HTTP Strict Transport Security (STS) Policy

Jeff Hodges (=JeffH)
PayPal Information Risk Management
<Jeff.Hodges@PayPal.com>
Agenda

• History
• Overall Use Cases
• Threat Model
  • Threats Addressed
  • Threats Not Addressed
• STS Policy Effects
• STS HTTP Header Design
• STS Policy Scope
• Design Issues
• Status
• Experience
• Futures
History

• *ForceHTTPS* conceived by Jackson and Barth in 2007
  • In response to others' approaches (e.g. Locked-Same-Origin)
  • Presented at WWW 2008 (April)
  • [https://crypto.stanford.edu/forcehttps/](https://crypto.stanford.edu/forcehttps/)

• General notion kicked around sporadically by various folks since publication
  • =JeffH (me) enters picture Spring 2009
  • Coalesced various folks' thoughts wrt ForceHTTPS
  • Initially spec was known as *ForceTLS*
  • Present (draft) STS spec pushed out 18-Sep-2009
Overall Use Cases

• *Web browser user* wishes to interact with various web sites in a secure fashion

• *Web site deployer* wishes to offer their site in an explicitly secure fashion
Threat Model

• We increasingly access web via random networks
  • e.g. wireless hotspots – eavesdropping and/or Man-in-the-middle opportunities

• Web sites can have config issues
  • E.g. not using secure transport where needed and/or consistently

• Browsers have lax security posture by default
  • Facilitate users in “clicking through” security
Threats Addressed

• Passive Network Attackers

• Active Network Attackers

• Web Site Development and Deployment Bugs
Threats Not Addressed

• Phishing

• Malware and Browser Vulnerabilities
STS Policy Effects

• STS server redirects insecure connections to secure ones

• UA terminates—without user recourse—secure connection attempts that generate any secure transport errors

• UA transforms insecure URIs to STS server into secure ones before loading
STS HTTP Header Design

• STS Server declares STS policy by returning STS response header:

"Strict-Transport-Security" ":" "max-age" "=" delta-seconds [ ":" "includeSubDomains" ]

• Examples:

  Strict-Transport-Security: max-age=65536
  Strict-Transport-Security: max-age=10000; includeSubDomains
STS Policy Scope

• STS policy only enforced if received by UA over secure transport

• Scope is:
  – Emitting domain
  – Subdomains (if “includeSubDomains” stated)

• Child domain can't set policy for parent or peers
Design Issues

• IncludeSubDomains (?)

• Mixed Security Context  aka mixed content
Status

• Publicly available draft spec (update coming soon)
  • draft-hodges-strict-transport-sec-05.plain.html

• Spec presently implemented by:
  • Google Chrome
  • NoScript and ForceTLSv2 FireFox extensions
  • Embedded implementation underway in FireFox
  • PayPal.com emits STS policy

• Working towards having STS spec adopted as a “working group deliverable” either in IETF or W3C
Experience

• Various sites experimenting with STS (heard through grapevine...)

• E.g. site emits STS policy with small max-age value (minutes or hour) and sees what breaks
  - e.g. some site components served insecurely from supposedly “secure domain”
  - Means to find site issues
Futures

• Additional directives (?)
  • LockCA
  • EVonly

• STS Site Registry
  • Shipped embedded in UAs a la root certs
  • How to vet inclusion applications?
Thanks!

Questions?

This Preso available at: