PKI AND TLS IN THE REAL WORLD

HÅVARD RADDUM
SIMULA@UIB
PKI IN THE REAL WORLD

OVERVIEW

- PKI model (Public Key Infrastructure)
- Three PKI/TLS stories from the real world
  - CloudFlare’s Flexible SSL
  - WoSign and SHA1
  - Lenovo, Superfish and Komodia
PKI IN THE REAL WORLD

PKI MODEL
SECURE CONNECTIONS ON THE INTERNET

- Client and server wants to communicate via secure connection

- Want:
  - Encryption - No one else learns the content of the communication
  - Integrity - Receiver knows communication has not been changed after it left the sender
  - Authentication - You know you are communicating with the intended server
PKI IN THE REAL WORLD

EASY AND HARD PROBLEMS

- **Easy:** Encryption and integrity
- **Hard:** Authentication

How to separate
PKI IN THE REAL WORLD

AUTHENTICATION SOLUTION

- Server has private key, client knows servers public key
- Server signs some data when initialising session
- Client verifies signature
NEW PROBLEMS

- Q: How does client find server’s public key?
  - A: Server has certificate, stating it owns the public key written on the certificate

- Q: How does client know certificate is genuine? (and not made by attacker posing as server)
  - A: Certificate is issued and digitally signed by a certification authority (CA). Client can get CA’s certificate and verify signature
PKI IN THE REAL WORLD

NEW PROBLEMS

Q: How does client know CA’s certificate is genuine? (and not made by attacker)

A: CA’s certificate is issued and signed by a root CA. Client gets root CA’s certificate and verifies signature

Q: How does client know root CA’s certificate is genuine?

A: Because!! … The root CA’s certificate is self-signed. Just trust the root CA, OK?
In PKI model root CA’s are trust anchors. We just have to somehow know that their certificates are genuine. We must also trust the root CA and all CAs below it to always act honestly and only include true information on certificates.
Web browsers/operating systems come with a long list of pre-installed root certificates.

We implicitly trust all of these to always act honestly and not undermine the PKI model.

Root CAs come from all over the world.

Regular users know nothing about them.
### PKI in the Real World

#### Certificate Authority List

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA Certificate Services</td>
<td>certifikat</td>
<td>1 Jan. 2020, 00:05:59</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>Actives Authentication Root CA</td>
<td>certifikat</td>
<td>22. sep. 2030, 09:21:23</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>AddTrust Uda 1 CA Root</td>
<td>certifikat</td>
<td>30. mar. 2010, 12:00:01</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>AddTrust External CA Root</td>
<td>certifikat</td>
<td>22. nov. 2030, 12:48:53</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>Admin-Root-CA</td>
<td>certifikat</td>
<td>17. nov. 2001, 08:17:17</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>AffirmTrust Commerz</td>
<td>certifikat</td>
<td>31. dec. 2030, 15:05:06</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>AffirmTrust Networking</td>
<td>certifikat</td>
<td>31. dec. 2030, 12:03:24</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>AffirmTrust Premium</td>
<td>certifikat</td>
<td>31. dec. 2030, 12:30:34</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>AffirmTrust Premium ECC</td>
<td>certifikat</td>
<td>31. dec. 2040, 13:02:34</td>
<td>Systemmønster</td>
</tr>
<tr>
<td>AKS, Stina, Root CA</td>
<td>certifikat</td>
<td>1. jan. 2020, 12:48:21</td>
<td>Systemmønster</td>
</tr>
</tbody>
</table>
CLOUDFLARE'S FLEXIBLE SSL
CloudFlare is a company based in USA selling services in web-hosting.

One of their products is «one-click SSL», or «flexible SSL».

- Web server www.example.com using «flexible SSL» do not need to get a certificate from a CA.
- Do not need to enable SSL/TLS on www.example.com, do not need to have a private key.
- Browser of visitors to www.example.com will still show the connection to be valid TLS!
CLOUDFLARE’S SSL AS A SERVICE (2014)

One-click SSL (Pro, Business and Enterprise)

Without CloudFlare
To add SSL, purchase a certificate and install it on your server.

