ACME
Extension for Public Key Challenges

draft-geng-acme-public-key-01

Feng Geng, Panyu Wu, Liang Xia
Huawei

IETF 122

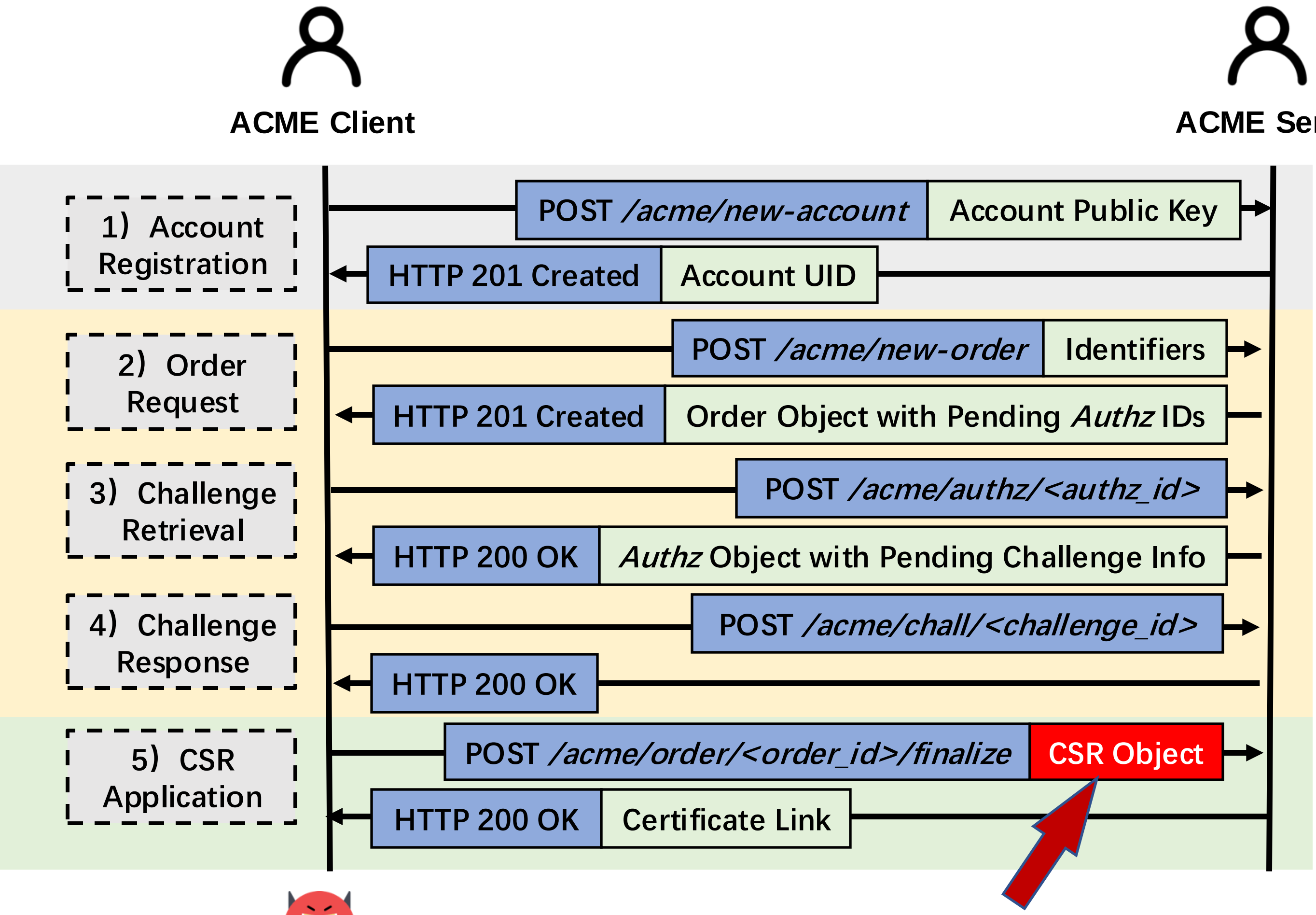
Summary

Present ACME pk-01 challenge

- **Identify** the Public Key replacement risk of the ACME Phase 2 and Phase 3;
- e.g. Risks on **ACME client deployed on EMS**(element management system) to apply certificates for managed network element;
- **Defending against Public Key replacement attacks;**
- An optional process for removing CSR;
- For user/device to apply for their own certificates, it can be restricted by adding whitelisting measures;
- **Best Practices: ACME + OPAQUE** for issue short term certificates to people;

Backgrounds – Identify risks and motivations

- Account Management phase
- Challenge phase (Identifier Validation)
- Certificate Issuance phase



can replace the public key in the CSR

All messages are protected by the public key signature of the account. It' s end-to-end security.

