Client Certs -- the old-new thing

CAcert

The Community CA
iang @ cacert.org

Authentication v0.0 to ...

- Login 0.0: everyone is trusted
- Login 0.1 passwords + usernames
- Login 0.3 SSO *(single-sign-on)* – the dream!
- Login 0.4 Federation...

What went wrong?

- 0.0 everyone is … Untrusted
- 0.1: passwords
  - N * complexity != support + security
- 0.3, 0.4 SSO + Federation

  a) Every site, a method...   (chicken)

  b) Every person, a method... (egg)

  c) Who's got my data?

  d) Who's got customer?

Haven't we got computers to deal with this authentication stuff?

- We have!
- They are called “client certificates”
  - public-private-key pairs,
  - (third party) signatures
- Really, they are like “crypto-passwords”
- Every browser, every webserver
- Why didn't they work?

Why Client Certs didn't work

- Enough software ...
- Data isn't at risk, nor customers
- every person needed a cert
  - Which was a drag … did not scale
- Problem (b) nobody had an egg.
  - Chicken & egg
- *(Don't ask.)*

CAcert gets into the Egg business

- Certificates => “Identity” => Assurance
  - The “*web of trust*”
- Audit!
  - How do you audit a web of trust?
  - Doco … standards … verifiability …
- CATS == CAcert Automated Testing System
  - All Assurers must be challenged!

Inspiration!

- CATS requires a client cert (no passwords)
  - Because we are a CA?
  - So our Assurers know about certs?
  - We want to look cool?
  - We want high-security access?
  - Or?
  - *Don't ask...*

The success of CATS

- Went live early 2008
- Obligatory early 2009
- 10k++ → 1000 → 2000 → 3000
    - Today: 3320 or so
    - Rule of thumb: serious test reduces to 1/3
- Assurer community is stronger

CAcert gets into the Chicken business

- Every Assurer has a cert!
- Therefore... every site can use certs (only)
- Migrate all to cert usage (only)
- Wordpress, Sympa, Voting … DONE!
- It's on the sysadm work list

Results... for the blog!

- Write-access if you have a cert
    - More authors, more articles...
- Spam is solved.
- No more lost-account, bad password problems
- → administrator is doing other things
- No more long arguments about WHO
- → users spend more time on articles...

Gotchas!

- #1 Multiple certs → Firefox confusion
  - (We're waiting for user-whitelisting)
- #2 Crazy messages...
  - Server rejects cert
  - Client says Server rejected handshake
  - User rejects it all...
  - Developers don't/won't agree on blame…
  - (wait for more user complaints)

Strategy 1. Hybrid

- 1. Hybrid: Password PLUS certs
  - Attack between the gaps: HTTP+HTTPS
  - Sounds a bit like phishing...
  - Forever coding the border
- Only if you know you have to.
  - (CA main site does this, for recovery)

Strategy 2. Only Certs

Only SSL, only Certs, always certs:

- 2.a. Apache does processing
  - (too little, too much)
- 2.b. App does processing
  - (gotta write some code)
- 2.b Recommended!

Strategy 2.b in Depth

- Gotcha #3: certs can & will change!
- Read cert into Database
    - (cert indexes → to account)
- For new Certs, scan for same details
    - Can match on email, and Name.
- If user changes Name & email …
    - More thinking required

Conclusion

- Certs do Work
  - Much better than passwords
  - Much less hassle once going...
  - Much easier on administrators
- Against other methods?
  - Higher security than OpenID
  - Available (once you buy some eggs...)

Your Challenge

- Problem (b): nobody's got an egg...
- Challenge for you: get certs to all users
    - “all are Cacert...” (borrow)
    - Build a site, any site, use certs (pull)
    - Internal: use factory certs (push)