draft-ietf-acme-acme

IETF 98

Nothing big (?)

Closed since last IETF meeting

We did some stuff!

57 pull requests since IETF 97, with **14** different authors!

Editorial

#221	#225	#226	#227	#229
#230	#231	#232	#233	#235
#243	#246	#247	#248	#252
#253	#254	#257	#260	#261
#265	#268	#270	#273	#279

From IETF 97

Require error message for bad nonce. **(#214)**
Comprehensibility improvements **(#213)**
Add MAC-based external account binding **(#212)**
Add registries for fields in account and order objects **(#211)**
Terminology update **(#210)**
Remove the 'requirements' abstraction **(#208)**
Change agreementRequired to userActionRequired **(#207)**
Add some cautionary words about TOS agreement **(#205)**

Naming / Abbreviation Cleanup

Normalizes 'revoke-cert', 'account' **(#282)**

Refer to problem document error 'codes' as 'types' instead **(#262)**

Editorial: some `reg`s to `account` **(#259)**

Data Model Cleanup

Remove key from Fields in Account Objects table **(#287)**
Remove "key" from account object. **(#266)**
Remove key equivalence section and simplify key roll-over procedure **(#263)**
Remove new-authz fields "existing" and "accept". **(#275)**
Remove extraneous link relations and change 'directory' to 'index' **(#250)**
Update errors and add an error field to orders. **(#264)**
Clarifies allowed fields in stub account objects **(#241)**
Updated order object registry authorizations field **(#224)**
Cleanup authorizations field of order object **(#223)**
Provide more specification for how servers handle unknown fields **(#219)**

Operational Cleanup

Clarify language relating to proactive certificate issuance **(#286)**
Register application/pem-certificate-chain. **(#276)**
Replace Content-Location with Location. **(#274)**
Responses to @martinthomson comments **(#271)**
Clarify that servers must reject disallowed revocation reasonCodes **(#251)**
Clarify DNS considerations section, always recommend DNS over TCP **(#249)**
Remove SHOULD for HPKP. **(#244)**
Specify server MAY follow HTTP redirects. **(#238)**
Have new-nonce return 204 **(#222)**
Put the immutability burden on the server **(#220)**
Add an error code for unsupported signature algorithm **(#218)**

Open Issues

Open PRs

Editorial discretion:

#240 Discuss invalidation of authz after compromise

#280 Accounts: Structure and fixes underspecification

#281 Replaces HTTP(S) URIs with URLs

#288 Fixes and clarifications

#284 Rewords token entropy justification

#285 Additional editorial changes

#289 Make Replay-Nonce uniqueness a MUST

To discuss:

#272 Use correct challenge names in the IANA list

Challenge Names

Current	RFC (current)	RFC (proposed)
`http-01`	`http`	`http-01`
`tls-sni-02`	`tls-sni`	`tls-sni-02`
`dns-01`	`dns`	`dns-01`
	Cleaner for future Future is bigger than the past	Avoid unnecessary breakage Might have to rev later anyway

Open Issues

Editorial discretion:

~~Challenge field "URI" should be "URL" (#278)~~

~~Consolidate on "field name" or "key" for JSON (#269)~~

Inconsistent challenges in examples (#277)

Rewrite Identifier types section (#258)

Discuss:

Enable cleanup of "leaked" pending authorizations (#242)

Recommend polling order rather than authz (#237)

ACME resources relations (#236)

Cleanup of “Leaked” Authorizations

- Let’s Encrypt had an issue with buggy clients
  - Create a bunch of pending authorizations that saturate a rate limit
  - Don’t keep around authz URLs that you would need to deactivate

- Probably not worth addressing this case very specifically
- Nonetheless, re-add “authorizations” field to accounts?

Recommend polling Order vs. Authz

Action	Request	Response
Get a nonce	HEAD new-nonce	204
Create account	POST new-account	201 -> account
Submit an order	POST new-order	201 -> order
Fetch challenges	GET authz	200
Respond to challenge	POST challenge	200
Poll for status	**GET ~~authz~~order**	**200**
Check for new cert	GET cert	200

ACME Resource Relations

- `authz --"up"--> order`
- But authorizations are reusable
- Which order should “up” point to?
  - The order that caused the authz to be created
  - All orders that used this authz
  - No orders at all (just remove the relation and have the client track)

Issues from Mailing List

Discuss:

[Acme] Require servers to accept at least one automated revocation method

[Acme] Allowing clients to understand the account creation functionality ...

[Acme] Retrying failed authorizations

[Acme] draft-ietf-acme-acme-06: Don't call it PEM Certificate Chain

Define Revocation More Clearly

> The server SHOULD consider at least the following accounts authorized for a given certificate:
>
> - the account that issued the certificate.
> - an account that holds authorizations for all of the identifiers in the certificate.
>
> The server SHOULD also consider a revocation request valid if it is signed with the private key corresponding to the public key in the certificate.

- Require that the server support at least one?
- … all of them?

Advertize Server Contact Support

> **The server SHOULD validate that the contact URLs in the "contact" field are valid and supported by the server. If the client provides the server with an invalid or unsupported contact URL, then the server MUST return an error of type "invalidContact", with a description describing the error and what types of contact URL the server considers acceptable.**

- Confusion here between "invalid" and "unsupported"
  - `{"contact": ["mailto:anonymous@invalid"]}`
  - `{"contact": ["moz://a"]}`
- Client can't predict what schemes a server supports
- Advertise support? Require some set? (e.g., "mailto" and "tel")

Retrying failed authorizations

- Submit an order with 10 SANs
- … get 10 authz objects, try to fulfill 10 challenges
- … 9 work, 1 fails because the DNS didn’t propagate fast enough
- Client can re-start the order, the authzs should carry over (at CA discretion)
- Can we be more efficient?
  - Allow challenges to be retried
  - Allow challenges to be regenerated
  - Allow authorizations to be regenerated
  - (rlb) Recommenda that CAs re-try validation queries

Don't call it PEM Certificate Chain

- Don’t call it “PEM” (?)
- Don’t require that the certificates be in order (?)
- Don’t use textual encoding at all (?)
  - `application/pkcs7-mime; smime-type=certs-only`
  - CMS `SignedData`

A COUPLE OF QUITE BIG THINGS

Where do we go from here?

To the IESG?

Since IETF 97

- Had a WGLC
- Fixed a bunch of WGLC comments
- Some implementation has started

Re-WGLC? Ready to send to the IESG?

Hold for implementation / interop? How long?