PK replacement attack: Phase 2 just challenge the applying permissions of certificate, but not security binding the public key in CSR. It' s not end-to-end security.

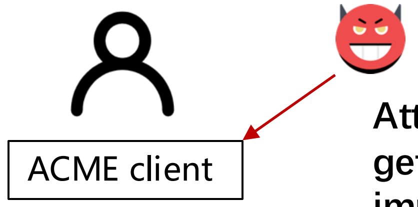
Motivation: the pk-01 can be used to verify that the client wants a given key to appear in its certificate and/or that it controls the corresponding private key.

Two categories and attack scenarios

resource category

proof of ownership of resources

type	identifiers	rfc/draft	status
dns-01	dns	rfc8555	standard
http-01	dns/ip	rfc8555/8737/8738	standard
tls-alpn-01	dns/ip	rfc8555/8737/8738	standard
dns-account-01	dns	draft-ietf-acme-dns-account-label-00	draft active
onion-csr-01	dns	draft-misell-acme-onion-07	draft active



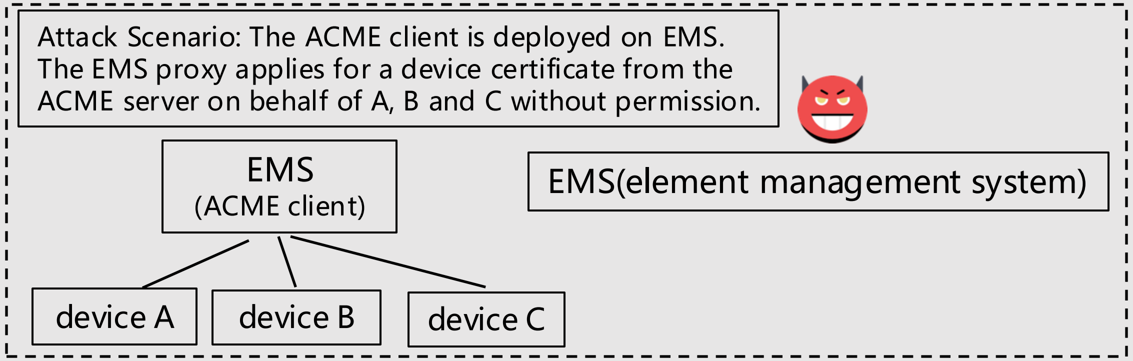
Attack Scenario:
get ACME account credentials
impersonates the victim's server

Our work mainly focus

user/device category

proof of ownership of digital identity

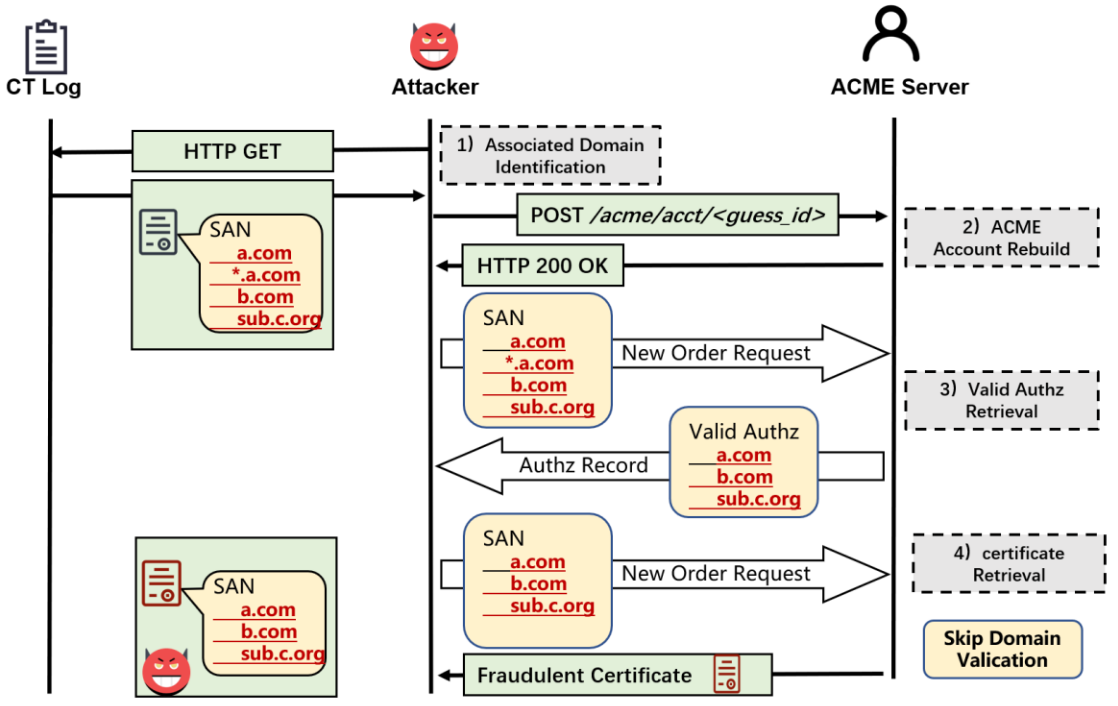
type	identifiers	rfc/draft	status
sso-01	email	draft-biggs-acme-sso-01	draft expired
device-attest-01	permanent-identifier	draft-acme-device-attest-03	draft active
email-reply-00	email	rfc8823	standard
otp-01/hotp-01/totp-01	-	draft-ietf-acme-client-09	draft active
cert-01	-	draft-ietf-acme-client-09	draft active
ppkp-01	-	draft-ietf-acme-client-09	draft active
tkauth-01	TNAuthList	rfc9447/9448	standard



