Malicious Browser Extensions at Scale

Bridging the Observability Gap between Web Site and Browser

Louis F. DeKoven¹, Stefan Savage¹, Geoffrey M. Voelker¹, Nektarios Leontiadis²

¹UC San Diego, ²Facebook
Attacks on Social Media

• Social media is targeted by malware
  • Reach a large number of users quickly
  • Users inherently trust content within a social network
Attacks on Social Media

• Social media is targeted by malware
  • Reach a large number of users quickly
  • Users inherently trust content within a social network

• Malware infects user’s browser then
  • Infect other social media users
  • Steal the user’s passwords
Attacks on Social Media

• Social media is targeted by malware
  • Reach a large number of users quickly
  • Users inherently trust content within a social network

• Malware infects user’s browser then
  • Infect other social media users
  • Steal the user’s passwords

• Leverage the vantage point of a social network to
  • Detect devices infected with malware
  • Clean up malware from infected devices
Objectives

• Detect and label malicious browser extensions quickly
  • Google Chrome
  • Mozilla Firefox

• Automatically cleanup infected devices

• Detect new malicious browser extensions automatically
Objectives

- Detect and label malicious browser extensions quickly
  - Google Chrome
  - Mozilla Firefox
- Automatically cleanup infected devices
- Detect new malicious browser extensions automatically

**Malicious Browser Extensions (MBE):** extensions that take actions on behalf of a user without their consent, or replace Facebook’s key functionality or content.
Browser Extensions

- Motivation
- Background
- Methodology
- Results
- Evaluating Alternatives
- Conclusion
Browser Extensions 101

- Enhance user experience beyond a Web page
- Can change visual appearance of Web pages
- Can change how the browser interacts with Web pages
Browser Extensions 101

- Enhance user experience beyond a Web page
- Can change visual appearance of Web pages
- Can change how the browser interacts with Web pages
Browser Extensions 101

- Enhance user experience beyond a Web page
- Can change visual appearance of Web pages
- Can change how the browser interacts with Web pages
- How?
  - Have elevated set of privileges
Browser Extensions 101

- Enhance user experience beyond a Web page
- Can change visual appearance of Web pages
- Can change how the browser interacts with Web pages

How?
- Have elevated set of privileges
  - Modify HTTP headers
  - Change Content Security Policy
  - Rewrite any Web site content
Browser Extensions 101

• Example MBE targeting Facebook
  • Steals user’s Facebook access token
  • Generates likes
  • Subscribes to YouTube channels
  • And more…

https://kjaer.io/extension-malware/
Defending Against MBE

- Harden the browser [1,2,3]
- Detecting extensions vulnerable to Web page JavaScript[4]
- Vetting code within extension marketplaces [5]
- Dynamic analysis and sandboxing [6,7]

It’s **Hard** to Detect MBE

- **Anti-malware products**
  - May run static analysis on extension JavaScript
  - Struggle with dynamic resources

- **Extension marketplaces/Browser vendors**
  - May track how extensions use the browser
  - Struggle with temporal badness

- **Researchers**
  - May run sandboxed analysis
  - Struggle with scale and temporal badness
A *Different Perspective*

Social media networks directly experience abusive extensions

Leverage the vantage point of a social media network
Detecting MBE

- Motivation
- Background
- Methodology
- Results
- Evaluating Alternatives
- Conclusion
Challenges in Detecting MBE

• How do we know what extensions are bad?
  • Facebook has to build signatures to detect MBE
Challenges in Detecting MBE

• How do we know what extensions are bad?
  • Facebook has to build signatures to detect MBE

• Facebook does not know what extensions are installed
  • Can detect user accounts acting in abusive ways
Challenges in Detecting MBE

• How do we know what extensions are bad?
  • Facebook has to build signatures to detect MBE

• Facebook does not know what extensions are installed
  • Can detect user accounts acting in abusive ways

• Facebook can not collect extensions from facebook.com due to browser security
  • Can build a binary to collect installed extensions
Challenges in Detecting MBE

- How do we know what extensions are bad?
  - Facebook has to build signatures to detect MBE

- Facebook does not know what extensions are installed
  - Can detect user accounts acting in abusive ways

- Facebook can not collect extensions from facebook.com due to browser security
  - Can build a binary to collect installed extensions

- Insight: We can link extension content to abusive content
System Methodology

Using signals from malware within Facebook enables the detection and remove MBE at a large scale

We do this by:

• Identifying compromised Facebook accounts

• With user consent, we fetch the installed extensions from devices exhibiting malicious behavior

• Determine if the extension is malicious or benign by comparing it to abusive content (while fetching extensions)

• If the extension is malicious remove it from the user’s device
System Design

• Detecting compromised user accounts
Detecting Compromised User Accounts

- Spiking content
  - Monitor time series of user activity
Detecting Compromised User Accounts

- Spiking content
  - Monitor time series of user activity

- Document Object Model (DOM) based detection
  - Periodically scan Facebook’s DOM for third-party elements
Detecting Compromised User Accounts

• Spiking content
  • Monitor time series of user activity

• Document Object Model (DOM) based detection
  • Periodically scan Facebook’s DOM for third-party elements

• Negative feedback
  • Feedback on posted content
System Design

• Detecting compromised user accounts

• Anti-malware scanner
Anti-Malware Scanner

- Facebook’s custom scanner is executed on the compromised device following user consent
Anti-Malware Scanner

- Facebook’s custom scanner is executed on the compromised device following user consent

- Uploads digital fingerprint of extensions to Facebook
  - MD5 hash

- New extensions are uploaded to Facebook

- When MBE are detected they are removed

- Third-party anti-virus scanner executed
System Design

- Detecting compromised user accounts
- Anti-malware scanner
- Static analysis pipeline
Static Analysis Pipeline

