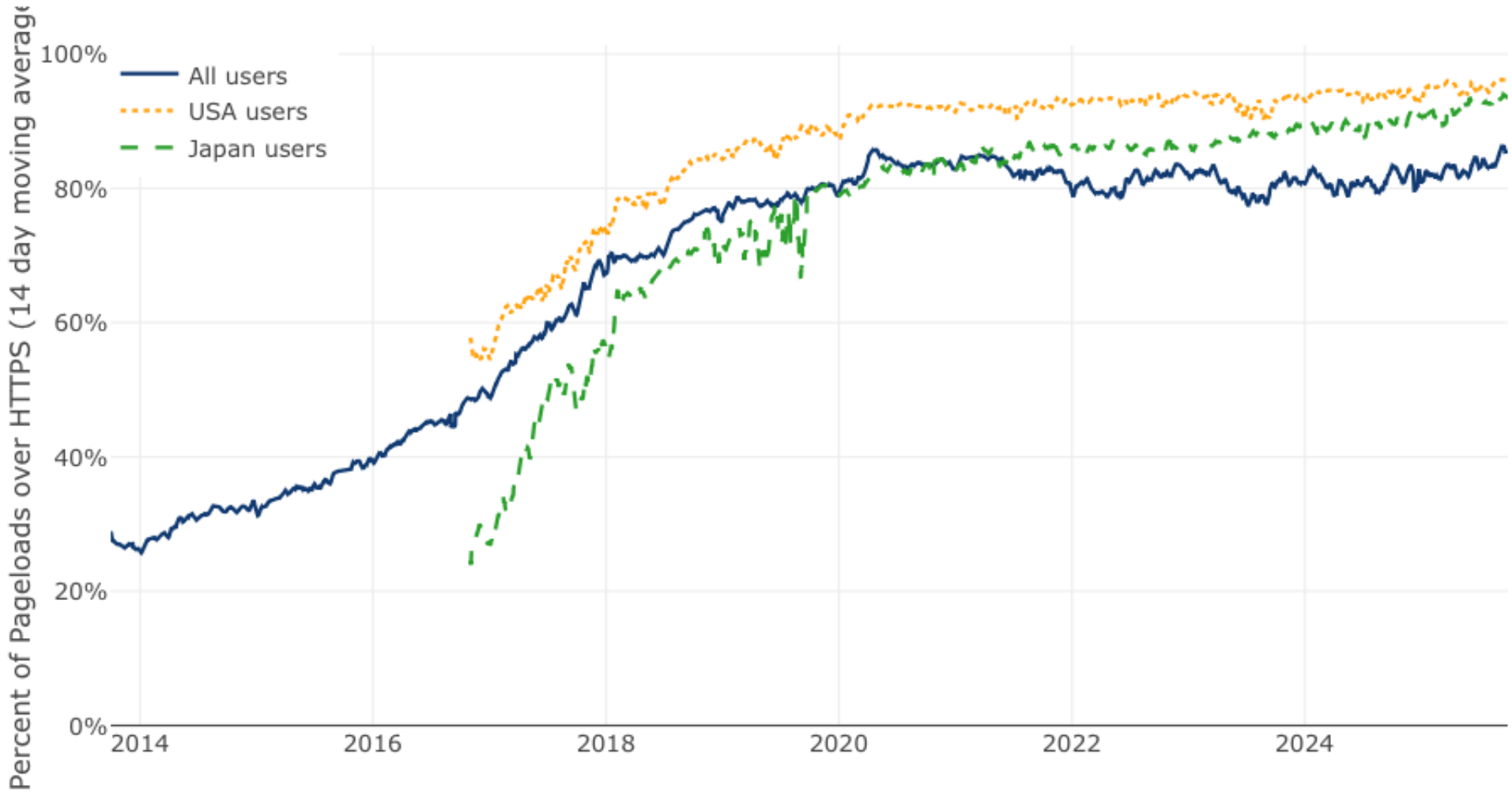
People build client tools to the standard API

