TO COVER TODAY

- Review of changes in -03
- A few open questions
- What do we need to get done before WGLC?
CHANGES IN -03
CHANGES IN -03

WE DID SOME STUFF!

- **27** pull requests
- Highlights...
# SECURITY IMPROVEMENTS

- #154 - Clarify the impact of SSRF
- #150 - Make key-change depend on both new and old keys
- #147 - Change from 'resource' to 'url'
Where once there was “resource”, there is now “url”

As we discussed, comparing URIs is non-trivial

Current text mostly punts. Suggestions for how to do better?

On receiving [a JWS] object in an HTTP request, the server **MUST compare** the “url” parameter to the request URI. If the two **do not match**, then the server **MUST** reject the request as unauthorized.

Except for the directory resource, all ACME resources are addressed with URLs provided to the client by the server. In such cases, the client **MUST** set the "url" field to the exact string provided by the server (rather than performing any re-encoding on the URL).
APPLICATIONS: GOALS

- Support a variety of CA issuance flows, e.g.:
  - CA specifying requirements only after CSR is submitted
  - Requiring re-authorization per issuance
- Enable the CA to have other requirements than domain validation
  - To give PHB his cash box
- These violate a number of current ACME assumptions
APPLICATIONS: APPROACH

- Instead of sending in the CSR as the last step, make it the first step

- CA can then respond with requirements, including:
  - Authorization: You need to prove you own $DOMAIN
  - Out-of-band: You need to go visit this website
APPLICATIONS: BEFORE

CHANGES IN -03

- POST
- POST
- GET
- POST

REGISTRATION

AUTHORIZATION

CHALLENGE

CERTIFICATE
CHANGES IN -03

APPLICATIONS: AFTER

POST POST POST GET GET
REGISTRATION
APPLICATION
AUTHORIZATION
CHALLENGE
CERTIFICATE
Client
APPLICATIONS

- Work for client is about the same (save one POST)
  - If the server allows reusable authorizations, future applications can use them
  - Endless loops avoided by requiring server to state all requirements up front
- Server has to do a little more state tracking
- Current spec is a first draft, bunch of questions in the next section
OPEN QUESTIONS
#128 – INDICATE ACME VERSION IN DIRECTORY (OR ELSEWHERE?)

- Current: No in-band signaling of the version the server thinks it’s speaking
  - Could be indicated, e.g., by using different URLs per version
- Might need some further thought
  - E.g., might be valid to use a v1 registration to do stuff in v2

Editor suggestion: Discuss
OPEN QUESTIONS

#157 – FETCH REGISTRATION RESOURCE

- Current: If a client attempts to re-register with the same key, return 409 Conflict
  - People are using this to retrieve the registration for a key
- Proposal: Return 200 instead of 409

Editor suggestion: Fix as proposed
OPEN QUESTIONS

#152 – REDUCE THE NUMBER OF OOB THINGS

- Current: Two out-of-band things:
  - “Requirement”: You must do this OOB thing before I will issue the certificate you requested
  - “Challenge”: You must do this OOB thing before I will consider you authorized for this identifier

Editor suggestion: Drop or Remove OOB challenge
#156 - UN-PARALLELIZE THE SIGNATURES IN THE KEY-CHANGE REQUEST

- Context: Want to change account key from oldKey to newKey
- Current: Message is \((M, \text{sig}_\text{old}(M), \text{sig}_\text{new}(M))\), where \(M=[\text{oldKey}, \text{newKey}]\)
  - Requires a multi-signed JWS, which is not supported by some libraries
  - Breaks the pattern of every POST being signed by a single key
- Proposed: \(\text{sig}_\text{new}(\text{sig}_\text{old}(M))\)

Editor suggestion: Fix as proposed*
OPEN QUESTIONS

#156 - ECONOMIZE ON NONCES

- Current: Server MUST provide a “Replay-Nonce” header in each successful response
  - Assertion: This means that the server has to track a bunch of nonces
- Proposal: Restrict the requirement to POST responses, plus some sort of priming source (e.g., directory)

Editor suggestion: Drop
A FEW (UN-FILED) THINGS NOTED IN THE SPEC

- Details of applications
  - Should they require a request to "activate" / "buy"? (Optionally?)
  - Should they be modifiable?
  - Should they allow multiple CSRs / Certificates?
- Recovery: Declare really and finally gone? Punt to a later extension?
- Should scoping of authorizations be more flexible?
WGLC PREP
You are here
WGLC BLOCKERS

- Anything left from the previous section that we didn’t solve
- What else?
- Would like to have at least a couple of implementations of the latest spec
WGLC PREP

ROUGH TIMELINE

- Mid-August - “Implementation draft”
  - Implements agreements from this meeting
  - Baseline for implementors to get working
- September - See how implementation proceeds
  - Maybe a virtual interim if there are issues that come up?
- October - WGLC
ONE MORE THING
ROCKET-SKATES

- Since the I-D deadline, been working on an implementation of -03
- https://github.com/bifurcation/rocket-skates
- Currently very much a work in progress
  - Implements client and server side of the ACME transport layer (JWS, etc.)
  - Server side of the ACME logic is mostly complete
  - No client-side ACME logic, and no challenges yet
  - Comments / Issues / PRs / interop reports very welcome