Agenda

• About Us
• Introduction
• What’s a Malware Freakshow?
• Anatomy of a Successful Malware Attack
• Sample Analysis + Victim + Demo
  • Sample SL2009-127 – Memory Rootkit Malware
  • Sample SL2010-018 – Windows Credential Stealer
  • Sample SL2009-143 – Network Sniffer Rootkit
  • Sample SL2010-007 – Client-side PDF Attack
• Conclusions
• Questions -> Q&A Room
About Us

Nicholas J. Percoco / Senior Vice President at Trustwave
  • 15 Years in InfoSec / BS in Computer Science
  • Built and Leads the SpiderLabs team at Trustwave
  • Interests:
    – Targeted Malware, Attack Prevention, Mobile Devices
      • Business / Social Impact Standpoint

Jibran Ilyas / Senior Security Consultant at Trustwave
  • 8 Years in InfoSec / Masters in Infotech Management from Northwestern University
  • Interests:
    – Antiforensics, Artifact Analysis, Real time Defense
Introduction

We had a busy year!!

- Over 200 incidents in 24 different countries
- Hundreds of Samples to pick from
- We picked the most interesting for you

New Targets This Year

- Sports Bar in Miami
- Online Adult Toy Store
- International VoIP Provider
- US Defense Contractor

Malware Developers were busy updating/improving their code

- Many improvements to avoid detection
- Maybe they saw our Freakshow last year 😊
What’s a Malware Freakshow?

We have access to breached environments

- Many of these environments contain valuable data
- Smash and Grab is old school
- Attackers spend average of 156 before getting caught
- With time, comes exploration and development
- Custom and Targeted Malware is the Norm, not the exception
- Gather and perform analysis on each piece of Malware
  - A Malware Freakshow demos samples to the security community
  - Benefit: Learn the sophistication of the current threats
  - Goal: Rethink the way we alert and defend!!!
Anatomy of a Successful Malware Attack

Malware development takes a methodical approach

- Step 1: Identifying the Target
- Step 2: Developing the Malware
- Step 3: Infiltrating the Victim
- Step 4: Finding the Data
- Step 5: Getting the Loot Out
- Step 6: Covering Tracks and Obfuscation (optional)

Before we discuss the samples, we’ll cover this process.
Anatomy – Step 1: Identifying the Target

Target the Data that will lead to the Money

- Credit Card Data
  - Exists in plain text in many type of environments
  - Cash is just 4 hops away

- ATM/Debit Card Data
  - Limited to only ATM Networks and places accepting debit
  - Need PIN as well
  - Cash is just 3 hops away
Anatomy – Step 2: Developing the Malware

Depends on the Target System, but focus on the Big Three

- Keystroke Logger
- Network Sniffer
- Memory Dumper

Design Considerations

- Naming Convention
  - blabla.exe – not the best name choice
  - svchost.exe – much better 😊

- Functionality
  - Slow and Steady wins the race

- Persistency and Data Storage
Anatomy – Step 3: Infiltrating the Victim

Three basic methods of planting your malware:

- **The Physical Way**
  - “Hi, I’m Ryan Jones. Look over there. Owned”

- **The Easy Way**
  - “Nice to meet you RDP & your friend default password”

- **The Über Way**
  - “Silent But Deadly”
Anatomy – Step 4: Finding the Data

The Software Holds the “Secrets”

- Task Manager
  - Busy Processes == Data Processing

- Process’s Folders
  - Temp Files == Sensitive Data

- Configuration Files
  - Debug Set to ON == Shields Down

- The Wire
  - Local Network Traffic == Clear Text
Anatomy – Step 5: Getting the Loot Out

Keep It Simple Stupid

- Little to no egress filtering, doesn’t mean use TCP 6667
- Don’t Reinvent to Wheel
  - FTP
  - HTTP
  - HTTPS
  - RDP
- IT/Security Professional Look for Freaks
  - Traffic on high ports == suspicious
Anatomy – Step 6: Covering Tracks and Obfuscation

Don’t Be Clumsy

• *Test Malware First!*
  - Crashing Systems = Sorta Bad
  - Filling Up Disk Space = Real Bad
  - Shells Popping Up = Very Bad
  - Stealing Mouse Focus = Just Stupid

Mess with the Cops

• MAC times to match system install dates
• Obfuscate Output file; even just slightly
• Pack the Malware
• Randomize Events
• Rootkits
# Sample SL2009-127 – Memory Rootkit Malware

