*Experience talk/Tools*

# Clien-side SSL certificates validation

Tetiana Yarygina,
University of Bergen,
tetiana.yarygina@ii.uib.no

Suppose…

You are given a mobile application with client-server architecture and RESTful API

No source code available (proprietary software)

But you are very curious how it works

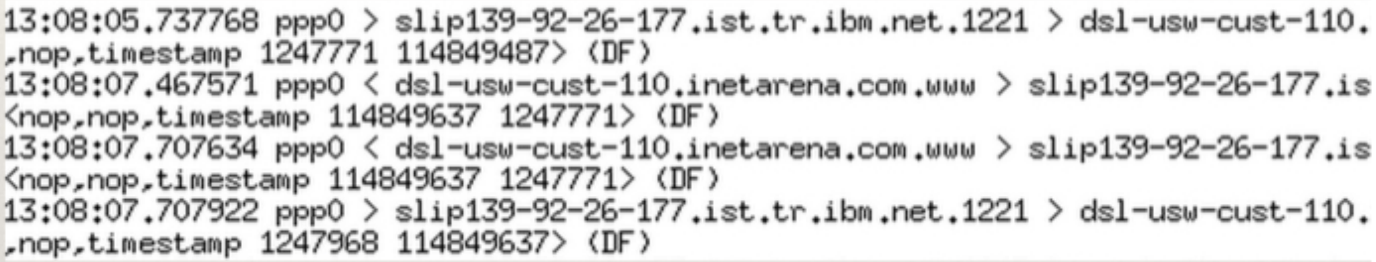
But you are too lazy to reverse engineer the binary

Brilliant idea appears…

Consider the application as a black box and temper with its input/output, e.g. intercept, analyse and modify its network traffic

# Network anlyzers

- Some of popular multifunctional analysers are:
  - tcpdump
  - Wireshark
  - …

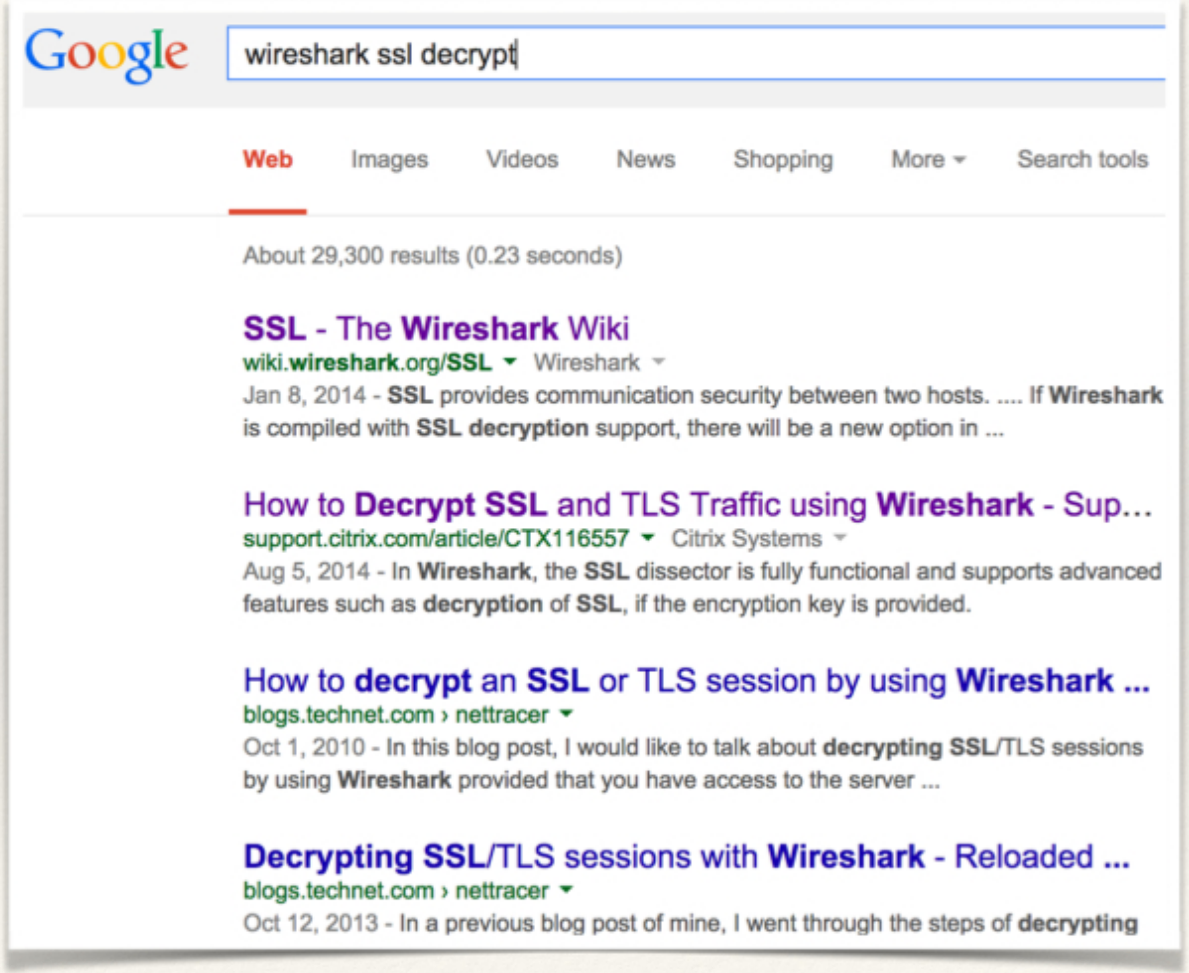


```
13:08:05.737768 ppp0 > slip139-92-26-177.ist.tr.ibm.net.1221 > dsl-usw-cust-110.
,nop,timestamp 1247771 114849487> (DF)
13:08:07.467571 ppp0 < dsl-usw-cust-110.inetarena.com.www > slip139-92-26-177.is
<nop,nop,timestamp 114849637 1247771> (DF)
13:08:07.707634 ppp0 < dsl-usw-cust-110.inetarena.com.www > slip139-92-26-177.is
<nop,nop,timestamp 114849637 1247771> (DF)
13:08:07.707922 ppp0 > slip139-92-26-177.ist.tr.ibm.net.1221 > dsl-usw-cust-110.
,nop,timestamp 1247968 114849637> (DF)
```

