Exposing the BlackBerry Boogey Man
Agenda

- Introduction/Background
- Enterprise Architecture
- Device Architecture
- Hardware Overview
- Cryptographic Overview
- Attack Surfaces/Methodologies
- Forensics/Anti-Forensics
- Enterprise Security
Who are you, and what are you doing here?

• **SRA**
  – Leading provider of technology and strategic consulting services and solutions - including systems design, development and integration; and outsourcing and managed services.
  – Comprehensive cyber security practice integrating security architecture, risk assessments, and certification & accreditation. SRA’s IA practice currently rated at NSA-CMM Level 3.

• **Adam**
  – Security Consultant
  – Penetration Test Team
  – Forensic Technician
  – Security Architect
  – Reverse Code Analysis
• Fall 2008 a promise is made
• Meet JK Benites
• This ‘genius’ left his name in the malware he wrote to steal banking credentials and ended up at a certain US Government Agency

i'm JK Benites.
I like the music, i love the rock N metal, i'm a person that like stranges things, like adredaline, he good with friends, make new things... i play the guitar, my guitar is my life, with she i can show that i feel.
i like the Pcs, too.
...
Visit my profil in Hi5: http://jkprotection.hi5.com
City: Piura
Hometown: Piura
BlackBerry Boogey Man

• “If you go to country X with a blackberry you are immediately owned”
• “We can decrypt in memory data with no password”
• “If you have a blackberry I can arbitrarily turn on the mic and listen to you”
• “Exploiting the blackberry is trivial”
• Remote Exploitation is theoretically possible, though no publicly available examples are available – therefore probably not ‘trivial’
• Remote Access Toolkits do exist
  – Tyler Shields – txsbbspy (publicly released code)
  – Flexispy (commercial)
  – Mobile Spy (commercial)
• Blackberry targeting has occurred
  – Etisalat
• The GoI, GoSA, GoX are trying to force RIM to provide monitoring capabilities
Caveats

- The scope of this presentation is Blackberry Enterprise devices and architecture – not blackberry stand alone devices (e.g. personal cell phones)
- This is all based on and using publicly available and unclassified information
Acronyms

- BES – BlackBerry Enterprise Server
- MDS – Mobile Data Service
- MVS – Mobile Voice System
- JVM – Java Virtual Machine
Agenda

- Introduction/Background
- Enterprise Architecture
- Device Architecture
- Hardware Overview
- Cryptographic Overview
- Attack Surfaces/Methodologies
- Forensics/Anti-Forensics
- Enterprise Security
BlackBerry Enterprise Architecture

Attachment Server

BES

Other BB Services
BlackBerry Enterprise Components

- Attachment Service – Converts Attachments for handheld
- Collaboration Service – Integrate corporate IM (e.g. Microsoft Communicator)
- Configuration Database – Database for BES component configuration
- Controller – Monitor all components health
- Dispatcher – Compress and encrypt data for handheld
- Manager – Administration Console
- MDS Repository – Stores MDS runtime applications
- MDS Connection Service – Bridge handheld to intranet for application/browsing
- MDS Services – Manage MDS integration and device access/installation
- Messaging Agent – Interconnect to mail/calendar/contacts
- Policy Service – Manage IT Policies on devices and provisioning
- Router – Connect services to wireless network
- Synchronization Service – Organize data between devices and messaging server
Enterprise Components

• Management
  – controls mobile devices and can enforce enterprise policies regarding usage, third party software, bluetooth, password, etc.
  – Database (MSSQL) of devices, users, etc

• Components
  – Interconnect into Microsoft Exchange Server
  – Best Practice is to use a separate system
  – Interconnect between blackberry device and existing enterprise applications and systems
• Introduction/Background
• Enterprise Architecture
• Device Architecture
• Hardware Overview
• Cryptographic Overview
• Attack Surfaces/Methodologies
• Forensics/Anti-Forensics
• Enterprise Security
Blackberry API Development

- JavaME Applications
- MDS Applications
- Browser Applications
Architecture

- MDS Runtime
- JavaME VM
- Kernel
Java Virtual Machine

- **JavaME VM**
  - Connection Limited Device Configuration (CLDC)
  - Mobile Information Device Profile (MIDP)
  - Bberry API

- **JVM Responsibility**
  - Inter-Process Communications
  - Memory Management (Garbage Collection)
  - Process Management
  - Exception Handling
Agenda

- Introduction/Background
- Enterprise Architecture
- Device Architecture
- Hardware Overview
- Cryptographic Overview
- Attack Surfaces/Methodologies
- Forensics/Anti-Forensics
- Enterprise Security
Joint Test Action Group (JTAG)
Qualcomm RTR6350
Dual mode GPS/Diversity

JTAG??

