Why Eve and Mallory Love Android
An Analysis of Android SSL (In)Security

Sascha Fahl
Marian Harbach
Thomas Muders
Lars Baumgärtner
Bernd Freisleben
Matthew Smith
Some Android Facts

- 330 million devices (as of Q1 2012)
- 930,000 activations per day (as of Q1 2012)
- 450,000 apps (as of June 2012)

Market Share (Q2 2012)

- Android: 67%
- iOS
- RIM
- Symbian
- Windows Phone
Appification

- There’s an App for Everything
What do Most Apps Have in Common?

They share data over the Internet

Some of them secure transfer using:

SSL (Secure Sockets Layer protocol)

(TLS) protocol
SSL Usage on Android

The default Android API implements correct certificate validation.

What could possibly go wrong?
SSL Usage on Android

- A server needs a certificate that was signed by a trusted Certificate Authority (~130 pre-installed CAs)
SSL Usage on Android

- A server needs a certificate that was signed by a trusted Certificate Authority (~130 pre-installed CAs)
- Some are quite strange...
SSL Usage on Android

- A server needs a certificate that was signed by a trusted Certificate Authority (~130 pre-installed CAs)
- For non-trusted certificates a custom workaround is needed
What about using a non-trusted certificate?

Q: Does anyone know how to accept a self signed cert in Java on the Android? A code sample would be perfect.
A: Use the EasyX509TrustManager library hosted on code.google.com.

Q: I am getting an error of „javax.net.ssl.SSLException: Not trusted server certificate“. I want to simply allow any certificate to work, regardless whether it is or is not in the Android key chain. I have spent 40 hours researching and trying to figure out a workaround for this issue.

Leibniz Universität Hannover, Matthew Smith, 20.02.2013
Our Analysis

- downloaded 13,500 popular and free Apps from Google’s Play Market
- built MalloDroid which is an androguard extension to analyze possible SSL problems in Android Apps
  - broken TrustManager implementations
  - accept all Hostnames
Static Code Analysis Results

- 92.8% Apps use INTERNET permission
- 91.7% of networking API calls HTTP(S) related
- 0.8% exclusively HTTPS URL
- 46.2% mix HTTP and HTTPS
- 17.28% of all Apps that use HTTPS include code that fails in SSL certificate validation
  - 1070 include critical code
  - 790 accept all certificates
  - 284 accept all hostnames
Tusting all Certificates

- Correct SSL certificate validation is so easy
  - Only a (costly) trusted CA signed certificate required
- What some Apps do:

```java
// Create a trust manager that does not validate certificate chains
TrustManager[] trustAllCerts = new TrustManager[] { new X509TrustManager()
{

  public java.security.cert.X509Certificate[] getAcceptedIssuers()
  { return null;


  public void checkClientTrusted(X509Certificate[] chain, String authType) throws CertificateException
  { // do nothing

  public void checkServerTrusted(X509Certificate[] chain, String authType) throws CertificateException
  { // do nothing

};
```
Allowing all Hostnames

- What other Apps do:
  - Check CA signature, but allow mallory.com for google.com

```java
KeyStore trustStore = KeyStore.getInstance(KeyStore.getDefaultType());
trustStore.load(null, null);

SSLSocketFactory sf = new MySSLSocketFactory(trustStore);
sf.setHostnameVerifier(SSLSocketFactory.ALLOW_ALL_HOSTNAME_VERIFIER);
```
TrustManager Implementations

- 22 different TrustManager implementations

- and all turn effective certificate validation off
Manual App Testing Results

- cherry-picked 100 Apps
- 21 Apps trust all certificates
- 20 Apps accept all hostnames

What we found:
Manual App Testing Results

39 – 185 million affected installs!

What we found:
BankDroid

- Swedish banking app
- Support for ~60 banks/payment services
  - PayPal
  - Steam Wallet
  - Eurocard
  - Swedbank
  - …
BankDroid - Aftermath

- 26 out of 41 broken
- Deliberately broken
- NO user warning
Anti-Virus

Zoner AV

- Anti-Virus App for Android
- Awarded best free Anti-Virus App for Android by av-test.org
Zoner AV

- Virus signature updates via HTTPS GET
- The good thing: It uses SSL
  - Unfortunately: The wrong way
- Does not check the update’s authenticity!

```java
static final HostnameVerifier DO_NOT_VERIFY = new HostnameVerifier() {
    @Override
    public boolean verify(String host, SSLSession session) {
        return false;
    }
};
```
Zoner AV

- We did the following
More Examples

- Remote Control App
- Remote Code Injection
- Unlocking Rental Cars
How Do (Good) Apps React to MITMAs?

- Technically ✔
- Usability ❓

Flickr

![Sign In Error](image)

Oops, a little hiccup here. Please adjust the time on your device to the current time.

OK

Facebook

![Login Failed](image)

Sorry, login Failed to reach Facebook servers. Please check your network connection or try again later.

- hostname in certificate didn't match: <api.facebook.com> != <*.mallory.com> [javax.net.ssl.SSLEException]
Browser Warning Messages

All do SSL certificate validation correctly...

... and warn the user if something goes wrong....
SSL Warning Messages – Android Stock Browser

•

<table>
<thead>
<tr>
<th>Go back</th>
<th>View certificate</th>
<th>Continue</th>
</tr>
</thead>
<tbody>
<tr>
<td>YouTube</td>
<td>Google Voice</td>
<td>Kalender</td>
</tr>
<tr>
<td>Reader</td>
<td>Google Earth</td>
<td>Docs</td>
</tr>
</tbody>
</table>
Online Survey

- To find out if the Browser’s warning messages help the users
- presented an SSL warning message
- To see if users know when they are surfing on an SSL protected website
- half of the participants HTTP
- half of the participants HTTPS
Online Survey - Results

• 745 participants

• 47.5% of non-IT experts believed they were using a secure Internet connection...although it was plain HTTP.

• ~50% had not seen an SSL warning message on their phone before.

• The risk users were warned against was rated with 2.86 (sd=.94) on a scale between 1 and 5

• Many participants stated they did not care about warning messages at all.
Our Recommendations

- Integrate SSL certificate validation testing into the development process
- Inform the user
  - `INTERNET_SSL` and `INTERNET_PLAIN` permission
  - global SSL warning message
- Move SSL handling to the OS
  - Developers should not have to write code to use SSL
  - SSL via config instead of code (still enough room for error)