Beta Presentation
Defeating Malware Payload Obfuscation

The Capstone Experience
Team Proofpoint

Adam Johanknecht
Nick Lojewski
Vivian Qian
Derek Renusch
Dan Somary

Department of Computer Science and Engineering
Michigan State University
Spring 2019
Project Overview

• Create a machine learning system to classify files as malicious or benign
  ▪ Accuracy goal: have at least the same accuracy as sandbox detonation
  ▪ Performance goal: be at least 50% faster than detonation in Cuckoo

• Display information in web dashboard
  ▪ High level system information
  ▪ Ability to look at details for individual files
System Architecture

- Sandbox
- Controller
  - Python
  - Cuckoo
- Inputs/Outputs
  - Queue Folder
  - Output Folder
- Machine Learning Algorithm
  - Scikit-learn
  - Python
- Web Monitoring System
  - HTML
  - Flask
Updated Main Dashboard

Queue
- File Name: pcal.png, MD5 Hash: 320cb0ad18bf570e6c9665a40cb0d5
- File Name: pcal.png, MD5 Hash: 320cb0ad18bf570e6c9665a40cb0d5

Processed
- File Name: gnoccihead.exe, MD5 Hash: 8f4927900d71031dfe95c89f9773996e76, Classification: Benign
- File Name: bcPayload.bin, MD5 Hash: d0d207bc483d111874785ad832030a4b64a, Classification: Malicious
- File Name: WannaCry.exe, MD5 Hash: b4d82839e5d212b4f3b76a617908b549, Classification: Malicious
- File Name: PaintDotNet.exe, MD5 Hash: 67739115b9b0516b292cd4be673a94e1, Classification: Benign
- File Name: Dieblo_R.Linux, MD5 Hash: 27074219327e30e48b50c4de71e8fd9, Classification: Benign
- File Name: nice.exe, MD5 Hash: d057a6f40b667ebcd2410c8b1abc1f1e, Classification: Benign

Processing in Cuckoo
- File Name: nihuru.exe, MD5 Hash: 79d4935ef203f32f12cd4b79ded9b915, Status: Running, Task ID: 24
- File Name: Firefox_Setup_65.0.1.exe, MD5 Hash: b6f0ec77ac4ef9bedae6b502a10beb7a8, Status: Pending, Task ID: 25

Statistics
- Benign Files Processed: 3
- Malicious Files Processed: 2
- Files classified by Full Datonation: 3
- Files classified by ML/Static Analysis: 3
- Ratio: 0.5
Image Drill-Down Page

File Drill Down: drowen.png

File Classification

<table>
<thead>
<tr>
<th>Filename</th>
<th>drowen.png</th>
</tr>
</thead>
<tbody>
<tr>
<td>MD5</td>
<td>d67f93763088270e191962ec0bde16b4</td>
</tr>
<tr>
<td>Score</td>
<td>0/10</td>
</tr>
<tr>
<td>Classification</td>
<td>Not very suspicious</td>
</tr>
</tbody>
</table>

File Attributes

<table>
<thead>
<tr>
<th>Filetype</th>
<th>PNG Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size</td>
<td>91.5859375 KB</td>
</tr>
<tr>
<td>IDAT Size</td>
<td>8192</td>
</tr>
<tr>
<td>IEND Detected</td>
<td>False</td>
</tr>
<tr>
<td>Hex Chunks</td>
<td>12</td>
</tr>
<tr>
<td>High Entropy Hits</td>
<td>0</td>
</tr>
<tr>
<td>Variance Detected</td>
<td>False</td>
</tr>
</tbody>
</table>
Office Document Drill-Down Page

```
<table>
<thead>
<tr>
<th>File Classification</th>
<th>File Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filename</td>
<td>Microsoft Office Document</td>
</tr>
<tr>
<td>MD5</td>
<td>Size 15.736328125KB</td>
</tr>
<tr>
<td>Classification</td>
<td>Creating Application</td>
</tr>
<tr>
<td></td>
<td>Contains Macro</td>
</tr>
<tr>
<td></td>
<td>Number of Yara Matches</td>
</tr>
</tbody>
</table>
```

File Drill Down: cse422_hw3.docx
### File Search Page

![File Search Page](image)

#### Search

- **Search**: Text input field for search queries.
- **Search** button to submit the search query.
- **Files Type**: Options to filter by file type (exe, bin).

#### Results

<table>
<thead>
<tr>
<th>#</th>
<th>File Name</th>
<th>MD5 Hash</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>WannaCry.exe</td>
<td>84c82835a5d21bce7f75e61706d8ab549</td>
</tr>
</tbody>
</table>
What’s left to do?

• Improve accuracy of image and office document classification
• Enhance reporting on system health
• Create documentation and refactor the code base
Questions?