

# Unix et Programmation Shell

## aucun document autorisé

27 juin 2023

Q 1. Cocher les hackers à l'origine du système GNU/Linux :

<input type="checkbox"/> L. Torwald	<input type="checkbox"/> B. Gate	<input type="checkbox"/> D. Ritchie
<input type="checkbox"/> K. Thompson	<input type="checkbox"/> R. Stallman	<input type="checkbox"/> D. McIlroy

Q 2. Quel outil permet (entre autres) de construire et maintenir à jour les exécutables ?

<input type="checkbox"/> gcc	<input type="checkbox"/> valgrind	<input type="checkbox"/> make
------------------------------	-----------------------------------	-------------------------------

Q 3. Cochez l'intrus :

<input type="checkbox"/> stack overflow	<input type="checkbox"/> stack smashing	<input type="checkbox"/> stack shreeding
---	---	--

Q 4. Cochez tous les filtres unix :

<input type="checkbox"/> tail	<input type="checkbox"/> bc	<input type="checkbox"/> cut
<input type="checkbox"/> grep	<input type="checkbox"/> head	<input type="checkbox"/> tr

```
1 int main( int argc, char * argv[] )
2 {     int z = atoi( argv[1] );
3     printf ( "%d", z );
4     return 1;
5 }
```

Listing 1 – source toto.c

Q 5. Quel est l'affichage produit par :

```
↪ gcc toto.c
↪ ./a.out 123
↪ echo $?
```

Q 6. Relever l'anomalie :

```
↪ ls -l /usr/bin/passwd | cut -d' ' -f1,3
-rwxr-xr-x. root
```

Q 7. Quel est l'affichage produit par :

```
↪ echo hello 1> /dev/null
↪ echo world 2> /dev/null
```

Q 8. 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 9. Préciser le statut de terminaison de :

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

<input type="checkbox"/> débordement de pile	<input type="checkbox"/> accès interdit
<input type="checkbox"/> sortie sans erreur	<input type="checkbox"/> boucle infinie

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

Listing 2 – ctrl.c

Q 10. Quelle ligne de code présente un risque de débordement de tampon ?

```
↪ whoami
root
↪ gcc ctrl.c -o ctrl.exe
↪ chmod u+s a+x ctrl.exe
```

Q 11. Le super utilisateur ne respecte pas un impératif de sécurité concernant l'emploi de la fonction `system`. Lequel ?