Recap on Recent Work: Use Cases and Requirments

HTTPS in Local Network CG F2F @ TPAC 2018, 10/25

Ryo Kajiwara @ ACCESS

Problem Description: Why do we need httpslocal?

Everything is becoming https

https (TLS) relies on certificates

Certificates rely on global DNS

Global DNS cannot provide names for hosts under local network

Thus, no https in local network

(Unless you use self-signed certs and add the certificate of your self-hosted CA to the root trust store)

What does "No HTTPS in local network" mean?

Publicly-servicing Web services cannot leverage the capability of local devices.

- Mixed Content when loading content from local storage devices
- Browser APIs that require HTTPS cannot be used against locally hosted servers

"Just have your customers install your root CA's certificate" is not a solution

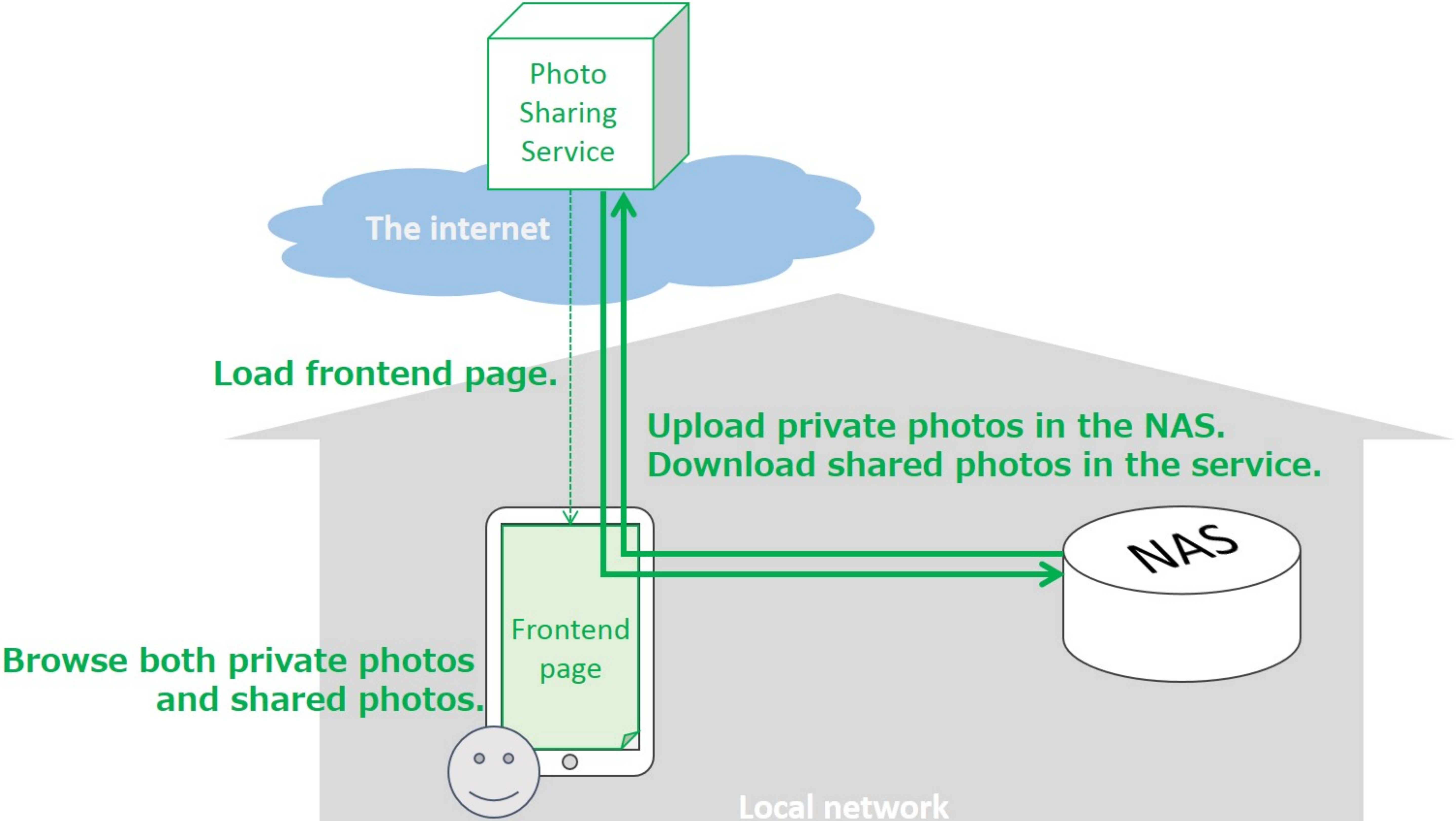
GitHub repo for use cases

https://github.com/httpslocal/usecases

https://github.com/httpslocal/usecases/blob/master/UseCases.md

Scenario Overview:

