PCI Standards and Mobile Acceptance

Troy Leach, CTO, PCI Security Standards Council
Nicholas J. Percoco, Senior Vice President, SpiderLabs, Trustwave
Mobile Commerce Risk & Security

Nicholas J. Percoco, Senior Vice President, SpiderLabs, Trustwave
Introduction
About Me

Started InfoSec Career in the 1990s

Formed SpiderLabs at Trustwave in 2005

Given talks all over the world
  • 7 times at DEF CON, 3 times at Black Hat, Briefings for DHS, US-CERT, and United State Secret Service

Research areas include Data Breaches, Malware, and Mobile Computing

Primary Author of the annual Trustwave Global Security Report

Find me on the Twitterverse... @c7five / @SpiderLabs
About Trustwave SpiderLabs®

Trustwave SpiderLabs uses real-world and innovative security research to improve Trustwave products, & provides unmatched expertise and intelligence to customers.

THREATS

- Real-World
- Discovered
- Learned

PROTECTIONS

- Customers
- Products
- Partners

Response and Investigation (R&I)
Analysis and Testing (A&T)
Research and Development (R&D)
Mobile Device Landscape
Mobile Device Landscape - Usage

There are over a half-billion devices on 3G/4G networks

- By 2020, there will be 10 billion devices

60% of all users carry their devices with them at ALL times

- For high-profile and business folks that is near 100%

A smartphone today has the same power as PC from 10 yrs. ago:

- Always-on network connectivity
- Location aware via GPS
Mobile Device Landscape – Interaction

Accessing sensitive information via smartphones is normal
• Users do not fear making purchases or financial transactions

Users trust a smartphone over a public computer or kiosk
• They rarely question their smartphone’s integrity

Limited “desktop real estate” mean sacrificing notices
• Security notices are often misleading and/or cryptic
Mobile Device Landscape - Security

There is no standard Security UI for Apps

- i.e., No OS-based “Lock” for “SSL established”

Zero functionality difference between an App that sends data in the clear vs. encrypted

- App developers want “working” apps
- Who ensures mobile commerce is secure?

OS level security problems often cascade to ALL Apps on device

- Can you decertify an App for a OS version?
- What’s the impact to mobile commerce?
### Mobile Device Landscape - Commerce

#### Two Primary Methods of Commerce

<table>
<thead>
<tr>
<th>App Based</th>
<th>Web-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>• User downloads an App</td>
<td>• User accesses shopping-carts via built-in browsers</td>
</tr>
<tr>
<td>• Creates profile</td>
<td>• Makes purchases just like in the traditional desktop world</td>
</tr>
<tr>
<td>• Stores Credit Card Number (client or server-side)</td>
<td></td>
</tr>
<tr>
<td>• User accesses App and makes purchases when needed</td>
<td></td>
</tr>
</tbody>
</table>
Mobile Attack Scenarios
Mobile Attack Scenarios

The market boils down to two platforms:

iOS
Developed by Apple and runs only on Apple hardware

Android
Developed by Google and runs on compatible hardware
Mobile Attack Scenarios – iOS Devices

Jailbreak Vulnerabilities

Issue: Attackers can access data on users devices.

- Virtual arms race between Apple and the Jailbreak community
- Often days after a new OS release, a Jailbreak is released
- Jailbreak is available for all iOS mobile devices
- Once Jailbroken, a malicious App could “hook” a commerce app and intercept the data before it is encrypted
- Drive-by malware installation feasible and practical
  – Trustwave SpiderLabs demonstrated this at ToorCon 12
Mobile Attack Scenarios – iOS Devices

SSL Man-in-the-Middle

**Issue:** Attackers can intercept sensitive data in transit.

- Discovered by Trustwave SpiderLabs (CVE-2011-0228)
- Flaw in iOS SSL library that failed to validate invalid SSL certs when chained to a valid CA
- Apple fixed the problem within 10 days of notification (fast == critical)
- Result: SSL communication from iOS devices <4.3.5 cannot be trusted; includes all Apps, Web-based, etc.

**Question:** What are mobile commerce developers doing about these scenarios?
Mobile Attack Scenarios – Android Devices

Market Malware

**Issue:** Attackers can plant malware on users devices.

- Google screens Apps submitted using “Bouncer”.
  - A submitted App will be available to download in minutes
  - SpiderLabs defeated Bouncer and presented findings at DEF CON 20.
  - Google also relies on “user feedback” to mark malicious Apps for removal
- Users can also easily install Apps from outside the Market
- Many different malware variants have been seen in the Market this year including Zeus, SpyEye, etc!
Current Attack Scenarios – Android Devices

Rootkits

**Issue:** Attackers can take over user’s device.

- Android is based upon the Linux Operating System
- A “rootkit” installs at the lowest level on the device (kernel)
- Coupled with a “root” compromise, a “rootkit” can take over all the mobile device’s functions
  - SpiderLabs demonstrated this at DEF CON 18
Focus Stealing Vulnerability

Issue: User are tricked into giving information to an attacker.

• A critical design flaw in the Android OS discovered by Trustwave SpiderLabs (TWSL2011-008)

• Allow any App to steal the focus from any other App running
  – Data Stealing Upon Entry
    • Credentials / Credit Card Numbers
  – SpiderLabs demonstrated this at DEF CON 19
Demos
Demo #1 – Mobile Malware
Demo #2 – FocusStealing
Mobile Commerce Attack Mitigations
Mobile Commerce Attack Mitigations

**Jailbreak Vulnerabilities**
Ensure users are not using vulnerable OS versions or that device is not already rooted when performing transactions.

