Fixing OCSP for Fun and Profit

David Adrian
@davidcadrian
OCSP Is Unreliable

Certificate revocation doesn’t work.

CRLs are too big and slow.

CRLSets are too small.

OCSP is slow / unreliable / fails open / leaks visited domains.

OCSP stapling is fail open.
Solutions

OCSP Must-Staple is too **scary**. Web servers are **bad at stapling**.

Chrome Expect-Staple preload list is only useful for measurement papers.

*What to do? How do we really solve revocation?*
Gamification!
OCSP Suspect-Staple

Client and server guess if web server will correctly staple

Commit to guess in Client/ServerHello

Incentivize servers to staple

Gamification!
Client Hello: client random, H(random + S_c)

Server Hello: server random, H(random + S_s)

Certificate: certificate chain (public key PK)

Certificate Status?: OCSP response

Client Key Exchange: Encrypt_{PK}(premaster secret),

\[ K_{ms} = \text{KDF}(\text{premaster secret, client random, server random}) \]

Client Finished: E_{Kms}(\text{Hash(m1 | m2 | ...))}, S_c

Server Finished: E_{Kms}(\text{Hash(m1 | m2 | ...))}, S_s

LSB 1 = Yes
LSB 0 = No
What are you doing?

Making the world a weirder place.

Starting with my eBay feedback page.

Comments:
Instead of office chair package contained bobcat.

Would not buy again.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Server Staples</td>
<td>—</td>
<td>—</td>
<td>Server sends client a bobcat</td>
<td>Client sends server candy</td>
</tr>
<tr>
<td>Server Does Not Staple</td>
<td>Client sends server a bobcat</td>
<td>—</td>
<td>Server sends client candy</td>
<td>—</td>
</tr>
</tbody>
</table>
Coming to an IETF standard near you!
Fixing OCSP for Fun and Profit

David Adrian
@davidcadrian