<table>
<thead>
<tr>
<th>Vitals</th>
<th>Code Name: Capt. Brain Drain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filename:</td>
<td>ram32.sys</td>
</tr>
<tr>
<td>File Type:</td>
<td>PE 32-bit, Kernel Driver</td>
</tr>
<tr>
<td>Target Platform</td>
<td>Windows</td>
</tr>
</tbody>
</table>

## Key Features
- Installs malware as a rootkit to stay hidden from process list
- Checks all running processes in kernel for track data
- Output dumped to file w/ “HIDDEN” and “SYSTEM” attributes
- Character substitution in output file to avoid detection
- At set time daily, malware archives data and flushes the data from output file to avoid duplication of stolen data

## Victim
- **Sports Bar in Miami**
  - An elite location that attracts celebrities
  - IT operations outsourced to Third Party
  - Owner throws away security and compliance notices as monthly IT expenses “give him a headache”.
  - Back Office server is also a backup DVR server
Sample SL2009-127 – Memory Rootkit Malware

It’s Demo Time!
### Sample SL2010-018 – Windows Credential Stealer

<table>
<thead>
<tr>
<th>Vitals</th>
<th>Code Name: Don’t Call Me Gina</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filename: fsgina.dll</td>
<td></td>
</tr>
<tr>
<td>File Type: Win32 Dynamic Link Library</td>
<td></td>
</tr>
<tr>
<td>Target Platform: Windows</td>
<td></td>
</tr>
</tbody>
</table>

#### Key Features
- Loads with Winlogon.exe process
- Changes Windows Authentication screen to a “Domain login” screen.
- Stores stolen credentials in ASCII file on system
- Only stores successful logins
- Attempts exporting logins via SMTP to an email address.

#### Victim
- **Online Adult Toy Store**
  - A 100 person company on the West Coast of USA.
  - Outsourced website hosting development to a low cost but well known provider
  - Admin page allows uploads of files
  - Database stores card data for 10 minutes post transaction
**Sample SL2009-143 – Network Sniffer Rootkit**

<table>
<thead>
<tr>
<th>Vitals</th>
<th>Code Name:</th>
<th>Clandestine Transit Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filename:</td>
<td>winsrv32.exe</td>
<td></td>
</tr>
<tr>
<td>File Type:</td>
<td>PE 32-bit</td>
<td></td>
</tr>
<tr>
<td>Target Platform:</td>
<td>Windows</td>
<td></td>
</tr>
</tbody>
</table>

**Key Features**
- PE Executable has components of malware embedded inside it - Ngrep, RAR tool and Config file
- Uses rootkit to hide malware from Task Manager
- Ngrep options contains Track Data regular expression
- At the end of the day, it RARs and password protects the temporary output file and creates new file for next day.
- Exports compressed and password protected data to FTP server set in the config file

**Victim**
- **International VoIP Provider**
- Seven person company (80,000 customers)
- Data Center was in barn; was home to 20 farm cats
- Used publicly available payment application for credit cards
Sample SL2009-143 – Network Sniffer Rootkit

Demo #3!
# Sample SL2010-007 – Client-Side PDF Attack

<table>
<thead>
<tr>
<th><strong>Vitals</strong></th>
<th><strong>Code Name:</strong> Dwight’s Duper</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Filename:</strong></td>
<td>Announcement.pdf</td>
</tr>
<tr>
<td><strong>File Type:</strong></td>
<td>Portable Document Format</td>
</tr>
<tr>
<td><strong>Target Platform:</strong></td>
<td>Windows</td>
</tr>
</tbody>
</table>

**Key Features**

- The attack is customized for victims with enticing email
- Malware attached in email looks like a normal PDF file
- PDF contains shell code which executes upon PDF launch
- Shell code calls a batch file which steals all *.docx, xlsx, pptx and txt files from user’s My Documents folder
- Stolen files are compressed, password protected and sent to FTP over TCP port 443

**Victim**

- US Defense Contractor
- Provides analytics service to US Military
- Egress filtering set to only allow TCP ports 80 and 443
- No inbound access allowed from the Internet without VPN
Conclusions (What we learned in the past year)

Customization of Malware
- One size fits all is not the mantra of attackers today

Slow and Steady wins the race
- Malware writers are not in for quick and dirty hacks. Since data is stolen in transit, persistency is the key.

AntiForensics
- Detection is not easy for these new age malware. MAC times are modified; random events configured and protection from detection built in.

Automation
- Attackers adding layers to malware to automate tasks so that they don’t have to come in to the system and risk detection.

Not Slowing Down
- Since Malware Freakshow last year at SecTor 2009, the techniques have improved significantly.