- **Unpacking**
  - Recursively unpack the extension and files

- **Indicator extraction**
  - Deobfuscate, decode, and repair broken URLs
  - Regular expressions extract indicators e.g. URLs, API keys
    - Treating each file as text

- **Insight:** Extensions collected by Facebook’s malware scanner exhibited malicious behavior at the time of collection
System Design

• Detecting compromised user accounts
• Anti-malware scanner
• Static analysis pipeline
• Extension labeling
Indicator Labeling

- **MALICIOUS**
  - Malicious with high-confidence

- **UNKNOWN**
  - Default label for all samples

- **NON_MALICIOUS**
  - Benign samples, or samples from trusted sources

- Labels produced by system that detects compromised accounts
Propagating Indicator Labels

• Apply vetted threat labels to indicators from static analysis

• How do we label extensions?
  • JavaScript contains a MALICIOUS URL
  • MALICIOUS label propagates to the file
  • MALICIOUS label propagates the extension

• Erroneously marked indicators
  • Propagate automatically
  • Rules in place to prevent single indicators from mass-labeling
  • Manual labels overrides automated labeling
System Results

- Motivation
- Background
- Methodology
- Results
- Evaluating Alternatives
- Conclusion
Malicious Indicators

<table>
<thead>
<tr>
<th>Extension Contents</th>
<th>JS</th>
<th>HTML</th>
<th>Extracted Indicators</th>
<th>Scan Sessions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total #</td>
<td>Malicious (#%)</td>
<td>#</td>
<td>%</td>
</tr>
<tr>
<td>Chrome Ext.</td>
<td>67 380</td>
<td>720</td>
<td>66 134</td>
<td>1 559 (2.4%)</td>
</tr>
<tr>
<td>Firefox Ext.</td>
<td>17 979</td>
<td>16</td>
<td>19 004</td>
<td>609 (3.2%)</td>
</tr>
<tr>
<td>Total Unique</td>
<td>84 905</td>
<td>733</td>
<td>73 281</td>
<td>1 516 (2.1%)</td>
</tr>
</tbody>
</table>

- 6-week measurement period
- Only a small number of all indicators are labeled MALICIOUS
Malicious Extensions

<table>
<thead>
<tr>
<th>All Extensions</th>
<th>Malicious Extensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>%</td>
</tr>
<tr>
<td>Chrome Ext.</td>
<td>23 376</td>
</tr>
<tr>
<td>Firefox Ext.</td>
<td>11 183</td>
</tr>
<tr>
<td>Total Unique</td>
<td>34 559</td>
</tr>
</tbody>
</table>

- A high proportion (5.2%) of malicious extensions is expected as our system targets devices exhibiting malicious behavior.

- 422 of 1,697 Chrome MBE were once online Google’s Web Store
  - Suggests a high number of MBEs to be side loaded
MBE Detection Rates

- Average 39.5 Chrome MBE/day
- Average 2 Firefox MBE/day

- 92% of new MBE are labeled by a median time of 21 seconds
- 8% of new MBE are labeled more than one day after collection
  - Detected on 9% of user devices cleaned during the experiment

This result is expected from an indicator-based labeling system as labels can change over time
Known False Positives

- 124 extensions are incorrectly labeled MALICIOUS
- 0.8% of all scan sessions removed one or more of these extensions
- Median detection time: 18 days

- This result is expected from an indicator-based labeling system as labels can change over time
- We find the low number of incorrectly labeled MBES to be an acceptable tradeoff
Comparing Systems

- Motivation
- Background
- Methodology
- Results
- Evaluating Alternatives
- Conclusion
Evaluating Alternatives

• Was it necessary to create a new system that detects MBE?
• Focus on Chrome extensions
  • Majority of extensions are for Chrome browser
  • Each Chrome extension’s Web store presence is checked
  • 2,200/23,376 Chrome extensions once on the Chrome Web store

• Facebook labels 422 (19.2%) MALICIOUS
• Facebook labels 1,778 (80.8%) UNKNOWN
VirusTotal

- Provided with 9,172 unique CRX from authors of Hulk[1]
  - VT was aware of only 73 extensions
  - Moreover 5 are labeled MALICIOUS by at least 1 anti-virus engine

Facebook cannot use general malware databases to detect MBEs

VirusTotal

- Provided with 9,172 unique CRX from authors of Hulk[1]
  - VT was aware of only 73 extensions
  - Moreover 5 are labeled MALICIOUS by at least 1 anti-virus engine

Facebook cannot use general malware databases to detect MBEs

- Of the 422 MBE identified by Facebook
  - 96 (22.7%) are labeled MALICIOUS by one or more anti-virus engine

Facebook cannot rely on anti-malware engines to identify MBEs

Google Chrome Web Store

- By the six-week period Google removed 367 of the 2,200
  - 70 MALICIOUS
  - 297 UNKNOWN

Facebook cannot rely on Google to remove all MBE targeting FB

- Does Facebook identify MBEs faster?
  - These 70 MBE have over 1 million installs according the the Web Store
  - Facebook identifies the 70 MBE with a median time of 2.8 days (67.3 hours) before they are removed from the Web store

Our system successfully reduces the median monetization time of MBE
Take Away

MBE are challenging to address from any single vantage point

- **Browser vendors**
  - Can restrict extension distribution
  - Have limited insight into abusive extensions in the wild

- **Abused sites**
  - Directly experience malicious behavior
  - But are not in a position to identify which extensions are implicated
Conclusion

• This system is currently running to protect users of Facebook
• As a result Facebook is able to very quickly detect and remove new MBE at scale

422 Chrome MBE MD5 hashes: https://pastebin.com/nzVGPLnr
• Samples available in VirusTotal and Facebook ThreatExchange