Comparison of two categories of attack models

Existing work: ACME++ Attack path

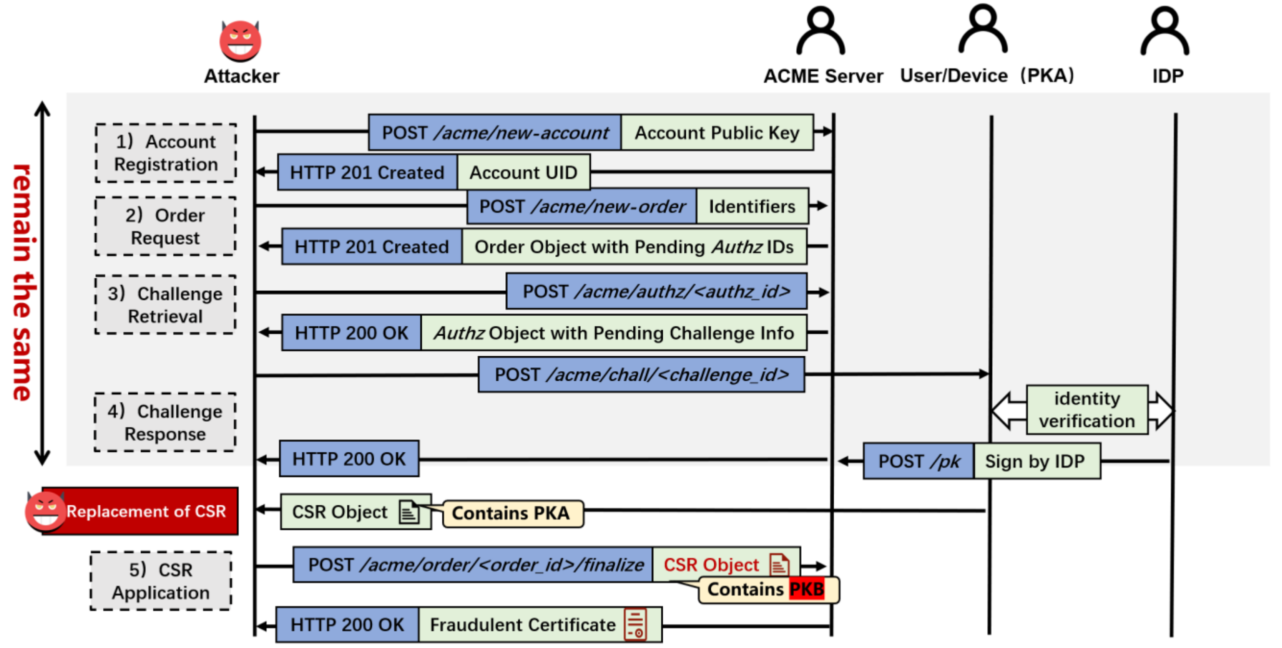
