EXPLOITING SERVER SIDE TEMPLATE INJECTION WITH TPLMAP

BY: DIVINE SELORM TSA

18 AUG 2018
Outline

- Introduction
- Template Engines
- SSTI
- SSTI Methodology
- Tplmap
- Demo
- Remediation
What is a template engine?

- Helps populate dynamic data into modern web pages
- Enables developers to separate data processing logic and presentation code
- Offers rich functionality through Wikis, CMS, blogs
- Uses:
  - Displays information about users, products, companies
  - Displays gallery of photos, videos..
  - Sends bulk emails
Example: Jinja

```
<html>
<head>
<title>{{ title }}</title>
</head>
<body>

Hello.
</body>
</html>
```

```
from jinja2 import Environment, FileSystemLoader
import os

# current directory
MY_DIR = os.path.dirname(os.path.abspath(__file__))

def render_html_page():
    # jinja2 environment
    env = Environment(loader=FileSystemLoader(MY_DIR), trim_blocks=True)
    print env.get_template('template.html').render(title='Hello gdieu from owasp Montreal')

if __name__ == '__main__':
    render_html_page()
```

```
"example2.py" 21L, 573C written
gdieu@stsi:~$ python example2.py
<html>
<head>
<title>Hello gdieu from owasp Montreal</title>
</head>
<body>

Hello.
</body>
</html>
gdieu@stsi:~$
```
Popular Template Engines

- PHP – Smarty, Twigs
- JAVA – Velocity, Freemake
- Python – JINJA, Mako, Tornado
- JavaScript – Jade, Rage
- Ruby - Liquid
What is template injection?
What is template injection?

- Occurs when invalid user input is embedded into the template engine
- Often XSS attack occurs but SSTI can be missed
- Can lead to a remote code execution (RCE)
- Developer error or intentional exposure
Methodology (based on James Kettle’s research)

https://portswigger.net/blog/server-side-template-injection
Detect

- Wappalyzer + builtwith + vulners scanner
- Test fuzzing – Tips:
  - Trying a basic XSS
  - Trying a math expression {{2*2}}
Exploit

- Read
- Explore
- Attack
Tplmap

• Tplmap assists the exploitation of Code Injection and Server-Side Template Injection vulnerabilities with a number of sandbox escape techniques to get access to the underlying operating system.
• The tool and its test suite are developed to research the SSTI vulnerability class and to be used as offensive security tool during web application penetration tests.

https://github.com/epinna/tplmap
Demo - Tplmap
Remediation

• Sanitization
  – Sanitize user input before passing it into the templates

• Complementary approach
  – Use a sandbox within a safe environment
References

- https://portswigger.net/blog/server-side-template-injection
- https://github.com/epinna/tplmap
- https://www.we45.com/blog/server-side-template-injection-a-crash-course-