CDNI Triggers extensions

draft-finkelman-cdni-triggers-sva-extensions-01

Sanjay Mishra, Ori Finkelman
IETF-103, Bangkok
November 2018
Refresher

• Versioning
  • Define versioning scheme for triggers
  • Define version 2 of the trigger commands and objects
  • Advertise version support via FCI

• Select content for trigger by:
  • Regular expression
  • Video playlist (manifest)

• Add generic extension objects
  • List of generic extension objects
  • Initial set of extension objects: LocationPolicy, TimePolicy
  • Support for FCI advertisement for the extension objects
Follow-up on IETF102

- Replace “abr-protocol” type with “media-protocol” – done
  - Consider removing MSS from the draft as it is hardly in use anymore

- Implication of regex complexity attacks – added to security considerations section

- Add local time option to TimePolicy, instructing the dCDN to execute the trigger at their own local time

- Propagate errors from all dCDNs along the trigger delegation path to enable backtracing of errors and identifying the dCDN that generated the error
Local time in TimePolicy

• Rational
  • Execution of triggers may be more cost effective at certain hours, for example off-peak hours
  • Off-peak is defined locally – for example 2AM-5AM at the local time

• Currently this can be achieved by combining TimePolicy with LocationPolicy
  • Drawback – requires multiple triggers, one for each region

• Alternatively add a “local-time” flag to indicate “execute at these hours on your local time”
Local time representation

• How to present the time window when using local-time

• Cannot use the exiting TimeWindow object
  [https://tools.ietf.org/html/rfc8006#section-4.2.3.2](https://tools.ietf.org/html/rfc8006#section-4.2.3.2) as it is defined in UTC and using UNIX epoch

• Can we add a new time slot object?

• Time representation using RFC3339 (Internet time)
  • There is no notation for “your local time” only for specific offset
  • Ideas? Other options?
Error propagation – Error Description V2

• Errors may happen in any dCDN down the trigger delegation path
• Mainly required due to the addition of generic extensions, a trigger may include a generic extension object that a certain dCDN does not support
• `errors.v2` adds the property “cdn” to indicate the failing dCDN
• When delegating triggers, the delegator dCDN MUST propagate errors they have received from dCDNs down the delegation path, and add these errors to the error.v2 array
• New error code “eextension” indicates an error in a trigger extension

```json
{
    "content.urls": [
        "https://newsite.example.com/index.html"
    ],
    "description": "unrecognized extension type CIT.LocationPolicy",
    "error": "eextension",
    "cdn": "AS64496:1"
}
```
Trigger Status V2

• Recap from previous draft
  • Version 2 of CI/T interface
    • `ci-trigger-command.v2` – use “trigger.v2” instead of “trigger”
    • `ci-trigger-status.v2` – use “trigger.v2” instead of “trigger”
    • `ci-trigger-collection` – no changes

• Trigger Status v2 MUST now also use v2 of error description

```json
{
    "errors.v2": [ { <properties of 1st error.v2 object> }, ...
        { <properties of Nth error.v2 object> }
    ],
    "ctime": 1462351690,
    "etime": 1462351698,
    "mtime": 1462351690,
    "status": "pending",
    "trigger.v2": { <properties of a trigger.v2 object> }
}
```
• Next steps – call for adoption
  • Requires re-charter of CDNI?

• Q & A

• Thank You !