- You run Wireshark and (surprise!) realize that traffic of the application is encrypted
- So now you need to decrypt traffic somehow…

# Network anlyzers

- Google suggests 29,300 potential answers, which are mostly complains
- You feel that it is not right
- …and you decide to use some dedicated and more flexible software



wireshark ssl decrypt

About 29,300 results (0.23 seconds)

**SSL** - The **Wireshark** Wiki
wiki.**wireshark**.org/**SSL** ▾ Wireshark ▾
Jan 8, 2014 - **SSL** provides communication security between two hosts. ..... If **Wireshark** is compiled with **SSL decryption** support, there will be a new option in ...

How to **Decrypt SSL** and TLS Traffic using **Wireshark** - Sup...
support.citrix.com/article/CTX116557 ▾ Citrix Systems ▾
Aug 5, 2014 - In **Wireshark**, the **SSL** dissector is fully functional and supports advanced features such as **decryption** of **SSL**, if the encryption key is provided.

How to **decrypt** an **SSL** or TLS session by using **Wireshark** ...
blogs.technet.com › nettracer ▾
Oct 1, 2010 - In this blog post, I would like to talk about **decrypting SSL**/TLS sessions by using **Wireshark** provided that you have access to the server ...

**Decrypting SSL**/TLS sessions with **Wireshark** - Reloaded ...
blogs.technet.com › nettracer ▾
Oct 12, 2013 - In a previous blog post of mine, I went through the steps of **decrypting**

# Man-in-the-middle attack

**Mitmproxy** is an interactive, SSL-capable man-in-the-middle proxy for HTTP with a **console interface** (mitmproxy.org).

Features:

- Intercept HTTP requests and responses and modify them on the fly.
- Make scripted changes to HTTP traffic using Python.
- Provide custom SSL certificates for interception.

