ACME-Based Provisioning of IoT Devices

Michael Sweet
Lakeside Robotics Corporation
March 30, 2023
ACME-Based Provisioning of IoT Devices

• Current I-D:
  • https://datatracker.ietf.org/doc/draft-sweet-iot-acme/

• Abstract:
  • This document extends the Automatic Certificate Management Environment (ACME) [RFC8555] to provision X.509 certificates for local Internet of Things (IoT) devices that are accepted by existing web browsers and other software running on End User client devices.

• Goal is to eliminate scary browser security warnings when accessing embedded web servers
Typical Home Network

- Wi-Fi router/modem provided by ISP
  - Router implements DHCP and DNS (passthrough) services along with NAT and firewall functionality
  - Little to no outbound traffic filtering, may provide inbound port mapping and/or DMZ functionality for a single host
  - Embedded web interface for configuration/status monitoring, speed testing, etc.

- Network clients connect to network and obtain IP address(es), default gateway/route, DNS server, and local domain (usually the ISP's domain name) via DHCP

- Printers, cameras, appliances, etc. provisioned/connected by end users using WPS, captive portal AP web interface, vendor mobile apps, and/or device control panel
Typical Enterprise Network

- Managed routers, modems, etc.
  - Multiple subnets/VLANs
  - DHCP service for each subnet
  - DNS service for each site/organization
  - Outbound traffic is filtered/monitored, inbound traffic may be completely blocked or limited to isolated subsets/VLANs, interior traffic is often filtered/monitored

- Dedicated authorization, certificate, etc. services

- Network clients may need to be explicitly provisioned

- Printers, cameras, appliances, etc. are managed by IT department and/or third-party service
Let's Encrypt has enabled the widespread use of HTTPS for public Internet web sites

- Certificates for ".local" domain names cannot be issued
- No way to do HTTP or DNS verification of local devices

A local ACME server can be configured to issue certificates for ".local" domain names (as well as site domains) and can do HTTP and DNS verification with local devices

Key issues:
- ACME server discovery
- Root certificate (trust anchor) for issued certificates
- Security considerations for local ACME server, clients, and IoT devices
ACME Server Discovery

- Use DHCP option and/or DNS-SD with local DNS service
  - DHCP is both commonly used and trusted for local device access/connection
  - DHCP option provides simpler way for home networks
  - DNS-SD integrates with enterprise infrastructure
  - *Cannot use mDNS for security reasons*
- Nominally one ACME server per network
  - Failover/load-balancing is possible via DNS but from the network device perspective there is a single service

![Diagram showing DHCP, DNS-SD, ACME Certificate Request, Certificate Response, and HTTP Verification]

1. DHCP
   - IP Addresses
   - DNS Server
   - Hostname
   - Domain Name (ACME Server)
2. DNS-SD
   - ACME Server
3. ACME Certificate Request
4. ACME HTTP Verification
5. ACME Certificate Response
Network Root Certificate ("Trust Anchor")

- CA-signed root certificate will work with existing CA infrastructure/support
- Self-signed root certificate requires some special handling
  - Trust On First Use (TOFU) when connecting to network
  - Only valid while connected to that network
  - Only valid for ".local" and local/site-specific domain
Security Considerations

- Local ACME server:
  - Only issue certificates for approved domains - ".local" and site-specific domain ("examplecorp.com")
  - Protect root certificate and private key
  - Support revocation/re-issue as needed
  - Long-duration self-signed root cert or CA-signed root cert to minimize time-of-use/MITM attacks
  - Short-duration issued certs to minimize exposure of compromised credentials
Security Considerations

- **Client devices:**
  - Limit trust of local root certificate to current network/domains
    - Challenge is network identification - SSID isn't unique, MAC address should be but isn't authenticated, TLS negotiation establishes ownership of private key but anybody can make a self-signed certificate
  - TOFU for "self-signed" root certificates

- **IoT devices:**
  - Protect ACME-issued certificate and generated private key
  - Do not reuse private keys
  - Ability to wipe/"factory reset" device
Level of Trust

• With self-signed root certificates, the level of trust is necessarily reduced
  • Browsers could choose to indicate this somehow, but based on prior research with "EV" certs that might not be useful/effective
  • Might also simply change the wording of the scary error message to something more like the SSH TOFU prompt

• Need to be comfortable with "less than perfect" security
  • Encrypting potentially sensitive communications is always a good thing
  • Providing a network-specific trust anchor provides better control and local validation of certificates and protects against MITM attacks
  • Using ACME allows the same certificate infrastructure to be used in both enterprise and home networks, which is especially important as more people have hybrid work scenarios