1. Direct Access from UA
2. Machine-to-Machine
3. Cross-Origin



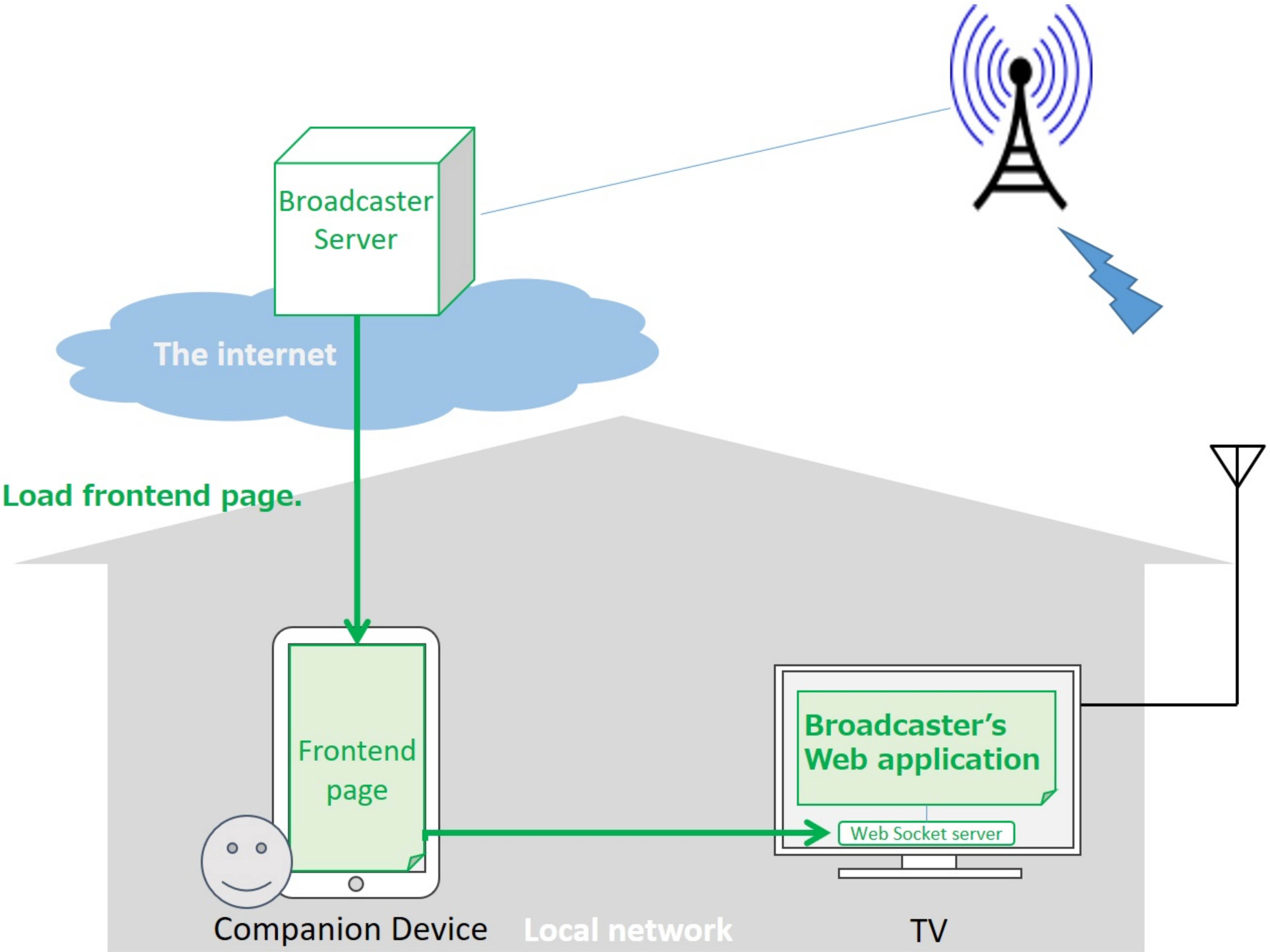
Load frontend page.

Upload private photos in the NAS.
Download shared photos in the service.

Browse both private photos
and shared photos.

UC-07: Secure Offline Communication for Home Automation

- User sets up a home gateway
- This use case is relevant since this case does not involve a browser; In this case the mediator between the Internet and the local network is the gateway


Broadcaster Server
The internet
Load frontend page.
Frontend page
Companion Device
Local network
Broadcaster's Web application
Web Socket server
TV

Requirements

https://github.com/httpslocal/usecases/blob/master/Requirements.md

For use cases that involve Device Discovery

(Outdated; narrow down items that are too specific to a certain use case) https://github.com/httpslocal/usecases/blob/master/Certificates.md

Checklist of pros and cons based on types of certificates

Public CA Certificate for devices accessible globally

- Give local devices public names and issue certificates to them. cf. Mozilla's "Things Gateway"
- Pros: No need to modify UA implementation/PKI, Can use ACME to automate certificate issuance
- Cons: Does not work when Internet connection is down, Domain name of the device will be publicly disclosed, The device is reachable from the Internet

Public CA Certificate for devices accessible only in the local network

- Delegated Credentials / STAR certificates / PLEX. Details will be shown in the later presentation.
- Pros: No need to modify UA implementation/PKI.
- Cons: Does not work when Internet connection is down, Domain name of the device will be publicly disclosed.

Private CA Certificate

- ".local" Server Certificate for HTTPS migration on local network @ TPAC 2016
- Pros: Works without Internet connection, Domain name not disclosed globally
- Cons: UA implementation or PKI has to be reworked/extended.

Self-Signed Certificate

- Using CORS-preflight request + Access-Control-{Request,Allow}-External proposed in CORS and RFC1918, user gives trust to certain origin explicitly through the UA
- Pros: Works without Internet connection, Domain name not disclosed globally
- Cons: UA implementation or PKI has to be reworked/extended, Forces the responsibility of trusting the certificate onto the user?

Moving On:

Is this even a thing? (Missing use cases? Scope too large?)

How can we make it a thing?