resource category



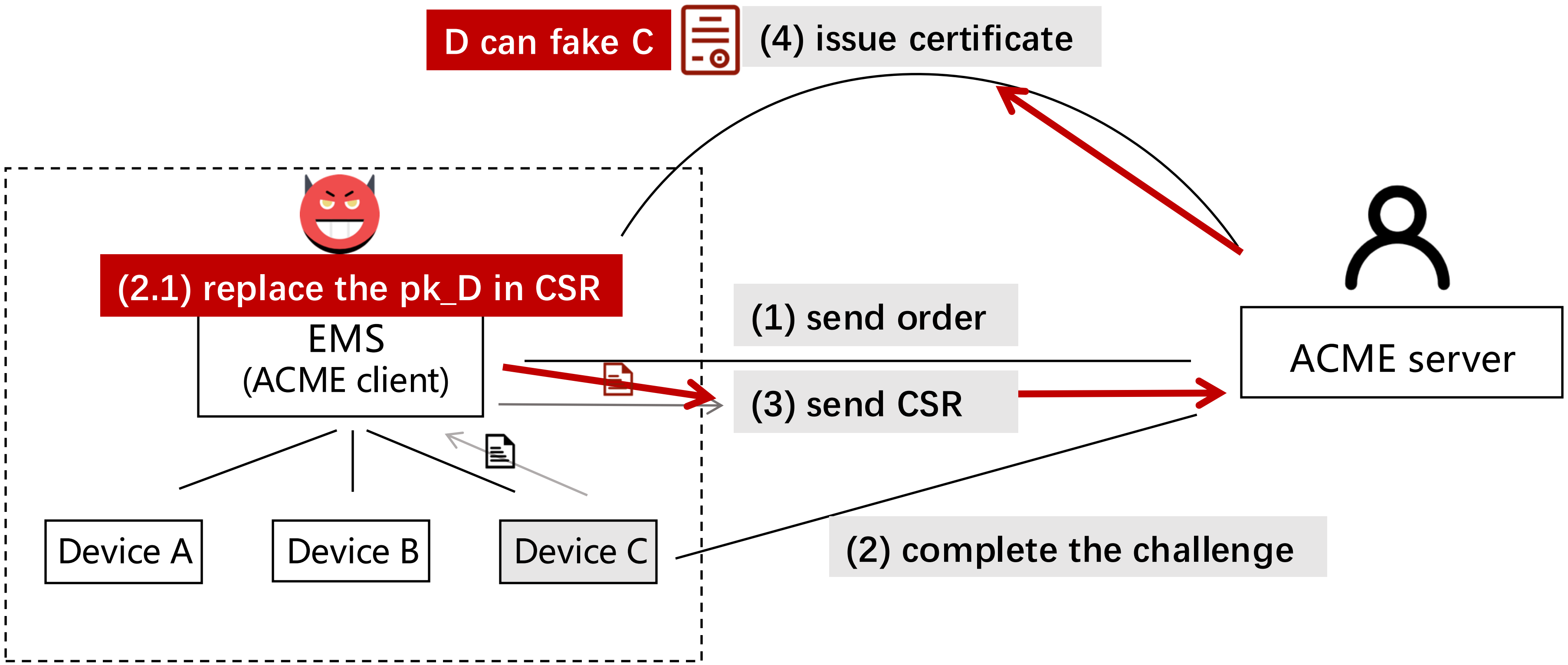
Ref: ACME++: Secure ACME Client Verification for Web-PKI