More websites use the standard API

More CAs build servers to the standard API

HTTPS for the Whole Web

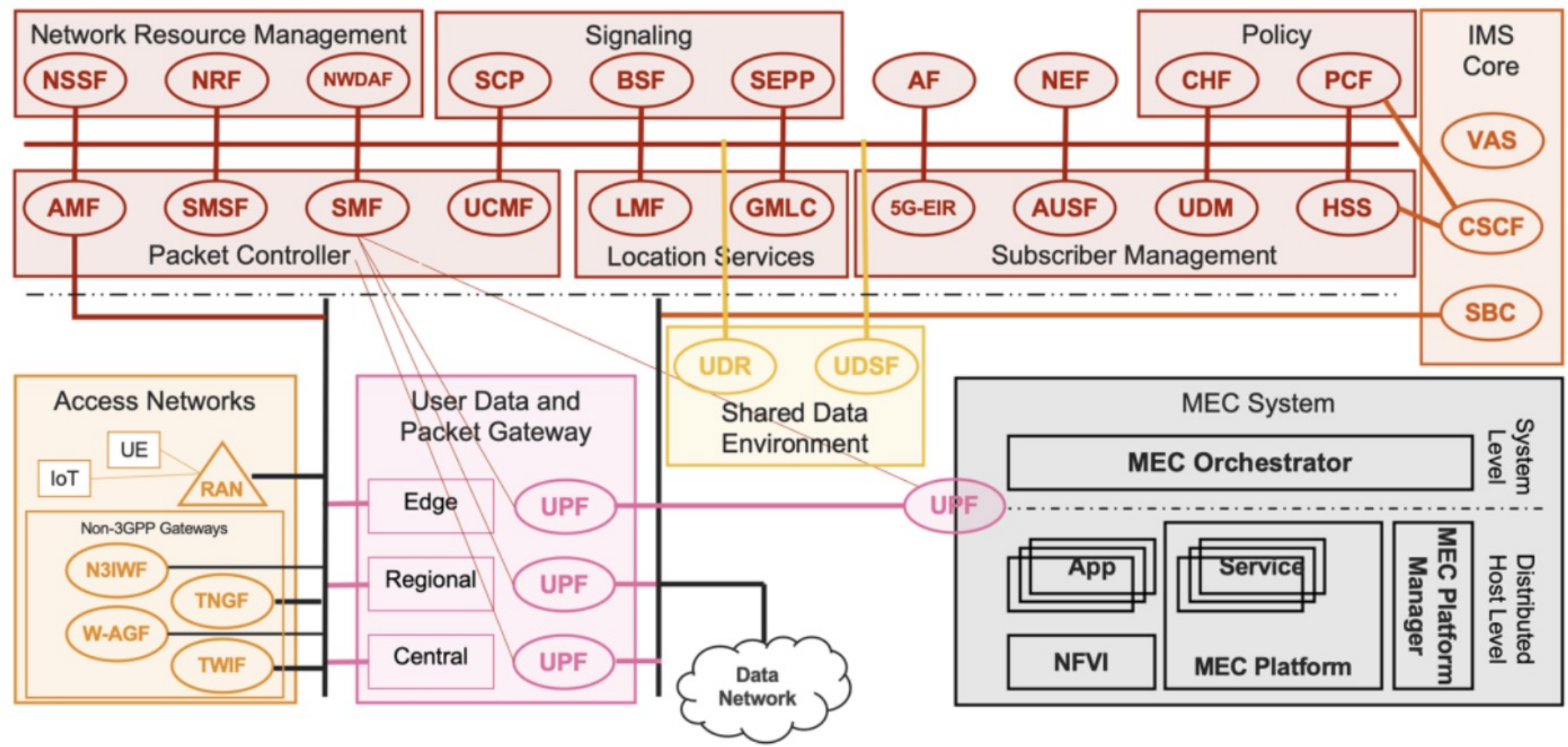
Source: Firefox Telemetry: https://docs.telemetry.mozilla.org/datasets/other/ssl/reference.html



Before: **30%**
After: **90%**

Certificate Management for 5G Core Networks

5G Core Service-Based Architecture (SBA)



5G Core Service Based Architecture with MEC System and with IMS Core
Source: Marin Ivezic, https://5G.Security

Centre for Risk Studies
UNIVERSITY OF CAMBRIDGE Judge Business School

CENTRE FOR RISK STUDIES
VIEWPOINTS

Expired certificates to blame in O2 outage

7 DEC 2018

by Jennifer Daffron and Kelly Quantrill

The nearly 24-hour outage of the O2 4G network that affected O2's 25 million customers has been traced to an expired certificate in a software used by their supplier Ericsson.

What exactly does this mean? What kind of certificate has the power to impact individuals in over 11 countries including the UK and Japan? Likely, it was a TLS Certificate.

