Captive Portal API

draft-ietf-capport-api-00

Tommy Pauly & Darshak Thakore

CAPPORT

IETF 101, March 2018, London

Basic API WG Draft

Define URI as accessible using HTTP over TLS

JSON dictionary is fetched from API server with a GET request

JSON keys are defined as:

- "permitted" (required, boolean)
- "hmac-key" (required, string)
- "user-portal-url" (required, string)
- "expire-date" (optional, datetime string)
- "bytes-remaining" (optional, integer)

Open Issues

Privacy

To ensure that per-client information (like the HMAC key) is not stored in shared caches, the document should specify that responses should include `Cache-Control: private`.

Open Issues

HMAC Key

The use of the HMAC key needs to be solidified in the architecture.

- Current version references a requirement for a “security token for validating ICMP messages”

- Which document should define the mechanism for this validation of ICMP (or other) messages?

Open Issues

Server Authentication

API document needs to describe the server authentication model

Certification Revocation Lists and OCSP validation will generally fail during captivity

OSCP Stapling would avoid the issues of reachability while captive

Should this be made mandatory? What does the UE do if OSCP stapling is not present?

Open Issues

Media Type

Rather than relying using only `application/json`, define a media type to allow better versioning, etc.

Options:

`application/captive+json`

`application/capport+json`

Open Issues

Follow-up to .well-known

Earlier version had suggested a `.well-known` URI, which was decided to be inappropriate on the mailing list

Suggestion was made to use Link Relations and possibly also specify that the UE first issues just a HEAD

Ultimately, we should have some text being very clear about what the URI means and how to distinguish it from a user-interactive landing page