Public Key Pinning Extension for HTTP

Ryan Sleevi

Changes since draft-03

- ABNF updated to match HSTS feedback.
    - ABNF describes generic syntax, text describes specific fields
    - Addresses "max-age" as being required (Issue 52)
- "strict"
    - Issue 53
- Public-Key-Pins-Report-Only Header
    - Issue 54
- "includeSubDomains"
    - Issue 56
- Normative text describing UA processing model and noting pins (Issue 55)

Issue 52 - max-age

- Previous drafts did not make it clear that this field was explicitly required in all occurrences of the header
- Solution: Adopt the approach from HSTS (RFC 6797)

Issue 52 - Possible problems

- draft-04 eliminates the "Pin Validity Times" solution proposed in previous drafts, which was mirrored from similar proposal in TACK
  - If an attacker can obtain a fraudulent-but-valid certificate, can set a long-lived max-age
- Should there be an upper bound for max-age?
- What are the security/implementations considerations from UAs that have the ability to "un-pin" (eg: via updated Preloads) in order to undo these actions.

Issue 52 Discussion

Issue 53 - Private Trust Anchors

- As implemented today, pins are not noted or observed when certificates chain to trust anchors outside of the user agents "default trust set" (vendor root list)
    - Reality is that users intentionally violate TLS's end-to-end guarantees for a variety of reasons, and user agents must deal with this. This is not RFC 1984.
- Proposed Solution: "strict" mode to indicate a site wishes to be pinned, even under such anchors

Issue 53 - Possible problems

- Greater opportunities for MITM to inject "bad" pins
  - e.g.: A site that *expects* to be issued by a public trust anchor may be pinned to a private trust anchor
- Greater opportunities for public sites to misconfigure
  - e.g.: A site with public trust anchor in strict mode may end up unintentionally denying access to users who are intentionally breaking TLS's E2E guarantees (eg: anti-virus software)
- Will it be implemented/respected? Whose policy takes precedence - site or user/UA?

Issue 53 Discussion

Issue 54 - Report-only mode

- Desire to see a Content-Security-Policy like mechanism for reporting pins and pin failures
- Proposed Solution:
  - Public-Key-Pins-Report-Only Header: Specify a fully distinct policy that is evaluated independently of Public-Key-Pins
  - report-uri directive to indicate a URI to report policy violations to (for both Public-Key-Pins and Public-Key-Pins-Report-Only)

Issue 54 - Possible problems

- **NOT** intended as a security measure
  - Attacker can simply block reports. To be a security measure, requires escalating attempts to exfiltrate data that run counter to many goals of UAs.
- Needs more clarification of processing/reporting mode
  - e.g.: A is pinned with a report-uri of B. B is pinned with a report-uri of A. Both pins are invalid. UAs should not go into an endless cycle of reporting failures
  - If A's report-uri is to A, over secure transport, should the pin report succeed (ignore pins during reporting) or fail (respect pins during reporting)

Issue 54 - Possible problems

- Are there new privacy/security considerations introduced by having clients report their pinning policy and observed certificate chain?

Issue 54 Discussion

Issue 55 - Interactions with Preloads

- Previous drafts left it ambiguous which pinning metadata took effect - observed or preloaded
- Proposed Solution: draft-04 indicates that the "latest observed" pinning metadata takes effect

Issue 55 - Possible problems

- UAs may have differing implementations of preloading, and thus may decide "latest observed" differently
    - UA X implements "latest observed" as "latest update for any pins within the set of all known pins" (eg: applies to all pins)
    - UA Y implements "latest observed" as "latest contact from specific site/site operator requesting pinning" (eg: applies per-site)
    - May make more sense to remove discussion of preloading altogether from draft, treating it as simply a source of "observations". Preloading is largely independent of how the header is recorded/observed.

Issue 55 Discussion

Issue 56 - includeSubDomains

- With pinning no longer inheriting from HSTS / RFC 6797, no way to specify pinning policy for subdomains, only a single host
- Proposed Solution: "includeSubDomains" directive, same semantics as with RFC 6797.

Issue 56 - Possible problems

- As described, no good way to indicate independent policies for subdomains.
    - Behaviour is contingent upon which policy is observed first - parent domain or sub-domain.
    - If "includeSubDomains" is used and seen first, any subdomain policy must be a strict (possibly more restrictive) subset of parent domain

Issue 56 Discussion

Other Issues?

- As an implementer, still working through issues such as what happens if a UA supports both HPKP and TACK or other pinning solutions.
    - Prioritization of policies
    - Possibility of conflicting policies