Search

Subscribe by email

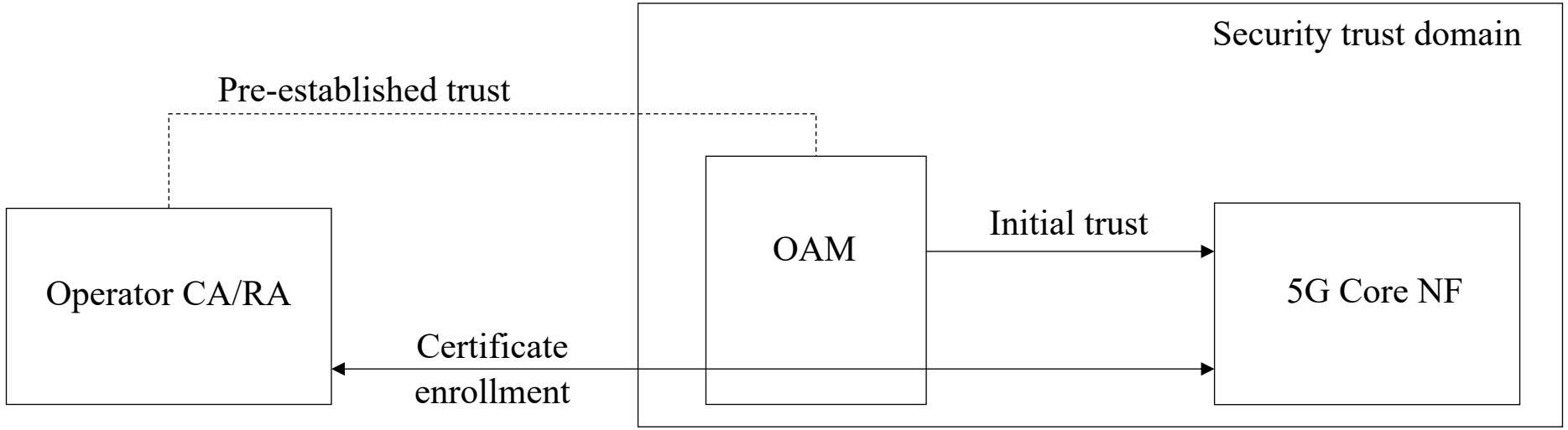
You may manage your subscription options from your profile.

About us

VIEWPOINTS is informed by the research being carried out at the Centre for Risk Studies at the University of Cambridge Judge Business School.

