Website Packaging (WPACK)

Dan York

york@isoc.org

@danyork

July 2018 - IETF 102

People would like to use websites, including those with "application" functionality...

- Offline
- In situations with low or minimal bandwidth
- In situations where it may not be possible to directly connect to the original server (ex. censorship)

Methods exist to package web applications for offline use (ex. W3C ServiceWorkers)...

... but how do you *distribute* and verify the authenticity without a network connection?

Use Cases (from draft-yasskin-webpackage-use-cases)

- Offline installation
- Offline browsing
- Save and share a web page
- Packaged Web Publications
- Avoiding Censorship
- Third-party security review
- Building packages from multiple libraries
- Privacy-preserving prefetch
- Cross-CDN Serving
- Installation from a self-extracting executable
- Packages in version control
- Subresource bundling

Web Packaging documents

- Use Cases and Requirements
  - https://tools.ietf.org/html/draft-yasskin-webpackage-use-cases
- Signed HTTP Exchanges
  - https://tools.ietf.org/html/draft-yasskin-http-origin-signed-responses
- Bundled HTTP Exchanges
  - https://tools.ietf.org/html/draft-yasskin-wpack-bundled-exchanges-00
- Jeffrey Yasskin of Google main point person (but is not here at IETF 102)
- Work underway on both specification and a Chromium implementation

Join in

- Mailing list: https://www.ietf.org/mailman/listinfo/Wpack
- Github: https://github.com/WICG/webpackage
- Possible Bar BOF / side meeting at IETF 102 (still being organized)

Thank You

Dan York

york@isoc.org

@danyork

FYI, these slides were created in Markdown and rendered using MARP (https://yhatt.github.io/marp/)

Slides can be found at: https://github.com/danyork/wpack-intro-ietf102