Our work: Problems with similar attacks

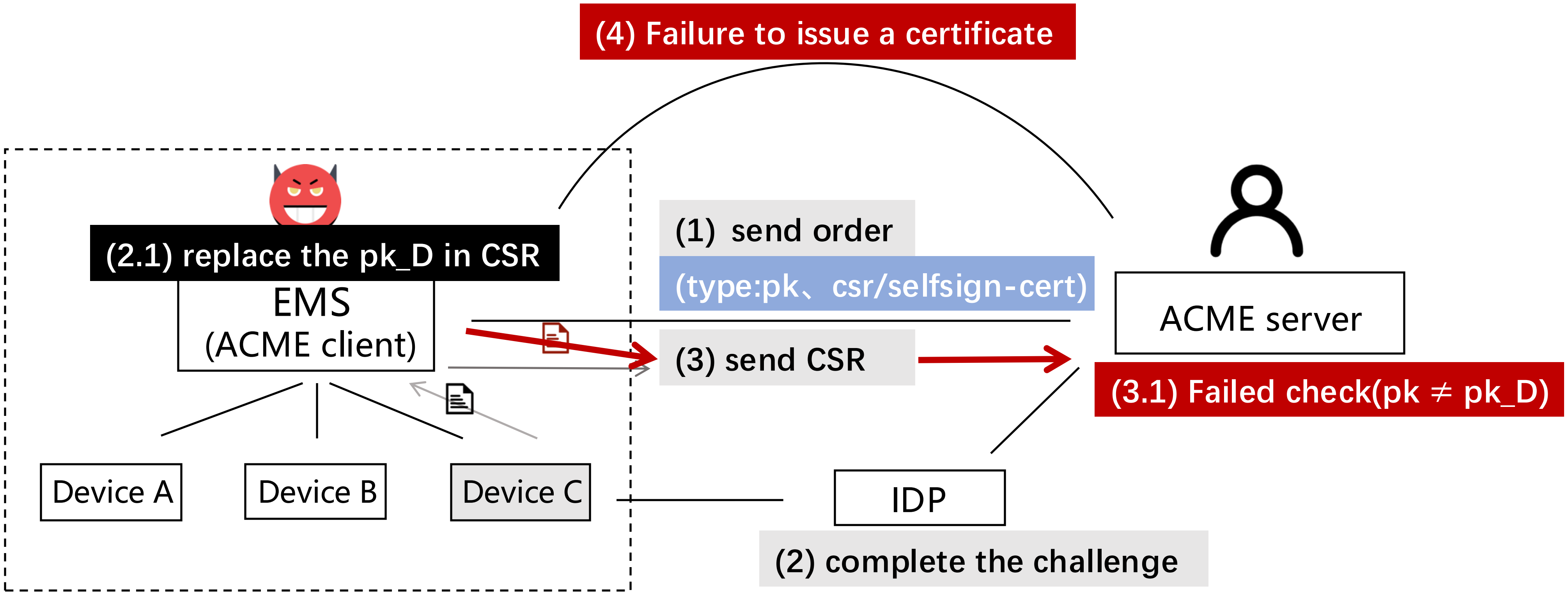
user/device category



Security Model -- ACME client is deployed on EMS （PK replacement attack）



pk-01--Defend against public key replacement attack



Solution -- new ACME Extension for PK Challenges : ACME pk Identifier Type

```
"identifier": { "type": "pk", "value":"MIGfMA0GC***GbQIDAQAB"}
"identifier": { "type": "selfsign-cert", "value":"MIIHSDCC***AU1GH3xQ="}
"identifier": { "type": "csr", "value":"MIICljCCA***RL64+taHbP"}
```

"pk" :