**SSL Man-in-the-Middle**
Use of Certificate Pinning in App has proved successful. Ensures app is communicating with your backend.

**Market Malware & Rootkits**
Until Google does a better job at screen apps, this will continue to be a threat. No great Anti-Malware exists today for any mobile device.

**Focus Stealing**
No solutions exists. Google needs to fix this “feature”. Including an “active” feature in the transaction screen may help users identify this attack.
Conclusions
Conclusions and Recommendations

Mobile Commerce is still in its infancy when it comes to ensuring the integrity of transactions.

The mobile platforms limit the user’s ability to ensure their transactions are, in fact, secure.

A “best practice” guide should be developed, by the industry, to educate mobile App developers on methods of securing commerce transactions and risks of not doing so.
PCI Standards and Mobile Acceptance

Troy Leach, CTO, PCI Security Standards Council
Agenda

The Council Perspective

Best Practices for Application Developers

Next Steps
Mobile is Changing Merchant & Consumer Behavior
75% new merchants using mobile

Billion devices

How many apps?

50% increase of merchant population
Mobile and Payments

How do we create integrity and confidentiality for payment transactions?

- Quickly changing environment
- Continuous accessibility and new attack vectors
- Additional dependencies
Mobile payments and the PCI Council

- Identified mobile applications that can be validated to PA-DSS
- Published merchant guidance for ‘mobile’ solutions leveraging P2PE
- Developed best practices for developers
- Next steps explored by PCI SSC
Differences of “Mobile”

Form Factor

Merchant Use

Solution Provider
Areas of Focus for Mobile

"MOBILE"

- **Devices**: Tamper-responsive, PTS Devices (e.g. SCR) using P2PE
- **Applications**: Requirements and/or Best Practices for authorization and settlement
- **Service Providers**: Service provider protection of cardholder data and validation
Why It Is Important to Get It Right: People

New group of merchants not aware of responsibility to protect cardholder data

New group of application developers providing access to cardholder data

New payment channel for administrators to monitor for abnormalities
Why Is Mobile Different: Process

May not use enterprise equipment

Process changes as “terminal” travels to various locations with persistent connectivity to various networks

Process to detect tampering and revoke card acceptance
**Why Is Mobile Different: Technology**

**Compared to other types of POS:**
- Lack of traditional controls

**Other entities with access to card processing environment**

Less responsive to evidence of tampering or privilege escalation

Challenges with Encrypting PIN Pad equivalent security for fallback acceptance
Agenda

The Council Perspective

Best Practices for Application Developers

Next Steps
Why guidance and not standards
Purpose of best practices
Purpose of best practices

Controls are broken into two categories:

Payment Transaction

Supporting Environment
Transactional controls

- CHD entering device: Prevent account data from being intercepted when entered into device
- CHD inside of device: Prevent account data from compromise while processed or stored within the mobile device
- CHD leaving device: Prevent account data from interception upon transmission out of the mobile device
Cardholder Data Entering the Mobile Device

Data should enter device via a secure (encrypted) channel

Communication is authenticated and authorized

Secured against client-side injections

PIN entry via PTS-approved devices such as Encrypting PIN Pad (EPP)
CHD Stored and Processed Within Device

Only processed inside of a trusted execution environment and not accessible outside of that trusted zone

Storage should be rendered unreadable per PCI DSS req 3.4

Temporary storage should be in a secure, isolated environment

Key management should prevent against exposure to unauthorized access

Should not retain Sensitive Authentication Data after authorization
Cardholder Data Exiting Mobile Device

Data that leaves the device should be properly encrypted

Prevent data leakage and false requests

Authenticate legitimacy of request to send CHD
Environmental controls

Unauthorized Access
- Logical device access
- End-user system access
- Escalation of privileges
- Applications
- Attachments

Remote management
- Detect theft/loss
- Disable payment application
- Trusted path for updates/patch management

Strong host-side controls compliant with PCI DSS

Indication operating in Secure state
Environmental controls (continued)

Secure Coding best practices
• Developers are trained on PCI requirements
• Implementation documented
• Vulnerability scanning
• Formal response plan to identify and mitigate new risk

Protect system from malware

Instructional material to be PCI DSS on implementation and execution

Merchant receipts
Additional considerations

Device Validation

Regional Jurisdiction

Technological Limitations
- Physical
- Data Accessibility
- Application Isolation

Indeterminable Risks
- Evolution of Technology and Unforeseen Attack Vectors
- Vulnerabilities Markets
- Intentionally Inserted Backdoors
What is next

Merchant-focused – how to leverage mobile acceptance

Continue to analyze validity of PA-DSS to validate mobile applications and other alternatives

Collaboration with mobile payment groups developing requirements for issuance

Continue to discuss issues with the Mobile Task Force and develop industry guidance
How to get involved: Mobile Task Force

Participating organizations and subject matter experts

Examples of subject matter experts:
- Security Assessors
- OS Platform Vendors
- Financial Processors
- Device Manufactures
Continued maturity in the space for traditional controls and security innovation

Need for secure development lifecycles for “agile” programming along with security awareness for developers and users

Address privileges that provide unauthorized access to financial applications and cardholder data

Need for mechanisms to demonstrate solutions are secure for merchants and consumers

Get involved
Questions?

Please visit our website at www.pcisecuritystandards.org