Cryptography for Software and Web Developers
Part 1: Web and Crypto

Hanno Böck

2014-05-28
Many webpages use some kind of mix between HTTP and HTTPS

This is (almost) always insecure - don’t do it!
How to people get to webpages? Type in URL, Link from elsewhere, Bookmark

If initial access happens through HTTP and forward to HTTPS only happens later we can do SSL Stripping

Change links from HTTPS to HTTP, Man-in-the-Middle: server - https - attacker - http - client

sslstrip is free and easy to use [url]
Cookies have a flag "secure" - you have to set it, this doesn’t happen automatically

If you don’t, every HTTP connection will reveal the cookie

Even if you don’t speak HTTP at all, attacker can still point victim to http://yoursite:443

(my intermediate thesis [url])
HTTPS website, HTTP JavaScript, CSS or other active content

This is mostly a non-issue today, browsers block this

Chrome and Safari still allow XMLHttpRequest and WebSocket mixed content

Images and other non-active content can be safe in some situations, I wouldn’t risk it
Would you like to proceed with this action?

Yes  No

Would you like to proceed with this action?

No  Yes

Can you spot the difference?
”But I can’t do HTTPS-only, it’ll kill my performance. Because... our service is so big and we have so many users.”
"In January this year (2010), Gmail switched to using HTTPS for everything by default. [...] In order to do this we had to deploy **no additional machines** and **no special hardware**. On our production frontend machines, SSL/TLS accounts for less than 1% of the CPU load, less than 10KB of memory per connection and less than 2% of network overhead.” (Adam Langley, Google developer) [url]
In most cases TLS is not a significant performance hit

Don’t believe things, test them (benchmarks)

Latest Intel/AMD CPUs contain AES instructions, about 2x speedup. Make sure your virtualization doesn’t prevent it

OpenSSL has 64-bit optimizations for ECC, not always enabled by default

SPDY: experimental, not necessarily a reliable implementation available for your software
HSTS sends a signal to the browser: "This domain is HTTPS only and extra secure"

Includes a time for which browsers should cache this information

Enables stricter HTTPS checks and prevents clicking away of warnings

Big improvement, prevents SSL stripping in most cases, use it!

Remaining problem: First access (chrome has some default-to-hsts-lists, DNSSEC advised HSTS could help)
I found and reported an XSS vulnerability on @sears: they replied that they use SSL and are safe... #fail @troyhunt pic.twitter.com/jtpWknLJ0R
Erh, no!

You can have an extra secure XSS or SQL injection, encrypted with AES-GCM, 256 bits, 4096 bit RSA and extra-strong Perfect Forward Secrecy - it’s still a vulnerability

Be aware what crypto can and can’t do

And regarding XSS and SQL injections: Use Content Security Policy to stop all XSS and prepared statements to stop all SQL injections
▶ Don’t mix HTTP and HTTPS, it’s never secure
▶ Set secure flag for cookies
▶ Use HSTS
▶ Don’t trust unfounded claims, demand real data
▶ Crypto won’t safe you from non-crypto issues
- sslstrip download and talk
  http://www.thoughtcrime.org/software/sslstrip/
- Session-Cookies and SSL
- Mixed Content http://blog.ivanristic.com/2014/03/
  https://blog.ivanristic.com/2014/03/
- Gmail, TLS and Performance
  https://www.imperialviolet.org/2010/06/25/
  overclocking-ssl.html
- XSS and SSL https://twitter.com/Cybpoulet/status/460438949257691136/photo/1