Android captive portal API

IETF 103 - capport

November 2018

Warning: early design ideas

These slides only contain early design ideas that may or may not be implemented, or may be implemented differently.
Their sole purpose is to get early working group feedback on usage of the discussed API.

Captive portal UX improvements: use cases

Usage information

Allow users to view how much time or data is left, possibly notify them when it runs out.

Possible solution: surface the API-provided time and data left in the system UI.

Session management

Allow users to view details of their plan, and modify it (extend, upgrade/downgrade).

Possible solution: surface the API-provided URL as a WiFi management shortcut to open the portal operator's page.

AP-published information

Allow users to view information published by the venue: in-flight entertainment, schedules & timetables, etc.

Possible solution: surface the (same ?) API-provided URL as a shortcut to AP information

General flow

- Discover the API endpoint (DHCP / IPv6 RA)
- Probe endpoint as a hint to detect portal open/close, and to get session information to show in UI
    - Only better UI controls - don’t take decisions to update default routes only based on it
- Periodically check for updates, until we have a “API info changed” notification mechanism

Session info & venue info

- Some APs may want to publish venue-specific info: in-flight entertainment, event schedules, timetables
- Some APs may not: universities, large subscription-based WiFi AP networks. Showing a venue info shortcut is not useful there.

Question: should we have separate API fields for user-portal-url (login info) and optional venue info ?

THANK YOU