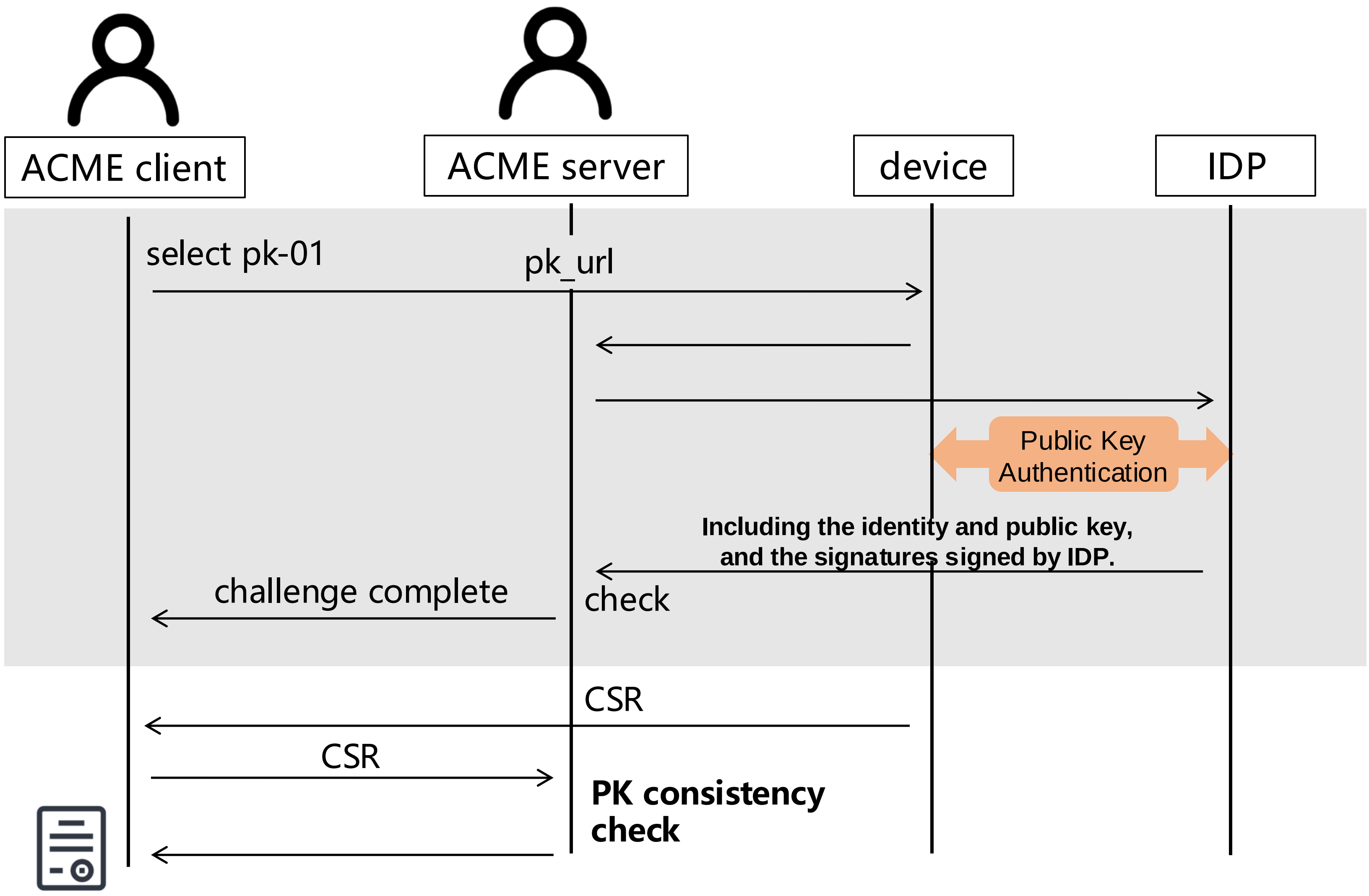
Used to request a certificate for a specific public key.
Example: requesting a certificate for a **device** that is tied to a user's identity.

"csr" / "selfsign-cert" :

Used to request certificates for applicants who need to be identified.
I.e., it requires **binding of specific identity information**.

Solution -- ACME pk-01 Challenge Standard Process

- Account Management
- **Challenge phase (Identifier Validation)**
- **Certificate Issuance**

