Project Plan
Improved Detonation of Evasive Malware

The Capstone Experience

Team Proofpoint
Ian Murray
Ryan Gallant
Jack Mansueti
Sean Joseph
Tae Park

Department of Computer Science and Engineering
Michigan State University
Fall 2018
Functional Specifications

• Sandbox is essential for malware analysis
• New evasive techniques hinder quarantine
• **Fundamental Solution:** Flag malware whose execution deviates in sandboxes.
• **Auxiliary Solution:** Support autonomous code modification to remove the ability to avoid sandbox execution
• Display in intuitive web UI
Design Specifications

• Evasive Malware Identification
  ▪ Scan for known existing signatures
  ▪ Develop own behavior detection methods

• Malware Modification & Detonation
  ▪ Modify sandbox checks with reverse engineering
  ▪ Forces malware to execute all relevant functions

• Web Interface
  ▪ Top-Level: Displays broad real time data
  ▪ Drill-Downs: Widgets, enters more detailed reports
Design Specifications

The Capstone Experience

Team Proofpoint Project Plan Presentation
Screen Mockup: Top Samples

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Wannacry</td>
</tr>
<tr>
<td>2</td>
<td>Yahoo! toolbar</td>
</tr>
<tr>
<td>3</td>
<td>Hotmail toolbar</td>
</tr>
<tr>
<td>4</td>
<td>Harddrive encryption</td>
</tr>
<tr>
<td>3</td>
<td>Performs a whatismyip.com lookup</td>
</tr>
</tbody>
</table>
## Screen Mockup: Top Techniques

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Starts listening on a high TCP port</td>
</tr>
<tr>
<td>2</td>
<td>Checks the local time</td>
</tr>
<tr>
<td>3</td>
<td>Performs a whatismyip.com lookup</td>
</tr>
<tr>
<td>4</td>
<td>Writes files to disk</td>
</tr>
<tr>
<td>5</td>
<td>Gets a list of running processes</td>
</tr>
<tr>
<td>6</td>
<td>Modifies Registry</td>
</tr>
<tr>
<td>7</td>
<td>Installs itself as autorun</td>
</tr>
<tr>
<td>8</td>
<td>Checks processor information</td>
</tr>
<tr>
<td>9</td>
<td>Creates a hidden file</td>
</tr>
<tr>
<td>10</td>
<td>Executes a separate process</td>
</tr>
</tbody>
</table>
Screen Mockup: System State

System State

<table>
<thead>
<tr>
<th>Postgres</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Flask</td>
<td>Unable to connect to Cuckoo!</td>
</tr>
<tr>
<td>Cuckoo</td>
<td></td>
</tr>
</tbody>
</table>
Screen Mockup: Sample Queue
### Screen Mockup: Results

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>sample1.exe</td>
<td>CLEAN</td>
</tr>
<tr>
<td>2</td>
<td>sample2.exe</td>
<td>SUCCESSFUL EXECUTION</td>
</tr>
<tr>
<td>3</td>
<td>sample3.exe</td>
<td>INTERESTING</td>
</tr>
<tr>
<td>4</td>
<td>sample4.exe</td>
<td>MODIFIED</td>
</tr>
</tbody>
</table>
Screen Mockup: Results w/ Filter
Technical Specifications

• Front End UI
  ▪ Bootstrap, jQuery, HTML5, and CSS3 are used to effectively present users with appropriate data from the malware detonation system.

• Web Application
  ▪ Apache, Flask, and Python are used to serve our web application.
  ▪ PostgreSQL is used for data storage outside the data Cuckoo’s API provides.
  ▪ SQLAlchemy is used for mapping Python Objects to PostgreSQL statements and schema.

• Backend Malware Analysis
  ▪ Cuckoo and Suricata are used for detonation and classification, Python is used to disassemble and modify malware samples classified as evasive.
System Architecture

Web Front End

Web Server

Database

Analysis

Bootstrap
jQuery

vmware

Python
Flask
Alchemy

PostgreSQL

The Capstone Experience
System Components

- **Software Platforms / Technologies**
  - **Front End**
    - Python 3.6
    - HTML & CSS3
    - Bootstrap CSS
    - Cuckoo API
    - Flask
    - jQuery
  - **Back End**
    - Python 2.7
    - Cuckoo
    - Suricata
    - PostgreSQL
    - SQLAlchemy
    - Apache
    - VMWare
Risks

• Reverse Engineering Difficulty
  ▪ Malware samples are rarely available as readable code.
  ▪ Variety of tools for disassembly.

• Multiple Language Proficiency
  ▪ Malware comes in variety of languages.
  ▪ Limit analysis to a subset of the greater universe of languages.

• Navigating Proofpoint’s Lab
  ▪ Unknown how customizable Proofpoint’s lab environment is.
  ▪ Client runs samples the team uploads via Secureshare.

• Malware Samples Evade through Unknown Means
  ▪ Unknown how a sample determines the difference between a live machine and a sandbox.
  ▪ Proofpoint has identified several evasive malware for the team to examine.
Questions?