

Unix et Programmation Shell

aucun document autorisé

23 mai 2023

Q 1. Cochez la case contenant un intrus :

L. Torwald	B. Gate	D. Ritchie
K. Thompson	R. Stallman	D. McIlroy

Q 2. Cochez les outils de débogage :

gcc	grep	gdb
valgrind	make	ptrace

Q 3. Cochez l'intrus :

text	pile	tas
code	source	stack

Q 4. Cochez tous les filtres unix :

tail	bc	cut
grep	gdb	stat

Q 5. Préciser le statut de terminaison de :

```

1 int main( int argc, char * argv[] )
2 {
3     char * p;
4     while ( 1 ) {
5         p = random();
6         *p = 0;
7     }
8     return 0;
9 }
```

sortie sans erreur	accès interdit
débordement de pile	boucle infinie

Q 6. Cocher les 2 affichages les plus probables!?

```
~> echo -n $?; echo $?
```

00	01	10	11
----	----	----	----

Q 7. Laquelle des variables PATH ci-dessous pose le plus sérieux problème de sécurité?

(1) \$HOME/bin:/usr/bin:/bin/tmp
(2) \$HOME:/usr/bin:/root
(3) /tmp:/usr/bin:\$HOME/script

Q 8. Préciser le statut de terminaison de :

```

1 int main( int argc, char ** argv )
2 {
3     main( argc, argv )
4     return 0;
5 }
```

débordement de pile	accès interdit
sortie sans erreur	boucle infinie

```

1 // extrait de la source ctrl.c
2 int main( int argc, char * argv[] )
3 { char cmd[ 128 ];
4   sprintf ( cmd, "grep -q %s %s", argv[1],
5           database );
6   if ( ! system( cmd ) )
7       traitement();
8   else puts("aces interdit ");
9   return 0;
10 }
```

Q 9. Commenter le résultat des commandes :

```

~> gcc ctrl.c -o ctrl.exe
~> str='@'
~> while ./ctrl.exe $str
do s=@$str done > /dev/null
Erreur de segmentation (core dumped)
~> echo ${#X}
116
```

Q 10. Comment obtenir la liste des utilisateurs privilégiés après avoir lancé les commandes :

```

~> gcc ctrl.c
~> echo oops > ctrl.exe
```