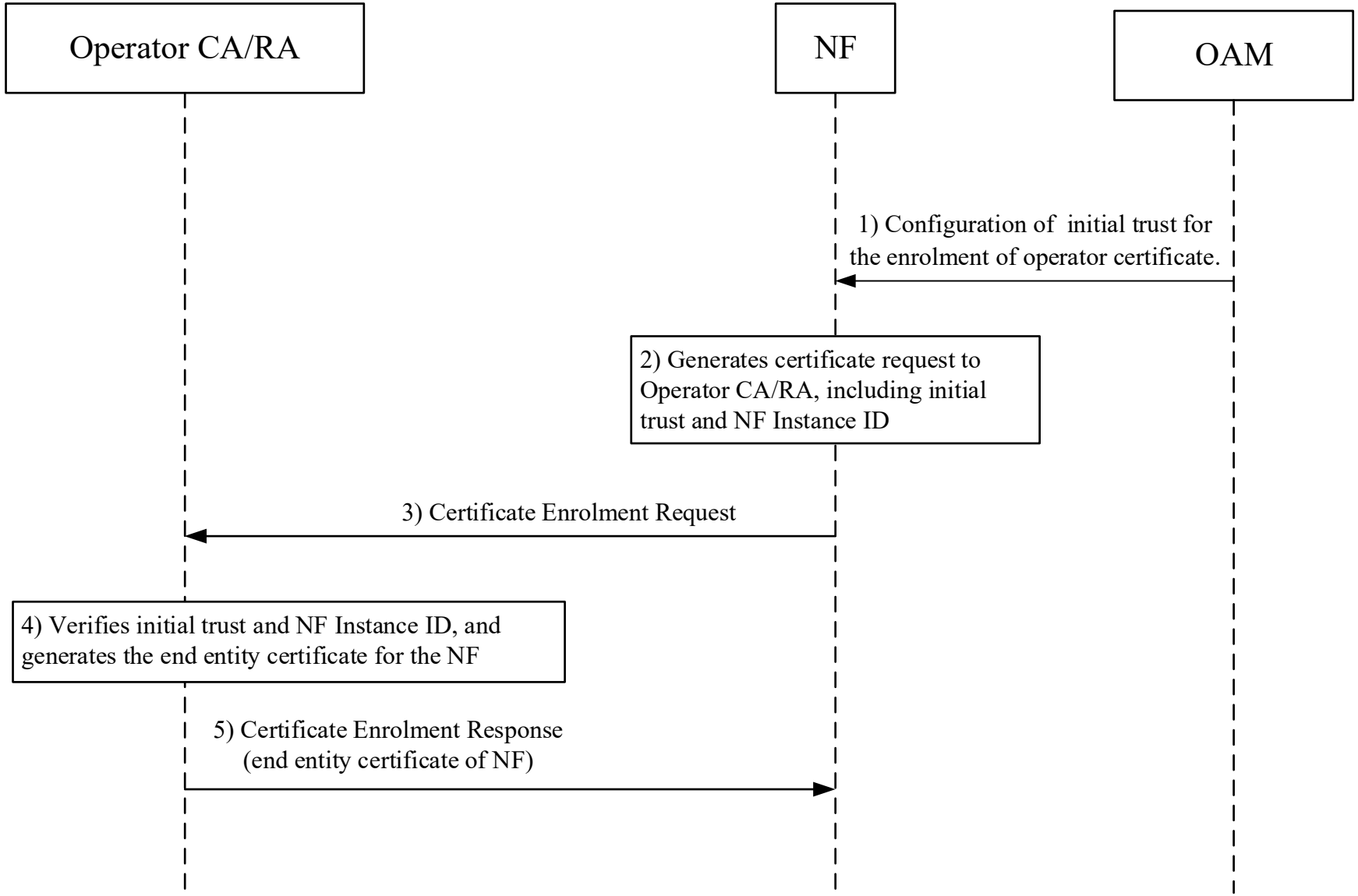
Certificate management for 5GC NFs

3GPP TS 33.310, Release 18

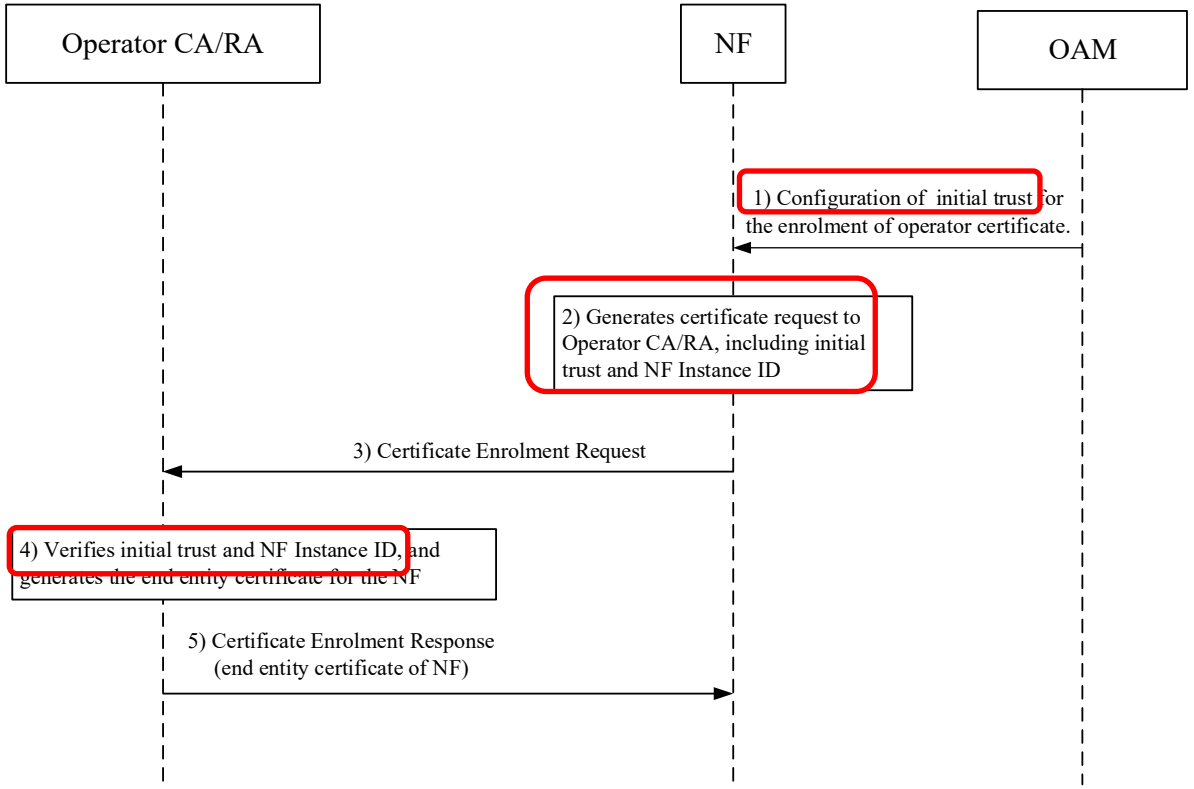


Certificate management for 5GC NFs

3GPP TS 33.310, Release 18



BIG step in the right direction, BUT



- Defined support for CMPv2 only
- CMP, and CMPv2, was not designed with automation in mind
- Three different initial trust mechanisms and all are underspecified
- How Operator CA/RA verifies NF Instance and the authority of the NF over it is left to implementation
- Works within a single vendor environment, but challenging to expand to incorporate 3rd party NFs

ACME for 5G Core Networks

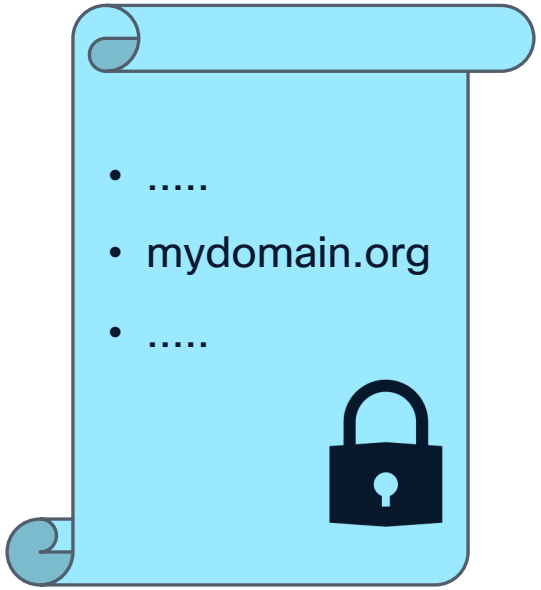
Justification

Why do we need another protocol

- 5G SBA is secured using certificates across all SBA components and corresponding NFs
- Virtualization and modularity of NFs has led to increase of multi-vendor environments
- Increasing common to deploy NFs from multiple vendors within cloud native environment from yet another vendor, all of which are independent of the CA that is authoritative for the certificates used to secure their communications
- In such deployments, it is impractical to manage certificates manually
- Release 18 work defined the use of CMPv2 for automated certificate management for SBA
- ACME is particularly well suited when considering NFs deployed on cloud native platforms (e.g., Kubernetes), which have built-in support for ACME
- An important benefit of ACME is automated validation of authority to represent an identifier (i.e., to be authoritative for the resource for which the certificate is issued)

Challenge types

Used by ACME Server to challenge ACME Client to prove it is authoritative for the identities associated with the requested certificate.



dns-01 challenge type

- Domain Name System (DNS is essential component for the dns-01 challenge type.
- ACME Server uses DNS to verify domain ownership by looking up specific TXT records that ACME Client is instructed to provision.