8830 PCB Back
Maxim Stereo Codec
Marvell PXA90
Marvell PMA800 Croma Power Amplifier
AWT6168 GSM/GPRS
TPS65800 Power Management
Winning with JTAG

• JTAG can provide access to bootloader
• JTAG can allow memory to be access from NOR/NAND
• JTAG may provide access to memory artifacts and other interesting things such as password hash
Agenda

• Introduction/Background
• Enterprise Architecture
• Device Architecture
• Hardware Overview
• Cryptographic Overview
• Attack Surfaces/Methodologies
• Forensics/Anti-Forensics
• Enterprise Security
Boot Process

- BootROM code stored in flash memory
- BootROM is signed by RIM
- Processor can verify BootROM signature
  1) Processor runs ROM check code
  2) If ROM matches RSA public key then boot occurs if not then processor stops running
1) User enters password
2) Device derives ephemeral AES 256-Bit Key
3) Device uses ephemeral key to decrypt the content protection key and ECC private key
4) Decrypted keys moved to RAM
5) After 128 ECC encrypted items are accessed the device re-encrypts them with content protection key
Additional Security Features

- Message security
- Secure Memory Wipe
  - Default uses standard java garbage collection
  - Secure overwrites memory with 0x00 after collection
Agenda

• Introduction/Background
• Enterprise Architecture
• Device Architecture
• Hardware Overview
• Cryptographic Overview
• Attack Surfaces/Methodologies
• Forensics/Anti-Forensics
• Enterprise Security
Cellular Phone Exploitation

• Variety of Operating Systems (OS)
  – Symbian
  – Iphone
  – Blackberry
  – Android
  – Others

• Symbian and iPhone have been actively exploited
Attack Options

- BlackBerry Physical Access
- Cellular Communications Channels
- Browser
- Operating System
- MDS Runtime
- Social Engineering
Physical Access

- **Device Memory**
  - Content protection (if enabled) uses 256-bit AES
    - Created using NIST approved PRNG
    - Derives key from password in accordance with PKCS #5
  - New data received while locked is encrypted using ECC public key of variable size
  - OS and non-protected data may be available

- **External Storage (Media Card)**
  - Blackberry supports external encryption
Device Forensics

Application Databases (contacts, mail, etc)

Digital Storage (pictures, video, audio)
Cellular Phone Attack Surface

Attack Command Set (AT commands)

Receive controls from Carrier

Encode Voice Communication (GSM/CDMA/etc)

Interception/Man-in-the-Middle
Blackberry Phone Attack Surface

- Attack Command Set (AT commands)
- Receive controls from Carrier
- Encode Voice Communication (GSM/CDMA/etc)
- Interception/Man-in-the-Middle
Browser

- Browser on mobile devices in general are subject to various attacks
  - Apple Safari PDF vulnerability = Jail Breaking
- Browsers can be targeted using social engineering for XSS attacks or remote exploitation
- Blackberry browser sucks (sorry)
- JavaScript does run on Browser so typical attacks using js can work
- Browser attack surface is very attractive for OS attack and JavaScript attacks targeting information, XSS
• Blackberry OS is Java based, uses database to store application information and data

• Sensitive OS API are restricted by signed certificate to use
  – Blackberry Runtime API
  – Cryptographic API (includes certicom)
  – Blackberry Application API

• Certificate can be acquired at: https://www.blackberry.com/SignedKeys/

• 20 dollars
OS Permissions

• OS maintains permissions for all non-rim applications
• Permissions are broken into three categories
  – Connections
  – Interactions
  – User Data
• Permissions are Allow, or Deny
• New applications can be loaded via USB using Blackberry Desktop or OTA (Over The Air)
• Trusted Access?
• Default permissions - Tyler Shields - Veracode
• Its Java
• OS handles Garbage Collection
• OS is optimized for speed and developers are encouraged to design using similar techniques
• Exposed API is limited
MDS Runtime

- Not installed by default
- Not very common
- Phasing out MDS on the device
Agenda

- Introduction/Background
- Enterprise Architecture
- Device Architecture
- Hardware Overview
- Cryptographic Overview
- Attack Surfaces/Methodologies
- Forensics/Anti-Forensics
- Enterprise Security
Forensics Tools

- Paraben (Commercial)
- Amber BlackBerry Converter (Freely available)
- BlackBerry Desktop Manager
- Java_Loader.exe
- Others?
Wiping a Device

- Wiping does not remove 3rd party or additional software ‘cod’ files
- Reflashing the OS provides complete OS reload
- Wiping overwrites data stores with ‘0x00’
Agenda

- Introduction/Background
- Enterprise Architecture
- Device Architecture
- Hardware Overview
- Cryptographic Overview
- Attack Surfaces/Methodologies
- Forensics/Anti-Forensics
- Enterprise Security
• Blackberry Enterprise pushes “IT Policy”
• Template (think GPO) for device security
  – set default application permissions
  – set password enforcement
  – network settings
Blackberry is extremely well architected to be secure
Malicious tools do exist for remote access
Social Engineering provides possible attack vector
Exploitation is not easily accomplished
If you have an AES decoder ring then nothing can be done
Blackberry Enterprise Architecture needs to be carefully monitored and protected or all bets are off
Best Practices

- Secure and Enclave Servers
- Ensure Content Protection is enabled
- Generate Strong IT Policy
- Change the SQL password on the server
- Security Technical Implementation Guides
Sources

- http://www.blackberry.com/solutions/resources/Placing_the_BlackBerry_Enterprise_Solution_in_a_Segmented_Network.pdf