Creating Secure Mobile Applications
Illuminating Mobile Threats

OWASP Software Assurance Day DC 2009
Friday, 13 March

Jason Rouse
jrouse@cigital.com
Agenda

- Introduction
- Mobile Architectures
- Mobile Threat Model – Attacks and Defenses
- Wrap-Up & Discussion
The Scale of Things

- The Internet Is big.

- There are approximately 1,000,000,000 people on the internet.

- And there are approximately 3,000,000,000 mobile handsets in use.

- What sort of attack surface, computational power, and force multiplication do cell phones have?
Mobile Platforms are Fragmented

- **Nokia**
  - Symbian (J2ME, C/C++)
  - UIQ (J2ME, C/C++)
- **SonyEricsson** (J2ME, C/C++)
- **iPhone** (J2ME, Objective C)
- **RIM** (J2ME, C/C++)
- **Motorola** (J2ME, C/C++)
- **Google Android** (Java, C/C++)
Mobile Platforms are Fragmented

- This fragmentation leads to tiny “islands” of content, applications, and use cases
- These islands will begin to disappear as carriers, handset manufacturers, and framework providers come together to monetize cell phones
- Once these islands are gone, we’ve got the good, and we’ve got the bad.
Mobile Platforms are Standardized

- The Good:
  - 1-stop shopping for content and applications
  - Everyone’s smart phone works with everyone else
  - Content and application providers will have an easier time converging functionality onto mobile devices
Mobile Platforms are Standardized

- The Bad:
  - 1-stop shopping for content and applications
  - Everyone’s smart phone works with everyone else
  - Content and application providers will have an easier time converging functionality onto mobile devices
“Convergence is the Way To Go™”

- Convergence of functionality, and the requisite data onto mobile phones is only increasing
- Mobile phones are becoming interesting targets for attackers wishing to do more than just play with OS vulnerabilities
- Mobile phones could represent an incredible efficiency boost, or a horrible liability
What do you put on your phone?

- Phone numbers
- Call history
- Music?
- Location-Based Services (Google Maps, Google Latitude, VZNav, BB Maps)
- Photos
- Email
- ... VPN keys?
- ... Passwords?
There is no doubt in my mind that secure converged devices are the way to go....
“Convergence is the Way To Go™”

- ...but we’ve got a long way to go before we have truly secure mobile devices!
Mobile Application Architectures
Mobile Application Architectures

- Easily characterized by how much information is stored on handset.
- Generally dependent on liability, performance, scalability.
- Share more common traits than you think.
- Almost any application architecture can be transformed into another, given enough $$ and time.
Complex Payment Architecture

- Stores important information on the handset.
- Requires tight integration between MNO and FI
- Requires high trust between MNO and FI
- Burdens the handset with information protection requirements
- Device loss could become liability for consumer, MNO, or FI
- Any other issues?
Web Front-End

- Does not require storage of important information on the handset
- No integration between MNO and ASP – essentially turns MNO into a “plumber” providing pipes connecting mobile browser to ASP website
- Usually cost-effective, as ASP can leverage previous investments in web applications to onboard mobile devices
- Example: BoA Online Banking for Mobile
Mobile Services Client (Hybrid)
Mobile Services Client (Hybrid)

- May require storage of important information on the handset
- Little or no integration between MNO and ASP – however, MNO often controls some aspect of application loading, provisioning, and personalization
- Usually cost-effective, as ASP can leverage previous investments in web applications/services to on-board mobile devices
- Example: VzW Visual Voicemail
Mobile Threats – Attacks, Defenses, and Data
5 Main Areas | Resources and Practices
5 Main Areas

- Directed SMS
  - Application event drivers
- Debugging & Logging
  - Wildly variable implementation
- Error Handling
  - Failures & Recovery
- Architecture & Design
  - “remote control” to “full mobile application”
- Device Loss or Capture
  - Remote control of content
5 Main Areas | Resources and Practices

Directed SMS
Debugging & Logging
Error Handling
Architecture & Design
Device Loss or Capture
Directed SMS

Diagram:
- Application Crashes
  - Application Logic Failures
  - Malformed Inputs
  - Operating System Crashes
- Clear-Text Credentials
  - Originating MDN
  - Authentication
- Protected Credentials
- Application Management System
  - Application-Specific
  - Authorization
  - Application-Directed SMS
Directed SMS

- Messages drive many events for handset applications
- Often, these messages contain actionable data, from content IDs to IP addresses
- This input must be carefully screened for malicious content
- Information contained in these messages must be protected as well as information stored on a handset!
Directed SMS

- How often do we authenticate the sender or receiver of an SMS message?
- How can we authenticate such principals?
5 Main Areas | Resources and Practices

Directed SMS
Debugging & Logging
Error Handling
Architecture & Design
Device Loss or Capture
Debugging & Logging

- Near & Dear to my heart
- Incredibly valuable to:
  - Programmers
  - Attackers
- Not so directly valuable to:
  - Users

- Let’s look at the topics separately
Debugging

- Need to know what to record and what not to record.
- Need to take into consideration where you’re storing this information
- Need to consider performance hits
- Need to consider remote-control ability for debug logs and troubleshooting
Logging

- Encryption
- Obfuscation
- Log Protection
- Protected Locations
- On-Device
- Log Output
- Velocity Checking
- Caching of Messages
- Performance Management
- Batch Message Writes
- Device Protection
- Size Limitations
- Alerting Functionality
- Endpoint Authentication
- Channel Protection
- Network-Based
- Hybrid Device/Network
Logging

- Very different from debugging – logs could conceivably stay on during normal deployments, and might even form a part of the application’s data model.