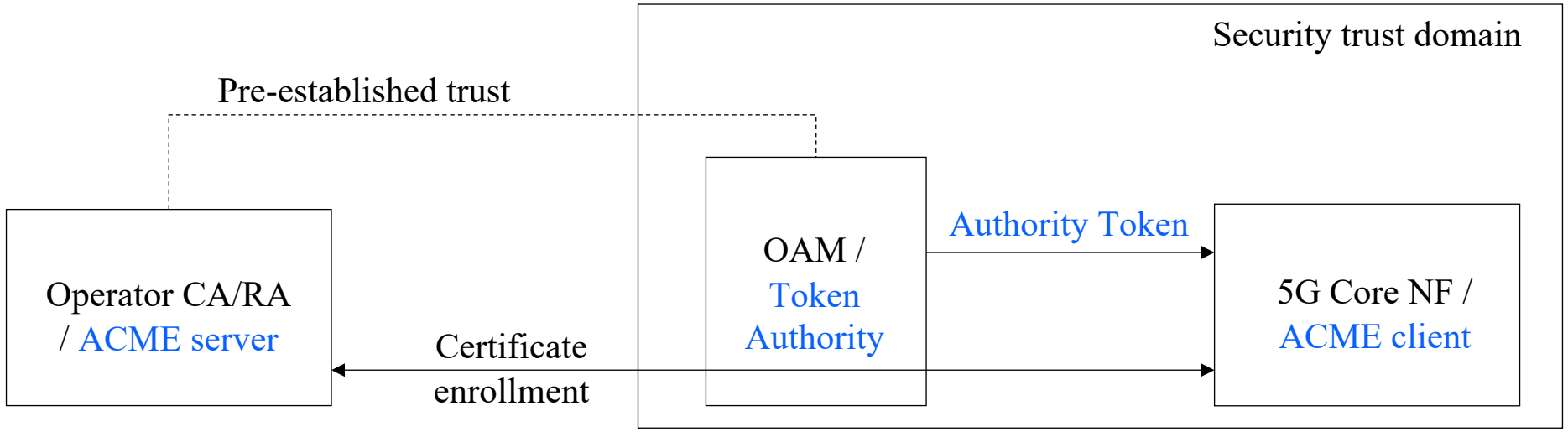
http-01 challenge type

- Web Server: Used for the http-01 challenge type.
- ACME Client provisions a specific HTTP resource on the domain's web server, which ACME Server then accesses to confirm domain control.

ACME Authority Token challenge type, “tkauth-01”

- Defined in IETF RFC 9447: "Automated Certificate Management Environment (ACME) Challenges Using an Authority Token”
  - Assumes a trust relationship between a CA and a Token Authority, i.e., that a CA is willing to accept the attestation of a Token Authority for particular types of identifiers as sufficient proof to issue a credential
  - When using ACME, the OAM system acts as a Token Authority that is trusted by the Operator CA/RA. As such, the OAM is trusted to act as the authority for the NF Instance ID namespace within the 5GC.
- New identifier type
  - A new ACME identifier type, "NfInstanceId", is defined. A NF uses its NF Instance ID as the value of the “NfInstanceId". The format of the value of the "NfInstanceId" is as defined in 3GPP TS 29.571.
  - NfInstanceId: string: String uniquely identifying a NF instance. The format of the NF Instance ID shall be a Universally Unique Identifier (UUID) version 4, as described in IETF RFC 4122. The hexadecimal letters should be formatted as lower-case characters by the sender, and they shall be handled as case-insensitive by the receiver.
  - Example: "4ace9d34-2c69-4f99-92d5-a73a3fe8e23b"

Certificate management for 5GC NFs with ACME



- NF is ACME client, Operator CA/RA is ACME server, and OAM system is Token Authority
- OAM system instantiates NF, provides it with initial trust needed for ACME account creation on Operator CA/RA
- NF instance ID, which uniquely identifies the NF within the 5GC, is assigned to the NF by the OAM system as part of its NF profile, as specified in TS 23.502
- NF profile parameters signed by the OAM and included in the Authority Token includes the NF Instance ID

3GPP TS 33.310 v19.5.0, includes support for ACME

- Annex J: This annex describes the requirements to support ACME as an automated certificate management protocol for 5GC NFs. The NF acts as an ACME client and the operator CA/RA acts as an ACME server. The overall certificate management procedure follows IETF RFC 8555.
- IANA registrations
  - New ACME Identifier Type
    - Label: NfInstanceId
    - Reference: 3GPP TS 33.310
  - New ACME Validation Method
    - Label: tkauth-01
    - Identifier Type: NfInstanceId
    - ACME: Y
    - Reference: 3GPP TS 33.310

Key takeaways and next steps

Interoperable automated certificate management

- Standard use of ACME defined for certificate management for 5G SBA
- One initial trust mechanism that is well defined and aligned with existing 3GPP specification
- One fully specified challenge type and validation mechanisms for Operator CA/RA to verify authority of the NF over its NF Instance ID
- Proven track record for ACME protocol to work in multi-vendor environments and scale to large deployments, e.g., the Internet
- Well suited for cloud native deployments where support for ACME is integrated into lifecycle management of containerized NFs

- Next steps: Running code, preferable open source (e.g., Boulder, OpenSSL, EJBCA Community Edition, Hashicorp Vault, ...)

Thank you!


CISCO