Laborator IV: Mecanisme de Protecție

Iosif George-Andrei

Tabelă de Conținut

Tabelă de Conținut

Tabelă de Conținut

Tabelă de Conținut

Tabelă de Conținut

Tabelă de Conținut

Tabelă de Conținut

Tabelă de Conținut

Mecanisme de Protecție

Eliminarea Informațiilor I

Eliminarea Informațiilor I

- Execuția nu necesită toate informațiile existente într-un executabil.

Eliminarea Informațiilor I

- Execuția nu necesită toate informațiile existente într-un executabil.
  - Numele unor simboluri (din secțiunile `.symtab` și `.dynsym`)

Eliminarea Informațiilor I

- Execuția nu necesită toate informațiile existente într-un executabil.
  - Numele unor simboluri (din secțiunile `.symtab` și `.dynsym`)
  - Informații pentru depanare (din secțiunile specifice formatului DWARF, numite `.debug_*`)

Eliminarea Informațiilor II

Eliminarea Informațiilor II

- Măsură de securitate aplicată după compilarea executabilului și înainte de distribuirea lui către utilizatori

Eliminarea Informațiilor II

- Măsură de securitate aplicată după compilarea executabilului și înainte de distribuirea lui către utilizatori
- Avantaje

Eliminarea Informațiilor II

- Măsură de securitate aplicată după compilarea executabilului și înainte de distribuirea lui către utilizatori
- Avantaje
  - Reducerea dimensiunii executabilului

Eliminarea Informațiilor II

- Măsură de securitate aplicată după compilarea executabilului și înainte de distribuirea lui către utilizatori
- Avantaje
  - Reducerea dimensiunii executabilului
  - Execuție mai rapidă

Eliminarea Informațiilor II

- Măsură de securitate aplicată după compilarea executabilului și înainte de distribuirea lui către utilizatori
- Avantaje
  - Reducerea dimensiunii executabilului
  - Execuție mai rapidă
  - Dezvăluirea a cât mai puține informații către utilizatori (eventual și atacatori)

Eliminarea Informațiilor III

Eliminarea Informațiilor III

- Instrumente pentru eliminarea informațiilor

Eliminarea Informațiilor III

- Instrumente pentru eliminarea informațiilor
  - `strip`

Eliminarea Informațiilor III

- Instrumente pentru eliminarea informațiilor
  - `strip`
  - `gcc -s`

Împachetare I

Împachetare I

- Mecanism care comprimă executabilul curent, încorporând rezultatul într-un alt executabil (care se ocupă numai de despachetare)

Împachetare I

- Mecanism care comprimă executabilul curent, încorporând rezultatul într-un alt executabil (care se ocupă numai de despachetare)
- Despachetare în memorie sau în fișier temporar

Împachetare I

- Mecanism care comprimă executabilul curent, încorporând rezultatul într-un alt executabil (care se ocupă numai de despachetare)
- Despachetare în memorie sau în fișier temporar
- Ultimate Packer for eXecutables (abreviat UPX) ca cel mai cunoscut utilitar multi-platformă pentru împachetare

Împachetare II

Împachetare II

- Roluri

Împachetare II

- Roluri
  - Reducerea dimensiunii executabilului și îngreunarea analizei de către posibili atacatori

Împachetare II

- Roluri
  - Reducerea dimensiunii executabilului și îngreunarea analizei de către posibili atacatori
  - Reducerea dimensiunii programelor malițioase și îngreunarea analizei de către analiștii de securitate

Canarii I

Canarii I

- Mecanismul constă în plasarea unor valori pe stivă, pentru a detecta tentativele de suprascriere.

