x86 shellcoding cakestar

Prepara i tuoi shellcode personalizzati per i tuoi exploit

HACK IN BO®
Safe Edition
2020
14ª EDIZIONE

CODICE INSICURO

Agenda

- Perché?
- !/bin/sh
  - Come faccio ora
  - Scheletro del mio shellcode
  - Lo traduco in assembler
  - Le prime riscritture
  - Lo riscrivo come ADD
  - Un pizzico di encoding
- Compiti a casa

https://unsplash.com/photos/neu4T59mTcY

Disclaimer

- Scrivere shellcode ed exploit non è illegale
- Fare penetration test in maniera professionale, manlevati dal proprio cliente, non è illegale
- Lanciare exploit su server a caso senza autorizzazione, **E'** illegale.
- Quello che vediamo qui è per
  - divertimento
  - imparare a non dipendere dai tool

https://unsplash.com/photos/GpOpP4YPu30

msfvenom vs Giovanni Analista

1.0 - execve(“/bin/sh”, 0, 0)

2.0 - manovre evasive!

2.1 - diamoci dei privilegi

2.5 - togliamo i null bye

3.0 - uno shellcode di ADD

Metto un registro a 0

- Prendiamo una word a caso: 0xC8D9EA45
- Calcoliamone il NOT: 0x372615BA
- La AND tra questi due numeri sarà sempre 0
- Qualsiasi numero in AND con 0 sarà pari a 0

QUINDI

```
AND EAX, 0xC8D9EA45

AND EAX, 0x372615BA
```

EAX = 0 qualsiasi sia il valore iniziale di EAX

Dove sono nello stack?

- `PUSH ESP`
- `POP EAX`

Ora EAX contiene la mia posizione nello stack

- `ADD AX, 0x964`
- `PUSH EAX`
- `POP ESP`

Mi sono fatto un po’ spazio forzatamente nello stack ed uso EBX come puntatore per poi fare un JMP ed atterrare nel mio shellcode

- `MOV EBX, ESP`

Creo uno shellcode usando solo ADD

- Utile quando ho un charset limitato
- Meglio usare come partenza uno shellcode piccolo perché introduco molto overhead
- Ispirato da:
  - http://phrack.org/issues/57/15.html#article
  - https://www.youtube.com/watch?v=gHISpAZiAm0

4.0 - nascondino

Giri di XOR

- Decodifico con una XOR ogni word di 32 bit del mio shellcode offuscato
- La XOR è un’operazione reversibile
- Otterrò nello stack il mio shellcode “in chiaro”
- Attenzione a lanciare direttamente l’eseguibile compilato da questo assembler. Va in SEGFAULT, perché?

Per le domande:
https://bit.ly/2XgUzSm

HACK IN BO®
Safe Edition
2020
14ª EDIZIONE

Grazie!