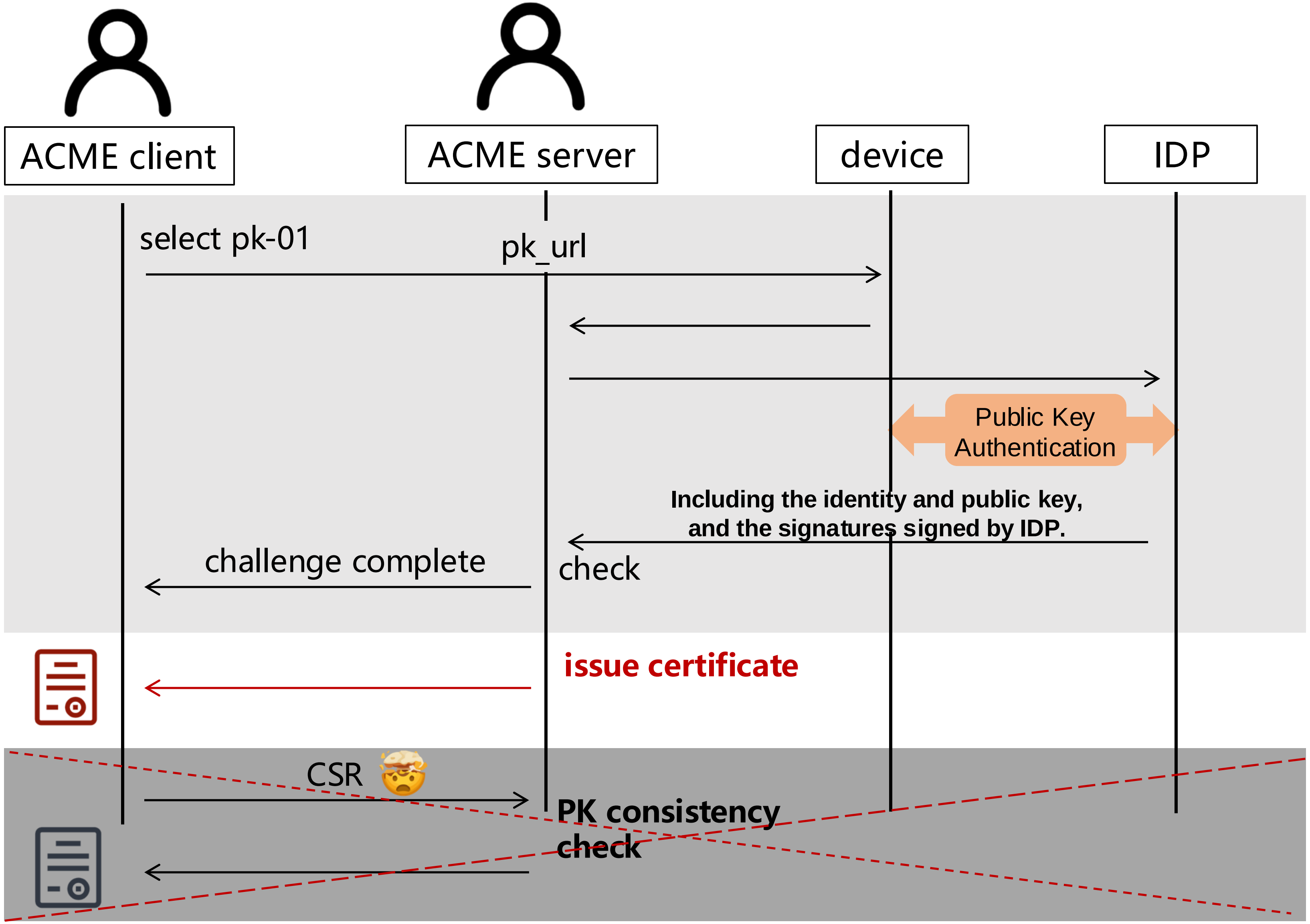


Solution -- ACME pk-01 Challenge could Remove CSR (Optional)

```
{
  “type”: “pk-01”,
  “url”: https://example.org/acme/chall/abc123_defg456,
  “status”: “pending”,
  “pk_url”: “https://example.org/acme/start-pk”,
  “pk_provider”: “https://pk-identity-provider.org/”,
  “standardization”: “standard”, “simplified”
}
```

Optional

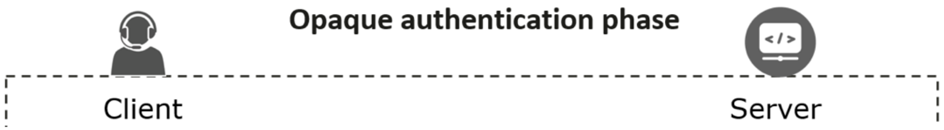
- Account Management
- **Challenge phase (Identifier Validation)**
- **Certificate Issuance**



Various Public Key Authentication Protocols

- challenge public key signature and verify signature (for example, **WebAuthn**)

- Opaque/AKE (draft-irtf-cfrg-opaque-18)