Canarii I

- Mecanismul constă în plasarea unor valori pe stivă, pentru a detecta tentativele de suprascriere.
- Nume provenit de la păsările care intrau înaintea minerilor în subteran, pentru a detecta niveluri prea mari de gaz

Canarii II

Canarii II

- Valori folosite

Canarii II

- Valori folosite
  - Fixe, de obicei un terminator pentru posibile funcții de copiere a șirurilor de caractere

Canarii II

- Valori folosite
  - Fixe, de obicei un terminator pentru posibile funcții de copiere a șirurilor de caractere
  - Aleatorii, de exemplu din `/dev/urandom`

Canarii II

- Valori folosite
  - Fixe, de obicei un terminator pentru posibile funcții de copiere a șirurilor de caractere
  - Aleatorii, de exemplu din `/dev/urandom`
  - Altele, de exemplu rezultate ale unor `xor`-uri

Canarii III

Canarii III

- Tehnici de evaziune

Canarii III

- Tehnici de evaziune
  - Deducerea valorii de canar

Canarii III

- Tehnici de evaziune
  - Deducerea valorii de canar
  - Folosirea de atacuri care pot scrie la o anumită zonă de memorie (de exemplu, cele cu șiruri de caractere de formatare)

Address Space Layout Randomization I

Address Space Layout Randomization I

- Constă în maparea segmentelor executabilului la adrese aleatorii de memorie.

Address Space Layout Randomization I

- Constă în maparea segmentelor executabilului la adrese aleatorii de memorie.
- Verificarea activării prin citirea conținutului `/proc/sys/kernel/randomize_va_space`

Address Space Layout Randomization II

Address Space Layout Randomization II

- Segmente vizate

Address Space Layout Randomization II

- Segmente vizate
  - Stivă

Address Space Layout Randomization II

- Segmente vizate
  - Stivă
  - Librării dinamice (cu ajutorul secțiunilor `.plt` și `.got`)

Address Space Layout Randomization II

- Segmente vizate
  - Stivă
  - Librării dinamice (cu ajutorul secțiunilor `.plt` și `.got`)
  - *Heap*

Address Space Layout Randomization II

- Segmente vizate
  - Stivă
  - Librării dinamice (cu ajutorul secțiunilor `.plt` și `.got`)
  - *Heap*
  - Cod (numai la activarea mecanismului Position Independent Code)

Address Space Layout Randomization III

Address Space Layout Randomization III

- Tehnici de evaziune

Address Space Layout Randomization III

- Tehnici de evaziune
  - Atacuri cu forță brută

Address Space Layout Randomization III

- Tehnici de evaziune
  - Atacuri cu forță brută
  - nop *sled*

Address Space Layout Randomization III

- Tehnici de evaziune
  - Atacuri cu forță brută
  - nop *sled*
  - `jmp esp` sau `call esp`

Address Space Layout Randomization III

- Tehnici de evaziune
  - Atacuri cu forță brută
  - nop *sled*
  - `jmp esp` sau `call esp`
  - Obținerea unor informații despre memoria procesului

Bitul NX

Bitul NX

- Imposibilitatea unei pagini de a avea drepturi de scriere și execuție în același timp

Bitul NX

- Imposibilitatea unei pagini de a avea drepturi de scriere și execuție în același timp
- Tehnici de evaziune

Bitul NX

- Imposibilitatea unei pagini de a avea drepturi de scriere și execuție în același timp
- Tehnici de evaziune
  - Atacuri de tip Return Oriented Programming (abreviat ROP)

Bitul NX

- Imposibilitatea unei pagini de a avea drepturi de scriere și execuție în același timp
- Tehnici de evaziune
  - Atacuri de tip Return Oriented Programming (abreviat ROP)
  - Apeluri către `mprotect`

Bitul NX

- Imposibilitatea unei pagini de a avea drepturi de scriere și execuție în același timp
- Tehnici de evaziune
  - Atacuri de tip Return Oriented Programming (abreviat ROP)
  - Apeluri către `mprotect`
  - Atacuri de tip Return-to-libc

Exerciții

Exerciții

Exerciții

1. Verificarea Activării unor Mecanisme de Securitate

Recomandări

Recomandări

- Folosiți comanda `man` pentru a primi ajutor la rularea anumitor comenzi.