TRANSPORT LAYER SECURITY BEYOND CRYPTO – NOTARIES AND PINNING TO THE RESCUE?

Artemios G. Voyiatzis
“If you think cryptography is the answer to your problem, then you don't know what your problem is.”
Who is the weakest link in security?
The app landscape

- >2 billion smartphones
- >2.2 million smartphone applications (apps)
- Capture and process sensitive user information
- Transfer information to/from remote sites
Is my communication secure?
TLS connection setup

- Handshake protocol
- Four phases
- In phase 2:
  - Server sends a certificate
Certificate validation before crypto

- Use the CA information of the trust store
- The client checks the validity of the server certificate
- Is the certificate authentic?
  - Is it signed by a trusted Certificate Authority?
  - Does the hostname matches the subjectAltname or CN?
  - Is it expired or still valid?
  - Is the certificate revoked?
SSL/TLS & Android apps

- Default HTTPS API in Android implements proper certificate validation

What could go wrong?
The central role of CAs
Who signed these certificates?

Chung et al., Measuring and Applying Invalid SSL Certificates: The Silent Majority, IMC 2016, November 14-16, 2016, USA.

Fadai et al., Trust me, I’m a Root CA! Analyzing SSL Root CAs in Modern Browsers and Operating Systems. ARES 2015, August 24-28, 2015, France.
Custom validation

• Fahl et al. (2012): Tested 13,000 apps
  o A 1,000 of them improperly handled validation

• In 2013, they asked the developers

“We added this piece of code because our client uses an SSL certificate for his web-service which was signed by a certificate authority that is not pre-installed on Android and actually we did not realize that this would cause such trouble.”

“This app was one of our first mobile apps and when we noticed that there were problems with the SSL certificate, we just implemented the first working solution we found on the Internet. […] We usually build Java backend software for large-scale web services.”

But things improve, don’t they?

- Experiment on 50,000 Android apps
  - Top 25,000 from Q4/2013
  - Top 25,000 from Q4/2014
- Test using Mallodroid script
- Focus explicitly on custom TrustManager implementations

---

Results

Apps 2013

- Trust Manager: 17% Broken, 6% Possibly Broken
- Hostname Verifier: 7% Broken, 1% Possibly Broken
- TLS Error Handling: 15% Broken

Legend: Broken, Possibly Broken
Results

Apps 2014

- Trust Manager: 23% Broken, 10% Possibly Broken
- Hostname Verifier: 13% Broken, 4% Possibly Broken
- TLS Error Handling: 29%

Legend:
- Blue: Broken
- Light Blue: Possibly Broken
How can we fix this for user?

- PinningTrustManager PoC code on github
- Device/OS-based rather than app-based (no hope)
- Defend against developer errors in cert. handling
- Combines dynamic instrumentation techniques and cert. pinning
- User is alerted if cert. changes (e.g., injected)
  - Still chance of TOFU pinning

---

Android 7.0 Nougat and pinning

- New approach – config file
- Much easier implementation/integration

```xml
<domain-config>
  <domain includeSubdomains="true">example.com</domain>
  <trust-anchors>
    <certificates src="/raw/my_ca"/>
  </trust-anchors>
</domain-config>
</network-security-config>
```

```xml
<domain-config>
  <domain includeSubdomains="true">example.com</domain>
  <trust-anchors>
    <pin-set expiration="2018-01-01">
      <pin digest="SHA-256">7HlpacktIAq2Y49orFOOQKurWxmSFZ8BCoQy8/Rh83Y</pin>
      <pin digest="SHA-256">4wza0LMPX0uZvRCBE1+4PyuldP0cf3UKg0/04cDMioE</pin>
    </pin-set>
  </trust-anchors>
</domain-config>
</network-security-config>
```
Happy developers (?)

- No need for custom code
- But need to maintain two versions
- What happens when the cert. expires?
  - Recent case with Mozilla plug-ins
- How do you update apps with new files?
  - How do you force to update?
Is pinning enough?
TLS Notary Service
ICSI Certificate Notary

App/Web Server

Check stored pins
Happy users!

• Certificate pinned on first use
  o Or even deployed with app ;)
• Feed Notary before app deployment
• No user involvement in decision
  o Only if TOFU && !Notary
• Better usability and better security
• PoC code also on github
  o Require rooted device (Thanks Google)
  o Would love see it integrated in next Android OS 😊
Conclusion

Protocol security

App security

User security
Credits
Thank you!

Artemios G. Voyiatzis

avoyiatzis@sba-research.org
@a_voyages