Structurally Insecure?

Several paradoxes in the market for Certificate Authorities, and some ideas for resolving them

Peter Eckersley
pde@eff.org
TLS/SSL is really valuable

Millions of services use it

It is a necessary condition of most secure uses of the Internet
Encryption is only as good as your ability to verify the other party
Currently, most uses of TLS (web, email, IM, many VPNs) rely on PKIX.
What is PKIX? Where did it come from?
Invented by Netscape, early 90s

In a hurry

Specifically to protect credit card #s

Before strong crypto was legal
Based on X.509, an ITU standard from the 1980s

(which predates the Web!)
Security via digital paperwork

X.509 certs can (and do) contain just about anything
#!/usr/bin/env python

# diversity.py -- estimate the number of different certificate types and
# combinations of fields in them

from dbconnect import dbconnect

db,dbc = dbconnect()
q = ""
SELECT *,`X509v3 extensions:X509v3 Key Usage`,
`X509v3 extensions:X509v3 Extended Key Usage`,
`X509v3 extensions:X509v3 Basic Constraints:CA`,
`X509v3 extensions:Netscape Cert Type`
FROM all_certs
WHERE certid >= %d and certid < %d
"

dbc.execute("SELECT count(certid) from all_certs")
n = int(dbc.fetchone()[0])
print n, "rows"

fset = {}
for i in range(n / 1024):
    q1 = q % (i* 1024, (i+1) * 1024)
    dbc.execute(q1)
    batch= dbc.fetchall()
    for row in batch:
        cert, type_fields = row[:-4], row[-4:]
        bits = 0
        for field in cert:
            if field==None:
                bits |= 0x01
            elif type(field) == str and ("critical" in field):
                bits |=0x02
                bits <<= 2
        key = (type_fields, bits)
        fset[bits]=True

print len(fset)
By this approximate measure:

10,320 kinds of X.509 certs were observed

1,352 kinds were sometimes valid

Not as bad as a million kinds, still hard to process automatically
How were the X.509 certificates to be issued?
Via a market!
Markets are a great idea!
CAs can compete with each other to offer the best services at the best prices
Competition will drive down prices,

and

there will be different levels of certification services
Expensive, thorough validation
Cheaper, weaker validation
Actually that isn't how the CA market works at all!
How the CA marketplace works
Regardless of what you pay for, you get every restaurant's food mixed together.
But PKIX is holding us back

Thesis: this is a product of the PKIX market structure
4 problems
Problem 1
TLS/SSL Authentication
Hundreds of CAs
Private keys safely locked in an HSM?
Domain validation

Alice asks for cert
CA emails root@domain.com
Someone clicks on link in email
CA gives cert to Alice
Attack surface:

The CAs
Their routers
Their ISPs' routers...
All of the DNS infrastructure
Apparently, ~52 countries
Problem 2
There is a problem with this website's security certificate.

The security certificate presented by this website was not issued by a trusted certificate authority.

Security certificate problems may indicate an attempt to fool you or intercept any data you send to the server.

We recommend that you close this webpage and do not continue to this website.

- Click here to close this webpage.
- Continue to this website (not recommended).
- More information
Fetch can not verify the identity of server "ftps-masquerade.fetchsoftworks.com".

An FTP connection to "ftps-masquerade.fetchsoftworks.com" could not be opened because the hostname of the server's SSL certificate does not match the hostname used to connect to the server.

Click Show Certificate for more information and an opportunity to have this certificate always accepted. Click Continue to proceed, or Cancel to disconnect.
99% of the time, this is not an attack!
Certificate warnings are useless for humans!
Surveillance and censorship by governments
X.509 PKI was not designed to survive this!
Examples
Problem 1 & 3
Problems 2 & 3