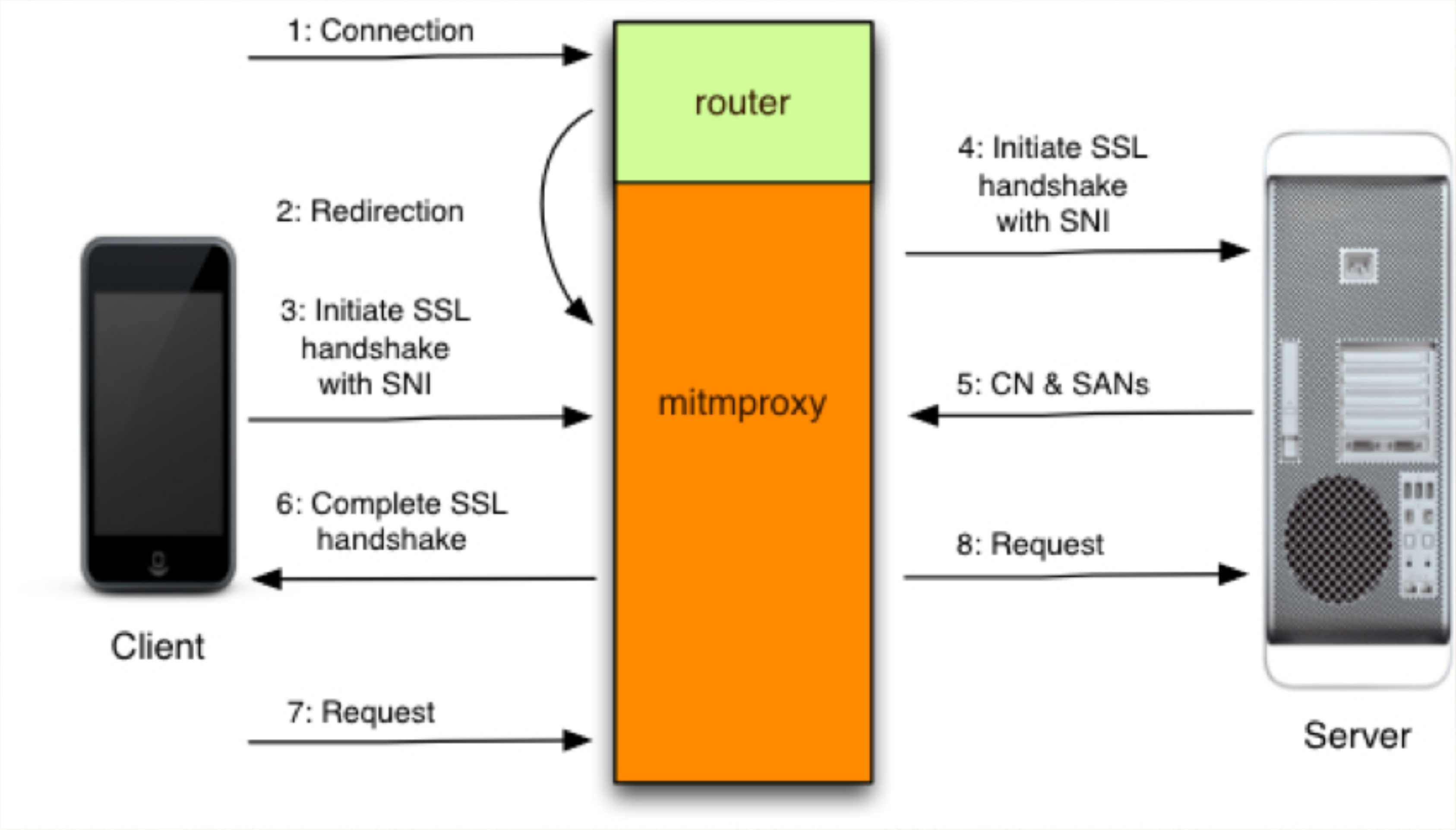
# How mitmproxy works

Conditions:

- Victim’s device is connected to malicious access point
- Attacker’s CA is registered as a trusted CA within the victim’s device