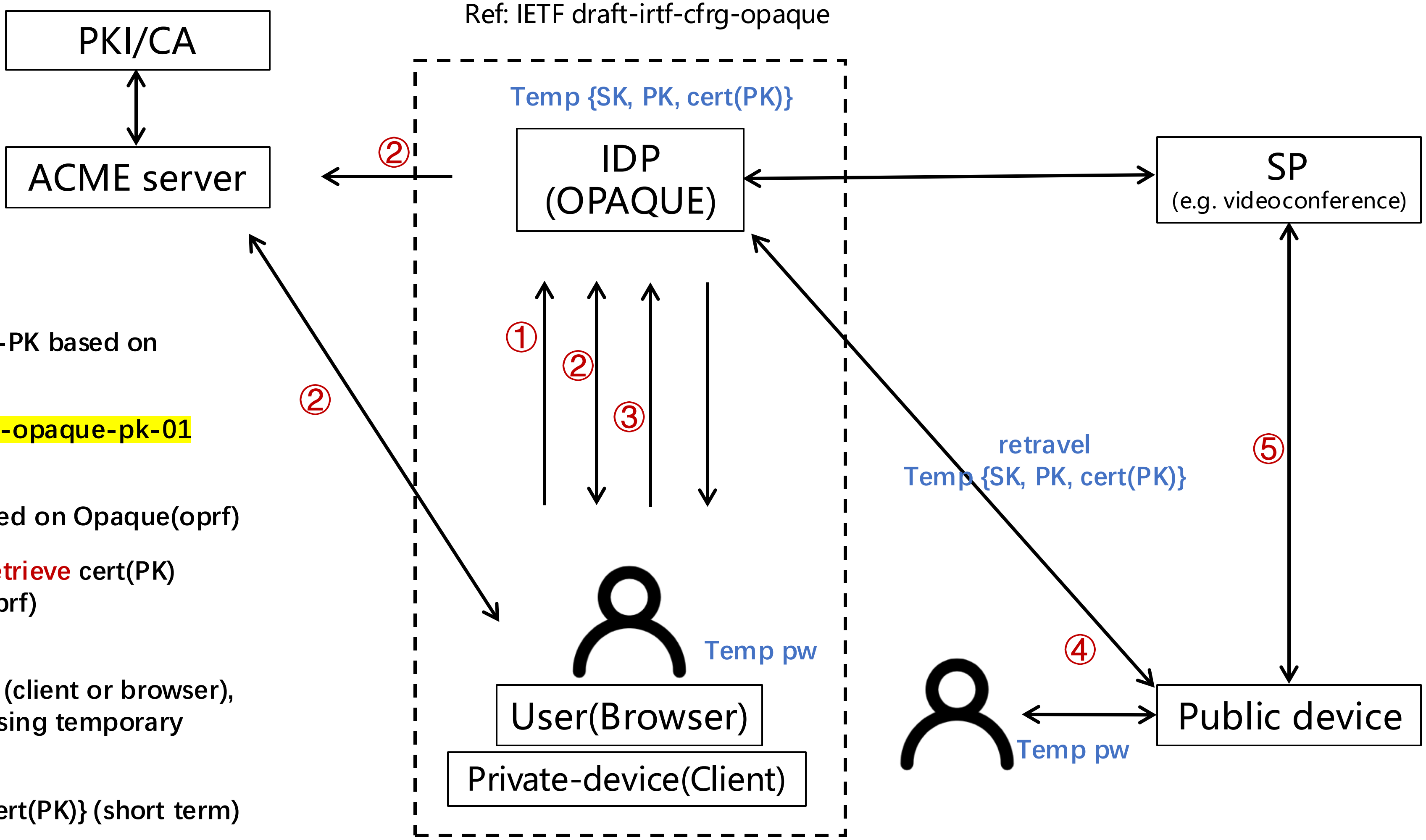


- non-interactive zero-knowledge (NIZK) discrete logarithm equality (DLEQ) proof

……

ACME + OPAQUE -- Apply short term Certificate (One-time Key) for people to be temporary used in the public device

Ref: IETF draft-irtf-cfrg-opaque



1. User account register temporary identity-PK based on Opaque(oprf)

2. Issue temporary cert(PK) based on ACME-opaque-pk-01
e.g. for a two-hour e2ee videoconference

3. User e2ee backup temporary cert(PK) based on Opaque(oprf)

4. When conference starts, User login and retrieve cert(PK) stored in public-device based on Opaque(oprf)

5. User access to SP.
e.g. e2ee videoconference via public-device (client or browser), complete two-way authentication with SP using temporary cert(PK).

6. User quit and delete temporary {SK, PK, cert(PK)} (short term)

Future Work – comments and improvement

Thank Aaron point out:

we should discuss the merits of various challenge types that can be used to verify that the client wants a given key to appear in its certificate and/or that it controls the corresponding private key. I look forward to discussing this further at IETF 122 and into the future.

There is some ideas:

1. **tls-alpn-01** (how to use the existing tls-alpn-01 challenge, with the additional requirement that the presented certificate contain the requested public key and that the TLS handshake complete successfully)
2. **pk-tls-01** (a new method similar to the above, but without the acme-tls/1 ALPN protocol and without the acmeIdentifier extension, allowing the applicant to use their currently-valid TLS certificate if they plan to keep using the same key)
3. **pk-dns-01** (similar to the existing dns-01 challenge, with the change that the server expects to find a hash of the pubkey in the TXT record rather than a key authorization)
4. **pk-dane-01** (similar to the above, but searching for a TLSA 3 1 X or TLSA 1 1 X record rather than a TXT record)
5. **pk-jwk-01** (as mentioned by Richard Barnes upthread, use the private key to sign a JWS over the challenge token, and POST it to the challenge URL like the new onion-csr-01 challenge)