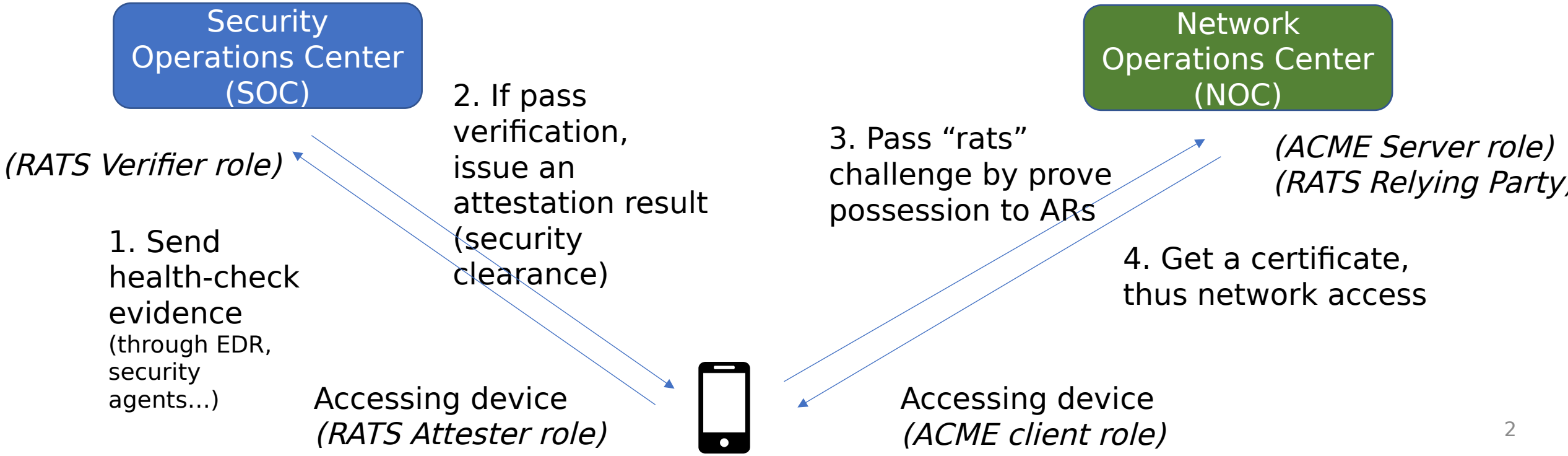
draft-liu-acme-rats

Chunchi Peter Liu, Mike Ounsworth, Michael Richardson

IETF 122 Bangkok

Why – client side certificates for mutual TLS
use case: Grant internal **enterprise network access** (certificates) to devices that pass security checks

- Enterprise usually have 2 teams: Security ops team and Network ops team, where to draw the line is always hard. This work helps them collaborate.



Other Motivating Usecases

- An ACME server (cloud domain host service provider) might put a policy on their ACME client (domain owner tenant)
- A CA (ACME server) may want to know certain attributes of the private key or of the device or application that will use it:
  - Private key resides in FIPS level 3 hardware and has `non-exportable=true`.
  - The policies that apply to certain (cloud) Key Management Service (KMS) instances.
  - TLS / OS / Docker stacks have been recently patched (ie <= 3 months old).
  - Etc.

Changes from last IETF

1. Changed “rats” challenge type to “device-attest” challenge type
   - Referencing `draft-bweeks-acme-device-attest-01`

2. Deleted `http` challenge type

3. Changed how the response carry evidence or attestation result
   - `keyAuthorization = token || '.' || ~~base64url(attestationResult)~~ cmw`

4. Added ACME Attest Claims Hint Registry
   - To facilitate the Server requesting attestation of specific types claims or properties
   - Add IANA request for a new registry
5. Mike and Michael provided contents and reviews, so added them as co-authors.
   - Those who wish to collaborate please do the same? :)

But we think maybe we want to re-think the design

Is Challenge the right place for remote attestation?

- draft-acme-device-attest-01 proves ownership of a permanent identifier, using device-attest-01 challenge.
- More **general** remote attestation may need to go in a different place in the ACME flow, not in the Challenge because:
  - Challenges are for proving ownership of identifiers, not system attributes.
  - Technical problem: The client is only required to respond to ONE challenge.
  - A separate system-level remote attestation could complement the identity challenge.

- So Challenge is not the right place.

Design Space – Proposed direction

POST /acme/order/TOlocE8rfgo/finalize HTTP/1.1

```
{
 "protected": base64url({
  "alg": "ES256",
   … snip …
 }),
 "payload": base64url({
  "csr": "MIIBPTCBxAIBADBFMQ…FS6aKdZeGsysoCo4H9P",
  “attestation”: “…”,
 }),
 "signature": "uOrUfIIk5RyQ…nw62Ay1cl6AB"
}
```

Remote Attestation can go inside the CSR using draft-ietf-lamps-csr-attestation

OR in a new “payload.attestation” param so that attestation(s) can be carried next to the CSR.

Design Space – Proposed Direction

The ACME server needs a way to:

1. Request that the ACME client **provides evidence/attestation results** of the requesting device.
   - For example in the POST /finalize.
2. Request **specific** remote attestation attributes / claims.
   - For example “proof-of-FIPS-level3”, “proof-of-OS-up-to-date” or “proof-of-application-stack”.
3. Provide an attestation **freshness nonce** to the ACME client.

- Idea (undeveloped): add this to the 201 CREATED response to POST /new-order.
  - Maybe in “authorizations” ??

Next steps

- We would like to start an ACME WG affiliated design team meeting once a month. (Maybe? Tuesdays or Thursdays early-ish morning
  N. America time. TBD)
- If you want in, email:
  - Liuchunchi(Peter) <liuchunchi@huawei.com>
  - Mike Ounsworth mike.ounsworth@entrust.com

- Henk has reviewed this slides.