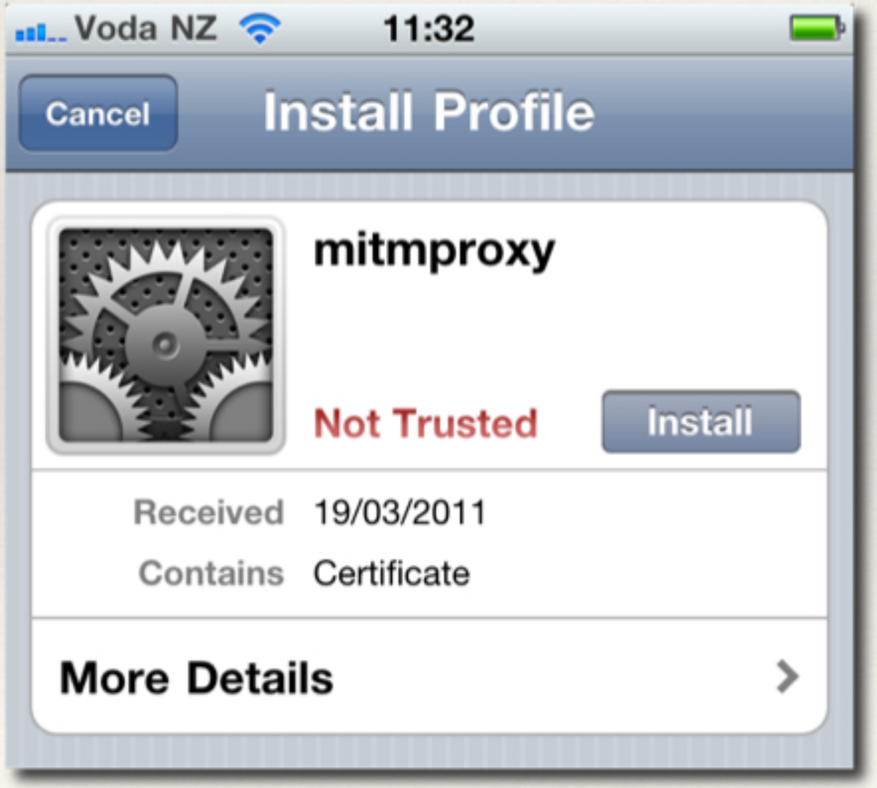
The first time mitmproxy is run, a set of certificate files for the mitmproxy CA are created in the config directory. This CA is used for on-the-fly generation of dummy certificates for SSL interception.

# How mitmproxy works


1: Connection
router
2: Redirection
3: Initiate SSL handshake with SNI
mitmproxy
4: Initiate SSL handshake with SNI
5: CN & SANs
6: Complete SSL handshake
8: Request
Client
7: Request
Server

# iOS. Getting the certificate onto the device

The easiest way to accomplish this is to set up the Mail app on the device, and to email it over as an attachment. After tapping on the attachment, user will be prompted to install a profile.