- Still have some of the same issues – what to log, how to log it, where to log it, etc…
5 Main Areas | Resources and Practices

Directed SMS
Debugging & Logging
Error Handling
Architecture & Design
Device Loss or Capture
Error Handling

- E911 LBS
  - Life Safety
  - LBS
  - Simple Functionality
  - Fail Open
  - Non-Billable Functionality
  - Information Protection
  - Billable Functionality
  - WDN Protection
  - Fail Closed
  - Error Handling
  - Premium Services
    - Device Theft or Loss
    - Subscriber Protection
  - Error Information Masking

Monday, March 23, 2009
Error Handling

- Error handling can be a make-or-break aspect of many mobile applications.
- Error handling can release protected content (fail open)
- Error handling can cause lost revenue when, for instance, an application uninstall is interrupted but the billing information is erased
- Error handling can even affect life safety, if we look at E911 services
Error Handling

- The biggest question to ask yourself is: Fail Open, or Fail Closed?
- The answer to this question will dictate any and all controls you must put in place downstream.
5 Main Areas | Resources and Practices

- Directed SMS
- Debugging & Logging
- Error Handling
- Architecture & Design
- Device Loss or Capture
The architecture can drastically affect where we store and process information. This means that we have to be cognizant of a number of areas, including:

- Authentication Tokens
- Information Leakage
- Content Protection
Authentication Tokens

- Session Identifiers
  - Service APIs
  - Web Service Sessions
  - Third-Party Protocols
  - Application PIN
    - Activation
    - Logon

- Location-Based Services
  - VoIP SIP Clients
  - IMAP Sessions

- Application-Directed Messaging
  - Session Identifiers
  - Origin Authentication
  - Application Passwords

© 2008 Cigital Inc. All Rights Reserved. Confidential.
Authentication Tokens

- Auth tokens are the holy grail of attackers
- If they can be stolen, predicted, fixed, or obviated, then we have lost, and the attacker has won
- The key issue here is to be aware of the tokens you use, how long you use them, and how they are disposed of!
Information Leakage

- Personally Identifiable Information
  - MDN
  - EIN
  - Phone Book Entries
  - Pictures
  - Registration Information
  - Location Fixes

- Configuration Information
  - URLs
  - IP Addresses
  - Quotas
  - Licensing Restrictions
  - Trial Limitations

- Environment and Executable
  - RMS
  - Persistent Storage
  - Registry
  - Filesystem
  - Resource Files
  - Application Object Files
  - Device Binaries
  - Key Stores
  - License Objects
Information Leakage

- We see many familiar things here – Personally Identifiable Information, like MDN, phonebook entries, LBS fixes...
- All of this is a potential customer-affecting issue!
- Information leakage must be curtailed during the architecture phase and managed with strict controls in deployment
- Handsets have a rich storage capacity in multiple formats and multiple transfer capabilities
Information Leakage

- We often forget, as developers, just how much information we leave on handsets!
- Debug PINs
- URLs
- Error Strings
- Authentication Clues
Content Protection

- Content Protection is an easy to understand issue on today’s networks: carriers seek to monetize content and its delivery
- Content protection can run the gamut from encrypted files with a robust key-management scheme to a simple “stream-on-demand” model that seeks to prevent content from existing on the handset for too long
- Some vendors are even pursuing watermarking of content as a deterrent
5 Main Areas | Resources and Practices

Directed SMS
Debugging & Logging
Error Handling
Architecture & Design
Device Loss or Capture
Device Loss or Capture

Device Loss
  - Content License
    - Remote Wipe
      - Device-Mediated
        - Handset Access PIN
        - OS Remote Wipe Capability
      - Network-Mediated
        - Native Application Support
        - Handset-Supported Application-Supported
    - Recovery
      - On Device
      - On Network
    - Revocation
      - On Device
      - On Network
  - Content Recovery
    - Side-Loading Provisions
      - Network Loading Provisions
Remote Wipe

- Often times it’s easiest to classify this functionality as “network” or “device” mediated.
- If the carrier/MNO can remotely wipe a device, there is a good amount of protection.
- If a local application, however, is able to wipe the device by using a dead-man’s switch, then this could catch criminals off-guard.
- True or False: There is rarely a need in consumer goods for robust network or device remote wipe!
When a device is lost, it is as important to recover a customer’s licenses as it is to recover their content. If those licenses cannot be recovered, then the device should support some form of revocation, to protect both the customer and the content owner from fraudulent uses of their data.
The biggest problem with content recovery is: where do I get my content from? Most mobile applications can reconstruct or restore a handset’s state by re-personalizing or re-provisioning a handset.

When we have hundreds of megabytes or more, however, things get complicated.

Side-loading is by far the easiest method to off-load the network, but it may cause headaches with OS support, client issues, etc...
Wrap-Up
Wrap-Up

- We’ve covered a lot of ground: mobile architectures, mobile threats.

- Take a moment to digest, and let’s talk about some of the relationships between these elements and any other questions we might have.
Discussion & Question Period