A Forensic First Look at a Point of Sale Device

CONFIRMING PCI DSS COMPLIANCE
Authors

• Stephen Larson
  • Associate Professor
  • Information Systems, Slippery Rock University of PA
  • stephen.Larson@sru.edu

• Jim Jones, Associate Professor
  • Computer Forensics Program
  • George Mason University
  • jjonesu@gmu.edu

• Jim Swauger
  • Partner, Binary Intelligence
  • jswauger@binaryintel.com
Agenda

• Intro
  • PCI DSS
  • POS breaches
  • Motivation

• Acquisition

• Analysis

• Results/Discussion/Conclusion
Payment Card Industry Data Security Standards (PCI DSS)

• The main goal of PCI DSS is to protect cardholder data using the PCI security standards.

• 12 requirements

• These requirements/standards “apply to all entities that store, process, and/or transmit cardholder data. It covers technical and operational system components included in or connected to cardholder data”.

• PCI DSS requirement 3
  • Concerns protecting stored cardholder data. “Cardholder data refers to any information printed, processed, transmitted or stored in any form on a payment card”
PCI DSS (continued)

• Guidelines to protect cardholder data include:
  • Store data only as long as necessary for business
  • Sensitive authentication data should not be stored after authorization (even if it is encrypted)
  • Only display either the first six or last four digits of the primary account number (PAN).
  • Encrypt the PAN when stored
  • Clearly document procedures to protect encryption keys
  • Implement procedures to protect encryption keys
PCI DSS (continued)

• Storage of certain data elements on a payment card is permitted
  • The primary account number (PAN) can be stored with no encryption.
  • The cardholder name, service code, and expiration date can be stored but must be encrypted. The full track data (the data on the magnetic stripe), the 3-4 digit value (CAV, CVC, CVV, or CID) value, and the PIN cannot be stored
PCI DSS (continued)

- PCI DSS Quick Reference Guide: Protect Cardholder Data (PCI 2018, 14-15)

- Requirement 3: Protect stored cardholder data

- Cardholder data refers to any information printed, processed, transmitted or stored in any form on a payment card. Entities accepting payment cards are expected to protect cardholder data and to prevent its unauthorized use – whether the data is printed or stored locally, or transmitted over an internal or public network to a remote server or service provider.
<table>
<thead>
<tr>
<th>Data Element</th>
<th>Storage Permitted</th>
<th>Render Stored Data Unreadable per Requirement 3.4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cardholder Data</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Primary Account Number (PAN)</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Cardholder Name</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Service Code</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Expiration Date</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td><strong>Sensitive Authentication Data</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Full Track Data</td>
<td>No</td>
<td>Cannot store per Requirement 3.2</td>
</tr>
<tr>
<td>CAV2/CVC2/CVV2/CID</td>
<td>No</td>
<td>Cannot store per Requirement 3.2</td>
</tr>
<tr>
<td>PIN/PIN Block</td>
<td>No</td>
<td>Cannot store per Requirement 3.2</td>
</tr>
</tbody>
</table>

1. Sensitive authentication data must not be stored after authorization (even if encrypted).
2. Full track data from the magnetic stripe, equivalent data on the chip, or elsewhere.
3. The three- or four-digit value printed on the front or back of a payment card.
4. Personal Identification Number entered by cardholder during a transaction, and/or encrypted PIN block present within the transaction message.
POS Breaches

• Restaurants: Applebee’s, Arby’s, Sonic, Chili’s, Zippys’, Subway, etc.
• Health care: Oregon clinic, etc.
• POS Vendor: Harbortouch (4200 customers), DataPoint POS, etc.
  • https://krebsonsecurity.com/2015/05/harbortouch-is-latest-pos-vendor-breach/
  • https://krebsonsecurity.com/2017/03/google-points-to-another-pos-vendor-breach/
• Hospitality: Holiday Inn, Hyatt, etc.
• Other: Heartland Payment Systems, Goodyear, AZ utility POS, Equifax, Forever 21, Target, Home Depot, etc.
POS breaches in 2018

• 321 POS devices
• 320 POS controllers (server)

• Many PoS terminals are built using embedded versions of Microsoft™ Windows®.

Motivation

• At restaurant, payment card connection failed, receipt printed anyway

• Called Heartland Payment Systems
  • No luck

• Called Verizon support
  • Transactions are cached
  • Then batch processed later that night

• In other words, data is on terminal/device for possibly hours
  • What if the device gets stolen?
Acquisition of POS Devices

• Bought 3 devices off craigslist.org
  • Verizon Omni Vx610 x 2 and Vx570 x 1
  • 2 different owners

• One owner was a caterer (food truck)
  • Said “Never connected to process transactions until evening”
Data Acquisition/Analysis

• Printed device information report
  • About 5 feet of “receipt” paper
  • Device info
    • Date/time
    • Owner information
    • List of all files (including config files)
      • Whether they reside in RAM or in Flash memory
      • Which of the 15 application groups they belong to (VMACCS, SPC, blank)
    • RAM, etc.
    • Ended with a summary and whether the device supports wifi, Ethernet, etc.)
Data Acquisition

• Connected to on-board ports (USB, RS232, 10BT, keyboard)
  • Unsuccessful

• Tried JTAG
  • Unsuccessful

• Sent to Binary Intelligence for chip-off data extraction
Chip-Off Extraction

Step 1 – the memory chip is physically removed. This is accomplished using appropriate heat (de-soldering) and chemicals (adhesive removal).