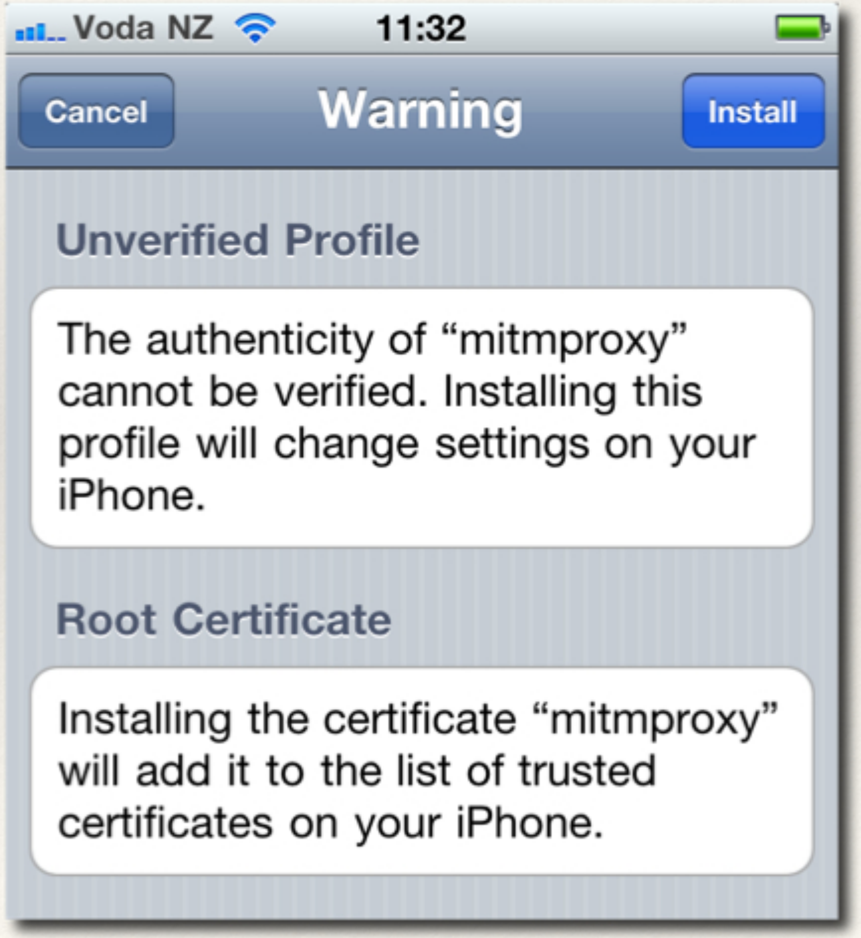
Install Profile

mitmproxy

Not Trusted

Received 19/03/2011

Contains Certificate



Warning

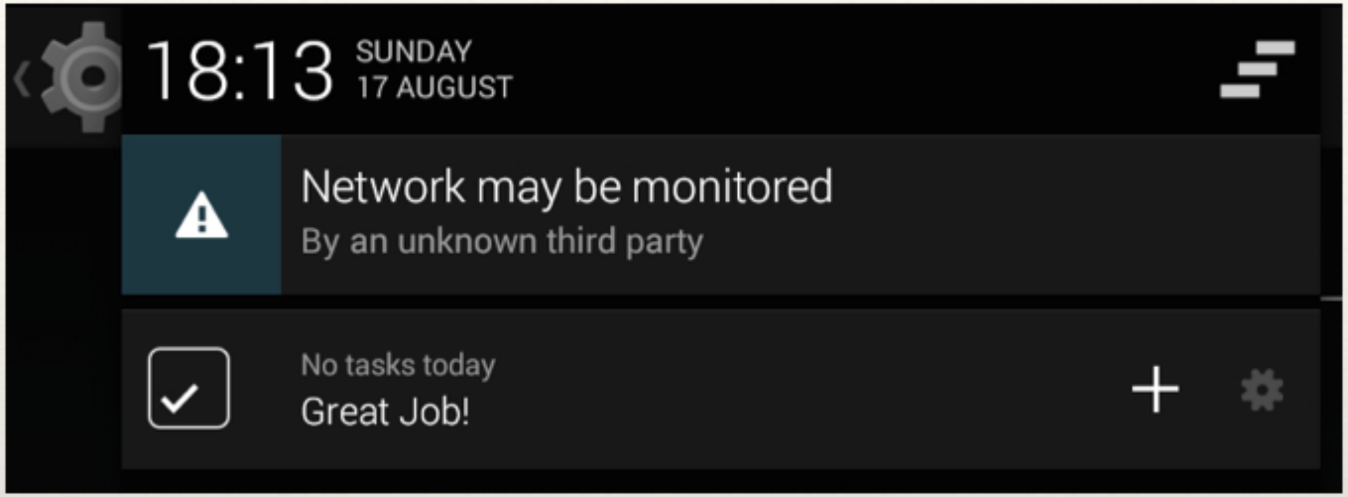
Unverified Profile

The authenticity of “mitmproxy” cannot be verified. Installing this profile will change settings on your iPhone.

Root Certificate

Installing the certificate “mitmproxy” will add it to the list of trusted certificates on your iPhone.