With CloudFlare
Easy CloudFlare SSL with an existing certificate from your hosting provider or self-signed certificate.

Flexible SSL:
There is an encrypted connection between your site visitors and CloudFlare, but not from CloudFlare to your server.

- You do not need an SSL certificate on your server.
- Visitors will see the SSL lock icon in their browser.

www.cloudflare.com
Flexible SSL

Flexible SSL encrypts traffic from Cloudflare to end users of your website, but not from Cloudflare to your origin server. This is the easiest way to enable HTTPS because it doesn’t require installing an SSL certificate on your origin. While not as secure as the other options, Flexible SSL does protect your visitors from a large class of threats including public WiFi snooping and ad injection over HTTP.

www.cloudflare.com
HOW DOES FLEXIBLE SSL WORK?

- Visitors to www.example.com have a valid TLS-connection to CloudFlare, even though address bar in browser says www.example.com.

- CloudFlare decrypts visitor’s traffic and sends it in plaintext to www.example.com.

- Certificate CloudFlare is using is issued to CloudFlare, but has www.example.com as an «alternative name» in Subject Alt Name extension.
PKI IN THE REAL WORLD

FLEXIBLE SSL EXAMPLE

- **https://www.icontrolwp.com/** is using CloudFlare’s flexible SSL certificate

- Visit site and examine certificate
CloudFlare’s certificate can be used for any of these domains.
Comodo is the CA issuing CloudFlare’s flexible SSL certificates

Flexible SSL certificates issued to CloudFlare has a long list of domains in Subject Alt Name extension

Subject Alt Name extension supposed to contain aliases and specific domains for the certificate owner

Comodo is signing on information it knows is not true, or Subject Alt Name extension has lost its meaning
FLEXIBLE SSL PRO AND CON

- Supporters of Flexible SSL:
  - A web site not buying a certificate and configuring their server for TLS will only be able to communicate unencrypted
  - Flexible SSL is an improvement, because traffic is encrypted at least part of the way

- On the other hand:
  - The user is being lied to when Flexible SSL is in use
PKI IN THE REAL WORLD

VIOLATION OF GOOGLE’S CA POLICY?

- Excerpt from Google’s root certificate policy:

It is imperative that a user of Google Chrome can be confident that when proper SSL indications are shown in the browser, the user is in fact communicating with the intended site and not an attacker or other man-in-the-middle using a root certificate obtained improperly from a CA. Anything that contravenes this principle, including issuance of certificates for a website to a party other than the legitimate operator of that website, or delegation of authority that results in the issuance of certificates for a website to a party other than the legitimate operator of that website, is a serious violation of trust that will be dealt with in accordance to this policy.

http://www.chromium.org/Home/chromium-security/root-ca-policy

- So why does Google Chrome accept CloudFlare’s and Comodo’s practice?
WOSIGN + SHA-1 = TRUE
WoSign and StartCom

- WoSign is a Chinese CA and StartCom is (was) an Israeli CA
- In November 2015 WoSign acquired StartCom
- When one CA buys another it must be publicly disclosed to the CA/Browser forum
- For unknown reasons, WoSign has tried to keep the acquisition secret and has argued StartCom continues to operate as an independent CA
- Team from Mozilla has gathered lots of evidence that StartCom is using (a copy of) WoSign’s infrastructure for certificate creation
TLS AND PKI IN THE REAL WORLD

SHA-1

- SHA-1 is a hash algorithm, can be used to generate digital signatures for certificates
- The security of SHA-1 has deteriorated to a level where the CA/Browser forum wants to phase out SHA-1
- Decided that certificates issued in 2016 and later can not use SHA-1 for digital signing; browsers will object
- May be costly for customers of CAs to upgrade their software to support accepted hash algorithms
WOSIGN AVOIDING SHA-1 BAN

- Discovered in September 2016 that WoSign has issued certificates in 2016 using SHA-1 in signature algorithm

- Avoid triggering browser alarm by back-dating validity period to start in late 2015

- Evidence these certificates were manually modified and not automatically generated by WoSign’s system
MOZILLA RESPONSE