Step 2 – the chip is cleaned and repaired (or re-balled) as necessary.

Step 3 – the raw data is acquired or “imaged” from the chip using specialized chip programmers and adapters.

Step 4 – the raw forensic image is then analyzed using industry standard forensic tools and custom utilities.
Vx610 4MB Flash Memory Chip
Chip-off

- GL032N90FFI02 chips removed using a Zephyrtronics ZT-7 hot air reflow unit
- no visible chip underfill or coating so no adhesive removal necessary

1. The Printed Circuit Board Assembly (PCBA) was mounted to the work cradle
2. The PCBA was preheated using a Zephyrtronics AirBath digital preheating unit @ 205°C
3. RMA-223 flux paste was applied around and under the target chip
4. The ZT-7 reflow unit nozzle was positioned over the target chip
5. The ZT-7 reflow unit was engaged to ramp to a setpoint of 285°C
6. The ZT-7 vacuum probe was activated for automatic chip removal
7. The target chip automatically lifted when the ZT-7 reflow unit nozzle temperature reached approximately 280°C
8. After cooling, the removed chip was cleaned using 99% isopropyl alcohol in an ultrasonic cleaner
Chip-off (continued)

• Each target chip was imaged using an UP&UP UP-828P universal chip programmer per the following steps:

1. An EBGA64P chip package adapter was attached to the UP-828P programmer unit
2. The UP-828P control software version 2.0.4.8 was launched
3. The control software was configured to read the target chip by selecting the S29GL032N90FFI02 chip profile
4. The READ function was used to download the chip data into the control software buffer
5. The VERIFY function was used to re-read the chip data and compare it to the buffer contents
6. The verified chip data was saved to a binary image file
The Spansion GL032N90FFI02 chips were successfully imaged with no errors and the verification process yielded the exact same data. At 4,194,304 bytes, the file sizes of the saved binary images matched the expected size as detailed in the chip manufacture datasheet.

- MD5, SHA1, and SHA256 cryptographic hash values calculated
- performed a big/little endian conversion by reversing the byte order so the ASCII strings would be human readable
- The flash image size was 4,194,304 bytes (4 MB) for both units
Analysis

• The two flash images were examined in a hex editor for strings, similarities, and differences.
• The kernel is located in bytes 0-65535 (64 KB); this is the first 128 sectors assuming a 512-bytes sector size.
• Device serial numbers and kernel versions are near end of the kernel block (~bytes 65310-65527).
• The flash file system begins after the kernel, located at byte offsets 65536-573440 (sector 128-1119).
• All files found in flash memory except for wizard.ram
Interesting finds

• Multiple locations in image 1 and 2: "PREDATOR" (user? also in ped.lib)
• Sector 1400 (image 1) and 1322 (image 2): FTP user (anonymous) and password (guest@vfi.com);
• Sector 1773 (image 1) and 1695 (image 2): "Password:" and "66831" (an access code?)
• Variable "*GO" in configuration file sets the startup program; both set to F:APLOAD.OUT (but couldn't find this file on either flash image)
• Sector 1389 (image 1), 1311 (image 2):
• Banshee.Carlos Only.CO561 Only..Carlos+CO561....MC56 Only...MC55 Only...EM3420 Only.CO710 Only..Carlos+MC56.Carlos+MC55.Carlos+EM3420...Carlos+CO710....Eisenhower
The Good News

- The image files contained mostly compiled code & encrypted data
- Any clear text entries were error messages, informational messages, initialization messages, menu items, file names and locations, configuration settings, device information, etc.

- There was no clear text transaction detail information (such as card type, transaction amount, cardholder’s name, last 4 digits of card number, etc.) found in the flash memory image files.
Good News (continued)

• Most of the data is either binary code or encrypted data, and the clear text appears to contain only informational and error messages, configuration information, and some access information.

• IP information such as network configuration (DNS, DHCP, IP addresses, etc.) were in clear text, as was the FTP username and password.

• PCI DSS Compliance with requirement 3 was confirmed
Bad News

- Able to print last transaction
  - Device owner
  - Terminal ID #
  - Transaction date & time
  - Credit card type & last 4 digits
  - Transaction amount
  - Credit card owner name
  - Transaction approved/not approved
  - Batch #
Conclusion

• POS devices are:
  • PCI DSS requirement 3 compliant
  • USB port does not allow data access

• Your info is safe on the device
  • Unless yours is the last transaction and gets printed by a ne’er do well person
?