# Android 4.4.4 SlimRom 7.0 reaction



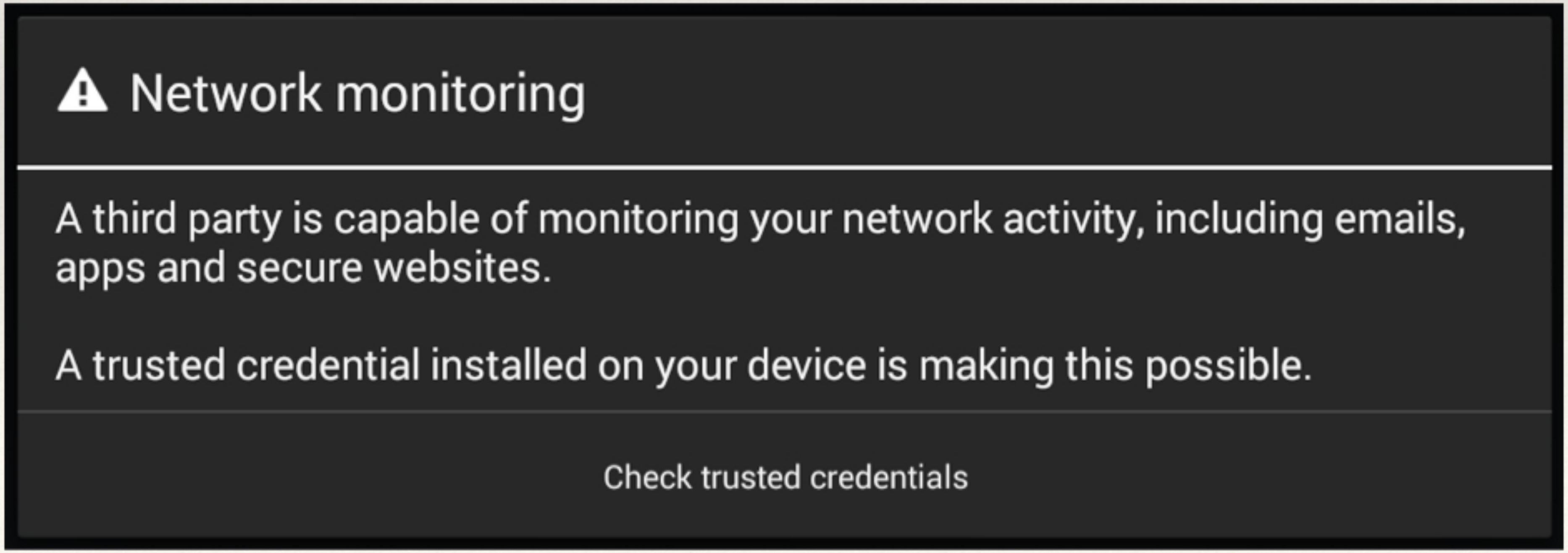
18:13 SUNDAY 17 AUGUST

Network may be monitored

By an unknown third party

No tasks today

Great Job!



⚠ Network monitoring

A third party is capable of monitoring your network activity, including emails, apps and secure websites.

A trusted credential installed on your device is making this possible.

# So far so good

- Now we can see traffic of almost all of our apps
- But essential condition of this MITM attack was presence of attacker's CA on victim's device, which may seem hard to achieve
- Maybe we can find some workaround just by crafting special certificates

# Types of certificates to try

- Signed by a trusted CA, correct CN
- Self-signed certificate
- Signed by an untrusted CA
- Wrong CN
- Parent domain’s CN
- Null character in CN
- SANsubjectAltNa Hostname is a DNSName in subjectAltName and in subject
- Hostname only a DNSName in subjectAltName
- Hostname in neither subjectAltName nor subject
- Hostname in subject but not in subjectAltName
- ExtendedKeyUsage is not critical, but lacks serverAuth
- ExtendedKeyUsage is critical, but lacks serverAuth
- Signed by an untrusted CA (provided in the chain) with the same name but a different key
- Bad signature
- Intermediate certificate where BasicConstraints sets CA:FALSE
- Intermediate certificate lacks BasicConstraints
- Intermediate certificate has bad signature from CA
- Signed with MD4/MD5
- Certificate that has expired
- Certificate that is valid in the future

# Vulnerable software*

- Amazon’s EC2 Java library and all cloud clients based on it;
- Amazon’s and PayPal’s merchant SDKs responsible for transmitting payment details from e-commerce sites to payment gateways; integrated shopping carts such as osCommerce, ZenCart, Ubercart, and PrestaShop;
- AdMob code used by mobile websites;
- Chase mobile banking and several other Android apps and libraries;
- Java Web-services middleware—including Apache Axis, Axis 2, Codehaus XFire, and Pusher library for Android;
- and many others

*Taken from “The Most Dangerous Code in the World: Validating SSL Certificates in Non-Browser Software”, M. Georgiev, et.al., 2012

# Vulnerable software

- Any.DO mobile app for Android accepts absolutely ANY certificate, attacker can see all user's notes in plaintext
- Some banking apps
- Analytics providers
- and more

# Common weaknesses

- Valid certificate for any host will be accepted
- Valid certificate for the targeted domain, except that its extendedKeyUsage states it is for some other purpose, such as code signing, non-repudiation, or OCSP signing, will be accepted

# Causes

- The root causes of these vulnerabilities are badly designed APIs of SSL implementations and data-transport libraries (such as cURL) which present developers with a confusing array of settings and options.

# Conclusion

- A lot of popular mobile apps are validating SSL certificates incorrectly or don’t validate them at all. It concerns not only small third party apps, but big services like Facebook, Google and others.
- Developers should pay more attention to privacy issues
- Automated checks for application SSL configuration and utilisation will be very helpful ("formal verification techniques")

# Questions