SwitchMan: An Easy-to-Use Approach to Secure User Input and Output

Shengbao Zheng¹, Zhenyu Zhou¹, Heyi Tang* and Xiaowei Yang¹

¹Duke University, *Tsinghua University
Sensitive user input/output data are vulnerable to data stealing attacks by keyloggers and screen scrapers

- e.g. Carbanak malware [1] in 2015 infects bank computers
- Stole almost one billion dollars from around 100 financial institutions

Challenging because OS provides user-level APIs for sharing the I/Os

- e.g. XGrabKeyboard() in X11
- Allow any malware to steal the keyboard input and screen output
Limitations of Existing Solutions

• All require significant user management
  • VM solution: which VM handles sensitive data
  • Using trusted device (e.g. mobile phone) input/output sensitive data

• Challenging for a non-expert user to manage these tasks

• Need automatically manage the switching to sensitive data input/output without user involvement
Contribution

- **SwitchMan architecture**
  - A secure terminal for handling sensitive input/output data
  - Server initiated switching
  - Defending data stealing attacks by keylogger & screen scraper malware

- **SwitchMan Network Protocol (SNP)**
  - Enables a server to invoke a secure terminal for sensitive data
  - Works even if the client’s software (e.g. a browser) is untrusted
  - Resistant to MITM attack

- **Implement a SwitchMan prototype using Linux**
  - Evaluate its performance
Outline

• Design goals, assumptions, adversary model

• SwitchMan Design
  • SwitchMan Architecture
  • Trusted Input/Output Proxy (TIOP)
  • SwitchMan Network Protocol (SNP)

• Evaluation
• Goals
  • Protect sensitive input/output data against user-level malware
  • Easy to use
  • Efficient

• Assumptions
  • Trusting OS and its vendor
  • Secure storage & network transmission

• Adversary model
  • No physical access
  • Malicious Man-in-the-Middle (MITM)
SwitchMan Architecture

Regular Account
- App
- Untrusted Browser
- Graphic Server

Protected Account
- Trusted I/O Proxy
- Graphic Server

SwitchMan
- Virtual Terminal 1
- Virtual Terminal 2

Filter Sensitive?

Yes

No

To / From Network

User Space

Kernel
Trusted Input/Output Proxy (TIOP)

• A simple web browser
  • Displays the sensitive output
  • Takes a user’s input
• The only application
  • Connects to the graphic server running under the protected account
• Attacker may mimic a TIOP
  • Choose a secret background image
  • Encrypted and stored with a user’s other login credentials
SwitchMan Network Protocol (SNP)

- Protocol to support server initiated switching
- Establish a separate secure connection with the server for sensitive data

Challenges:

1. Support non-SwitchMan-upgraded client
   Negotiation during TCP Handshake
2. MITM attacks & malicious browser
   Separate the secrets for establishing secure connection into two parts:
   TCP Option + HTTPS
Step 1: TCP handshake

SYN with Option(SM)
Step 1: TCP handshake

SYN-ACK with Option(SM Echo)
Step 1: TCP handshake
Step 2: Server Initiated Switching
Step 2: Server Initiated Switching
Step 2: Server Initiated Switching

// First half of the secret

TCP Option(nonce 1, nonce id)
Step 2: Server Initiated Switching

// Second half of the secret

HTTPS(JS(URL_sensitive, nonce 2, nonce id, signature))
Step 3: TIOP Connects to the Server

HTTPS (URL sensitive, nonce 1, nonce 2, nonce id)
Step 3: TIOP Connects to the Server
Step 4: Switching back to the regular account
Evaluation

• Compare SwitchMan with three other systems
  • Qubes OS, CloudTerminal, BitE
  • Usability
  • Security

• Performance
  • Implement a SwitchMan Prototype
  • Measure the latency of Alexa Top 10 financial websites
  • Compare the original response time vs. extra latency
<table>
<thead>
<tr>
<th>Factor</th>
<th>Qubes</th>
<th>CloudTerminal</th>
<th>BitE</th>
<th>SwitchMan</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**USABILITY**
<table>
<thead>
<tr>
<th>Factor</th>
<th>Qubes</th>
<th>CloudTerminal</th>
<th>BitE</th>
<th>SwitchMan</th>
</tr>
</thead>
<tbody>
<tr>
<td>USABILITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nothing-to-carry</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
<td>✓</td>
</tr>
</tbody>
</table>

Duke
<table>
<thead>
<tr>
<th>Factor</th>
<th>Qubes</th>
<th>CloudTerminal</th>
<th>BitE</th>
<th>SwitchMan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nothing-to-carry</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>No user management effort</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
</tbody>
</table>

**USABILITY**
<table>
<thead>
<tr>
<th>Factor</th>
<th>Qubes</th>
<th>CloudTerminal</th>
<th>BitE</th>
<th>SwitchMan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nothing-to-carry</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>No user management effort</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>No noticeable performance degradation</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

**USABILITY**
<table>
<thead>
<tr>
<th>Factor</th>
<th>Qubes</th>
<th>CloudTerminal</th>
<th>BitE</th>
<th>SwitchMan</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>USABILITY</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nothing-to-carry</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>No user management effort</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>No noticeable performance degradation</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td><strong>SECURITY</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor</td>
<td>Qubes</td>
<td>CloudTerminal</td>
<td>BitE</td>
<td>SwitchMan</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------</td>
<td>---------------</td>
<td>------</td>
<td>-----------</td>
</tr>
<tr>
<td><strong>USABILITY</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nothing-to-carry</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>No user management effort</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>No noticeable performance degradation</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td><strong>SECURITY</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCB size</td>
<td>VMM + guest OS kernel + graphic system</td>
<td>kernel modules + hypervisor + cloud</td>
<td>kernel + mobile OS</td>
<td>kernel + graphic system</td>
</tr>
</tbody>
</table>
SwitchMan Latency

Original Load Time

Extra Latency
Conclusion

• SwitchMan
  • an architecture that enables a server to automatically switch a user to a secure terminal for sensitive user input/output

• Lightweight and easy to use

• A valuable design alternative for the real-world to adopt
Thanks