

Crypto API for Web Application Client-side Instances (aka “web pages”)

Jeff Hodges channeling Stephen Farrell, Sean Turner, Peter Saint-Andre
IETF Security Area and Applications Area

Position Paper for W3C Workshop on Identity in the Browser
May 24 and 25, 2011 – Mountain View, CA

Given that...

- **Web App Client-side instances** (hereinafter “web pages”, or “page”) **sometimes need to do things like...**
 - Cryptographically sign some data, and/or,
 - Verify a cryptographic signature, and/or,
 - Encrypt/Decrypt some data
- **E.g...**
 - encrypted & integrity-protected
 - Local data storage
 - Web app state management (aka “cookies”)
 - Web app authn & authz
 - Netflix “device id”

Then...

- Typical software development needs arise...
 - Should everyone and their brother and cousins invent their own crypto API, and,
 - Implement their own crypto primitives (hopefully derived from proven algorithms)?

And...

- Is it really a good idea for “web pages” to dynamically – potentially *insecurely* – obtain crypto implementations ?

- e.g.:

- Should folks jam this in their web apps:

```
<script src=  
https://github.com/bitwiseshiftleft/sjcl/raw/master/core/aes.js"/>
```

- Is the above a good idea?
- Or how 'bout this..

```
<script src="http://www.hanewin.net/encrypt/rsa.js"/> ?
```

Good thing about “implementations” ...

- ..is having so many to choose from...
 - <https://github.com/christkv/node-pure-crypto>
 - <http://code.google.com/p/jsencrypt/>
 - <http://www.hanewin.net/encrypt/>
 - <http://crypto.stanford.edu/sjcl/>
 - Etc...

The big question

- General agreement we need to do something more coherent WRT having crypto available to client-side web app implementations
- Various APIs + Implementations available today
- *Who* do we put into a room to coalesce it, and
- *Where* is the room – e.g. W3C, IETF, WhatWG, none, etc. ???