Semantic Insecurity
Security and the Semantic Web

Harry Halpin
@harryhalpin
Harry.halpin@inria.fr
harry@w3.org
What are Open Standards?

- An open collaborative process
- Wide Review
- Easy identification of the 'spec'
- Interoperability
- Anti-Trust
- Patents
Internet Engineering Task Force (IETF)

- Review by Internet Architecture Board
  "Anarcho-Hippie Commune?"
  https://www.ietf.org/

- Process around Request for Comments (RFC) (Informational Draft → Internet Standard)

- Lots of mailing-list discussion

- Quarterly Face-to-Face Meetings
  (Conflicting with PETS!)

- Mandatory Security Considerations
World Wide Web Consortium

- Directed by Tim Berners-Lee
  https://www.rfc-editor.org/rfc/rfc3979.txt

- Clear Process (Working Draft → Recommer
  https://www.w3.org/Consortium/Patent-Policy-20040205/

- Formal Objections.

- Uses Github

- Interoperability Requirements Before Standardization

- Focussed on Web Browsers (HTML, CSP, WebCrypto, etc.)

- WHATWG (implementer-driven version of HTML):
  https://whatwg.org/
W3C Process

- 400-ish members vote on a charter
- Charter made by W3C Team and members.
- IPR commits made to charter, then Rec.
- Drafts (Member Submissions) can be submitted anytime
- Last Call (comments from public) - 95 new bugs for WebCrypto
- Candidate Rec: Testing
- Proposed Recommendation: IPR commits from WG members
Software Patents

• IETF Note Well:
  https://www.rfc-editor.org/rfc/rfc3979.txt

• W3C Royalty-Free Patent Policy
  https://www.w3.org/Consortium/Patent-Policy-20040205/

• Applications to Crypto Patents?
Snowden's NSA Revelations

The United States was monitoring ALL non-encrypted internet traffic
Mass Surveillance is an Attack

- Pervasive Monitoring is an Attack (RFC 7258)
- Reversing the Traditional Crypto Flow (NIST/NSA → Standards) from Community → Standards
- CFRG RFC on Curve25519 (RFC 7748)
- https://datatracker.ietf.org/doc/rfc7748/
- IRTF Human Rights Considerations of Protocols (HRPC)
  https://datatracker.ietf.org/rg/hrpc/charter/
- TLS 1.3, etc.
Public and Private Keys
Sending an Encrypted Message

Hello Alice

x0Ak3o$2Rj

Hello Alice

x0Ak3o$2Rj
Encryption and Standards

“ I worked with the National Security Agency on the design of a secured version of the internet but we used classified security technology at the time and I couldn't share that with my colleagues. If I could start over again I would have introduced a lot more strong authentication and cryptography into the system.”

Vint Cerf (Google Hangout)
Semantic Security

“Whatever is efficiently computable about the cleartext given the ciphertext, is also efficiently computable without the ciphertext”


- Define security properties in terms of adversary (threat models).
- The foundation of modern cryptography are proofs that show that even the most powerful adversary cannot break the protocol
- The foundation for Transport Layer Security (TLS) proofs for TLS 1.3 (HTTPS)
THIS IS THE SIXTH TIME WE HAVE TRIED TO ENCRYPT THE NET
AND WE HAVE BECOME EXTREMELY EFFICIENT AT IT.
The W3C Web Cryptography API

- No access to real cryptography in the browser – until now!
- Started in 2012 by Web Cryptography Working Group, finishing now
- Editors: Ryan Sleevi (Google) and Mark Watson (Netflix)
- Constant time functions for cryptographic primitives (RSA, ECDSA, ECDH, SHA256, ..)
- Cryptographically strong pseudo-random number generation (assuming entropy)
- Key storage in localStorage
- Already in major browsers
Passwords are broken
Abstract

This specification defines an API enabling the creation and use of strong, attested, cryptographic scoped credentials by web applications, for the purpose of strongly authenticating users. Conceptually, one or more credentials, each scoped to a given Relying Party, are created and stored on an authenticator by the user agent in conjunction with the web application. The user agent mediates access to scoped credentials in order to preserve user privacy. Authenticators are responsible for ensuring that no operation is performed without user consent. Authenticators provide cryptographic proof of their properties to relying parties via attestation. This specification also describes...
W3C Web Authentication

Test of User Presence → Public-Key Crypto

- Password Reuse
- Phishing
- Interception
W3C Web Authentication

"I promise a user is here",
"the server challenge was: 337423",
"the origin was: accounts.google.com",
"the TLS connection state was: 342384"
W3C Web Authentication Works

“I promise a user is here”,
“the server challenge was: 337423”,
“the origin was: accounts.google.com”,
“the TLS connection state was: 342384”.