- Team from Mozilla lead investigation on WoSign’s practice
- Team recommends that Mozilla products start to distrust WoSign/StartCom certificates
- Team also suggests to no longer accept audits from Ernst & Young’s Hong Kong office.
LENOVO, SUPERFISH AND KOMODIA
LENOVO, SUPERFISH AND KOMODIA

- Lenovo is a PC manufacturer from China
- In the period August 2014 - January 2015 Lenovo sold Windows laptops with pre-installed root certificate from Superfish
- There is no real CA named Superfish
- Laptops also had software that would include a proxy on the laptop as a man-in-the-middle in any TLS-connection
A Superfish-infected laptop contains a proxy intercepting all web traffic between browser and website.

Proxy also acts as a local CA.

When browser wants to set up TLS to some web site:

- Regular TLS-connection set up between proxy and web site.
- Proxy issues certificate on the fly for web site, signed by Superfish certificate.
- Proxy sets up TLS-connection between itself and browser, using the just-issued certificate.
PKI IN THE REAL WORLD

HOW IT WORKS – MAN IN THE MIDDLE

PC

BROWSER

Issuer: Superfish
Subject: Superfish
Public key
Signature

TLS request

Superfish
TLS connection

ISSUER: Superfish
SUBJECT: Superfish
PUBLIC KEY
SIGNATURE

PROXY

Private key

WEB SITE

Issuer: Real CA
Subject: Web site
Public key
Signature

Real TLS connection

TLS request
WHAT IS THE SECURITY PROBLEM?

- Anyone knowing the private key for the Superfish root certificate can issue certificates for any web site
- These certificates will be accepted as valid by all Superfish-infected laptops
- Attacker may become man-in-the-middle in a TLS connection to any web site, if he knows private key for Superfish root certificate
PKI IN THE REAL WORLD

SUPERFISH PRIVATE KEY

- Private key resides in the proxy on every Superfish-infected laptop
- Private key protected by password
- One analyst found the private key, working only for three hours, password was ‘komodia’
- Komodia is name of company developing the TLS-proxy
- The same private key used on all infected laptops
- Private key for forging certificates is public knowledge!
PKI IN THE REAL WORLD

WHY DID THEY DO IT?

- Superfish dynamically adds advertisements to web pages
- **Problem for Superfish:** How to add advertisement in web page secured by TLS?
- **Solution:** Set up proxy/local CA on each machine
  - Decrypt TLS-protected web page
  - Add the advertisement
  - Re-encrypt web page for the proxy - browser link
PKI IN THE REAL WORLD

WHY DID LENOVO PLAY ALONG?

- **New problem for Superfish:** How to get Superfish root certificate into OS of laptops?
  - Adding root certificates requires privileged access

- **Solution:** Have root certificates (and proxy) pre-installed from PC manufacturer

- Estimated that Lenovo made approximately $250,000 by agreeing to produce Superfish-infected laptops
LENOVO AND SUPERFISH RESPONSE

- Both Lenovo and Superfish claimed initially there was no security problem with their practice when it made headlines.

- After attack scenarios were presented in detail:
  - Superfish blamed Komodia for «inadvertently introducing the vulnerability».
  - Lenovo cut off all cooperation with Superfish, and started helping customers remove proxy and root certificate from infected machines.
  - Komodia did not make any comments.
PKI IN THE REAL WORLD

- Komodia is an Israeli company
- Front page of komodia.com:

Komodia’s SSL Digestor

Our advanced SSL hijacker SDK is a brand new technology that allows you to access data that was encrypted using SSL and perform on the fly SSL decryption. The hijacker uses Komodia’s Redirector platform to allow you easy access to the data and the ability to modify, redirect, block, and record the data without triggering the target browser’s certification warning.

www.komodia.com
When PKI model comes in conflict with business, creative solutions are found to circumvent TLS limitations

Solutions may undermine TLS security

CAs may very well play an active role

The trust assumption in PKI model does not fit the real world; the vast majority of users know almost nothing about PKI, and absolutely nothing about which CAs to trust or not