Syria attacks Facebook

---

### Certificate Details

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Issued To</strong></td>
<td>s.static.ak.facebook.com</td>
</tr>
<tr>
<td><strong>Common Name (CN)</strong></td>
<td>s.static.ak.facebook.com</td>
</tr>
<tr>
<td><strong>Organization (O)</strong></td>
<td>Facebook, Inc.</td>
</tr>
<tr>
<td><strong>Organizational Unit (OU)</strong></td>
<td>Facebook</td>
</tr>
<tr>
<td><strong>Serial Number</strong></td>
<td>00:C6:4F:50:11:B3:65:DC:B9</td>
</tr>
<tr>
<td><strong>Issued By</strong></td>
<td>s.static.ak.facebook.com</td>
</tr>
<tr>
<td><strong>Common Name (CN)</strong></td>
<td>s.static.ak.facebook.com</td>
</tr>
<tr>
<td><strong>Organization (O)</strong></td>
<td>Facebook, Inc.</td>
</tr>
<tr>
<td><strong>Organizational Unit (OU)</strong></td>
<td>Facebook</td>
</tr>
<tr>
<td><strong>Issued On</strong></td>
<td>01/05/2011</td>
</tr>
<tr>
<td><strong>Expires On</strong></td>
<td>30/04/2012</td>
</tr>
<tr>
<td><strong>Fingerprints</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Issued To</strong></td>
<td><a href="http://www.facebook.com">www.facebook.com</a></td>
</tr>
<tr>
<td><strong>Common Name (CN)</strong></td>
<td><a href="http://www.facebook.com">www.facebook.com</a></td>
</tr>
<tr>
<td><strong>Organization (O)</strong></td>
<td>Facebook, Inc.</td>
</tr>
<tr>
<td><strong>Organizational Unit (OU)</strong></td>
<td>&lt;Not Part Of Certificate&gt;</td>
</tr>
<tr>
<td><strong>Issued By</strong></td>
<td>DigiCert High Assurance CA-3</td>
</tr>
<tr>
<td><strong>Common Name (CN)</strong></td>
<td>DigiCert High Assurance CA-3</td>
</tr>
<tr>
<td><strong>Organization (O)</strong></td>
<td>DigiCert, Inc.</td>
</tr>
<tr>
<td><strong>Organizational Unit (OU)</strong></td>
<td>DigiCert, Inc.</td>
</tr>
<tr>
<td><strong>Issued On</strong></td>
<td>15/11/2010</td>
</tr>
<tr>
<td><strong>Expires On</strong></td>
<td>03/12/2013</td>
</tr>
<tr>
<td><strong>Fingerprints</strong></td>
<td></td>
</tr>
</tbody>
</table>
Some cases of problem 3 to consider
Internet censorship in...

Australia (?!), Bahrain, Burma, China, Cuba, Iran, Syria, Thailand, Turkey, UAE, USA (?!)

...and dozens of other countries
Kazakhstan, June 2011:
demand for access to all Kazakh Google users' data

(Google briefly unplugged google.kz)
Note:

The Kazhakh government can control https://google.kz if it wishes
Google cannot stop users from typing “google.kz” into a browser
The lesson of vb.ly
The lesson of vb.ly?
Who else uses .ly?
bit.ly – 2 billion URLs / month
bit.ly – we were lucky!
The country ↔ TLD mapping was a cute idea
That ship has sailed
Problem 4
HTTP needs to be retired
That won't happen with PKIX
Solutions to these problems?
For Problem 1...
<table>
<thead>
<tr>
<th></th>
<th>DANSE</th>
<th>Pinning</th>
<th>Perspectives/Convergence</th>
<th>Sovereign Keys</th>
<th>Cert Transparency</th>
<th>Mecai</th>
<th>Decentralised Observatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protects against attacks owning the server's upstream router</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>Protects against victim's DNS zone / gov'ts with leverage</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>not really</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>Copes with messy/partial cert rollover</td>
<td>?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>HTTPS server admins don't need to do anything</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>sort of</td>
</tr>
<tr>
<td>Works for all SSL/TLS protocols (not just HTTPS)</td>
<td>Y</td>
<td>N</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Protects 1st connection</td>
<td>Y</td>
<td>If preloaded</td>
<td>Y</td>
<td>Y</td>
<td>undefined</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Protection, once in place, is automatic and robust</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Avoids cert warnings</td>
<td>N</td>
<td>N</td>
<td>some</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

Also: NameCoin, other projects?
But what about problem 4?