Different key for different origins (relying parties).

this is where the key is

this guy knows the key
W3C Web Authentication

“I promise a user is here”,
“the server challenge was: 337423”,
“the origin was: accounts.google.com”,
“the TLS connection state was: 342384”

Makes this phishing-resistant.
W3C Web Authentication

“I promise a user is here”,
“the server challenge was: 337423”,
“the origin was: accounts.google.com”,
“the TLS connection state was: 342384”

Makes this MITM-resistant.
DRM, Encrypted Media Extensions, and W3C

- Encrypted Media Extensions = Shim to Content Decryption Module (DRM) in every browser.
- To prevent copying of videos
- Started as HTML Extension, now a Candidate Recommendation
- Recommendation in December 2016.
- EFF wants to have a protection for security researchers and end-users.
- https://www.w3.org/html/wg/
• Protecting Security Researchers Petition

• [URL] - researchers tell W3C to protect researchers who investigate browsers

• Email: cory@eff.org

• Subject: W3C Security Researchers Sign On

Signatories:

Bruce Schneier, USA

Alan Cox, UK, Honorary Fellow University of Wales: Trinity St David

Emiliano DeCristofaro, UK, University College London

Dr Steven J. Murdoch, UK, Principal Research Fellow, University College London

Harry Halpin, France, INRIA

Ian Goldberg, Canada, University of Waterloo

Ron Deibert, Canada, Professor of Political Science and Director of the Citizen Lab at the University of Toronto

Jon Andersen, USA

Sergey Bratus, USA, Research Associate Professor, Computer Science Department, Dartmouth College

Joel R. Voss, USA

Paul Garrett Hugel, USA

Jacob Appelbaum, Germany, the Tor Project

Roger Dingledine, USA, the Tor Project

Ronald L. Rivest, USA, MIT

Prof. Dr. Tanja Lange, The Netherlands, Technische Universiteit Eindhoven

Frederic Jacobs, Switzerland, Swiss Institute of Technology (EPFL)

Dr Ian Brown, UK, Oxford Internet Institute, Professor of Information Security and Privacy, University of Oxford
Protest at W3C Advisory Council Meeting

W3C Encrypted Media Extension Factsheet:
https://www.w3.org/2016/03/EME-factsheet.html
Standards: Good, Bad, and Ugly

The Good:

- WebCrypto:
- Web Application Security (CSP, SRI, etc.):
- Web Authentication
- Security/Privacy Review: https://w3ctag.github.io/security-questionnaire/

The Bad:

- Encrypted Media Extensions
- Privacy leaks (Battery API, etc.)
RDF and the Semantic Web

http://www.w3.org/2000/10/swap/pim/contact#Person
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/People/EM/contact#me
http://www.w3.org/2000/10/swap/pim/contact#fullName
http://www.w3.org/2000/10/swap/pim/contact#mailbox
mailto:em@w3.org
http://www.w3.org/2000/10/swap/pim/contact#personalTitle

Dr.

Eric Miller
Linking Data on the Semantic Web with \texttt{owl:sameAs} and “follow your nose”
URIs identify real world things: Are they names or web-pages?
Linked Data: Names for 'real world things' should retrieve web-pages but be different than the web page.
TLS Everywhere - Including the Semantic Web

- \texttt{https://www.example.org} vs. \texttt{http://www.example.org}
- Less than 1\% of Semantic Web URIs use HTTPS
- If you don't use TLS, then \textit{ANYONE can intercept and ALTER your connection}
- Hard to Update (HSTS, Upgrade Insecure Requests) without first sending cleartext HTTP headers - that can \textit{also be intercepted and altered.}
- Worse with Linked Data, as you are supposed to follow URIs for new data when you encounter them.
- SOLUTION: Use TLS for ALL new Semantic Web URIs
- When given a HTTP URI, see if you can find a HTTPS URI
WebID+TLS

- Associate a URL for a person http://www.example.org/#me with a public key.
- Use that public key to authenticate to websites using a self-signed client certificate with <keygen> HTML tag
- Signatures done using insecure MD5 hash function.
- <keygen> deprecated by Chrome, will soon be deprecated by Firefox.
- Simply use W3C Web Authentication instead!
Securing Semantics

- Use HTTPS for URIs (TLS 1.3)
- Don't create your own protocols
- Don't use Legacy Algorithms
- Use Web Authentication
- For new protocols, use W3C Web Crypto
- Think about privacy and security