



INF4420A – Sécurité Informatique

Hiver 2020

Travail Pratique 1

Groupe 3

1896939 – Celtis de Chardon

1846754 – Kevin Pastor

1721035 – Mazigh Ouanes

Soumis à : Amine Badaoui

Instructeur du cours : Frédéric Buteau-Tremblay

Question 1

Phase de reconnaissance

1. Une authentification est requise pour se connecter. Il faudra alors entrer un username et password. Puisque nous ne disposons pas de ces identifiants, il est alors impossible de se connecter vers la session de cette façon.

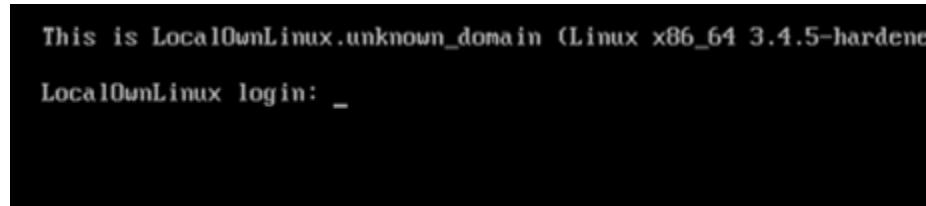


Figure 1 : Démarrage de la machine virtuelle

2. Une authentification est requise pour se connecter (password). On ne peut donc pas y accéder.

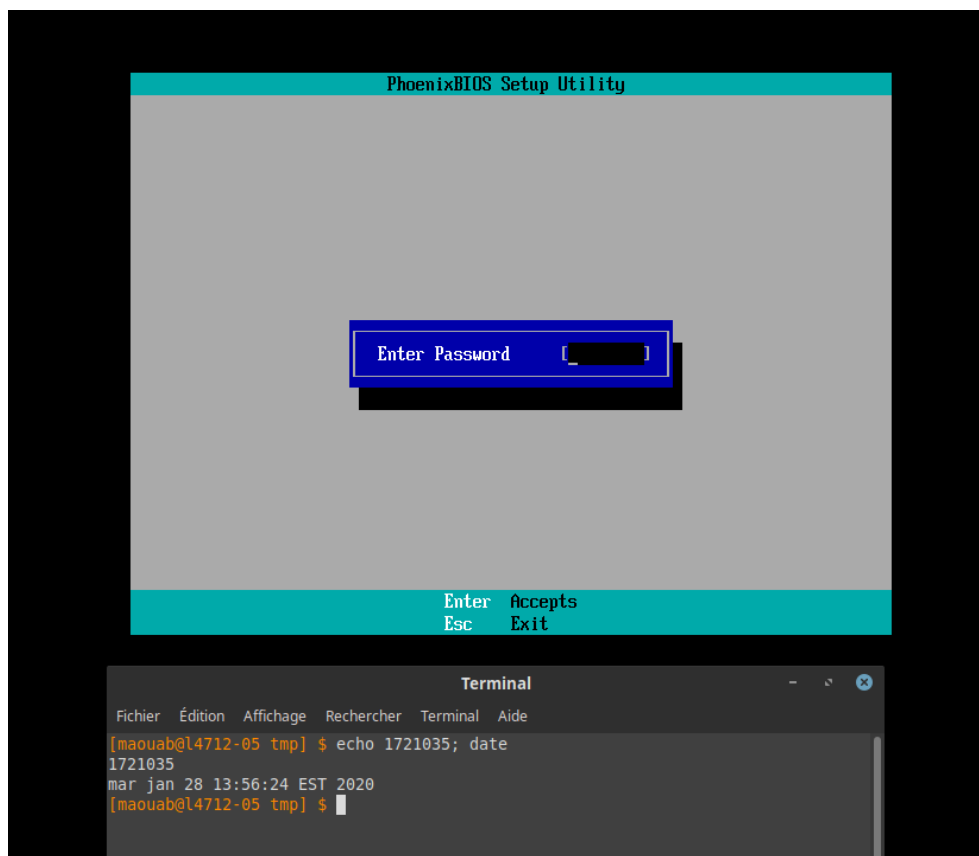


Figure 2 : Page d'authentification

3.

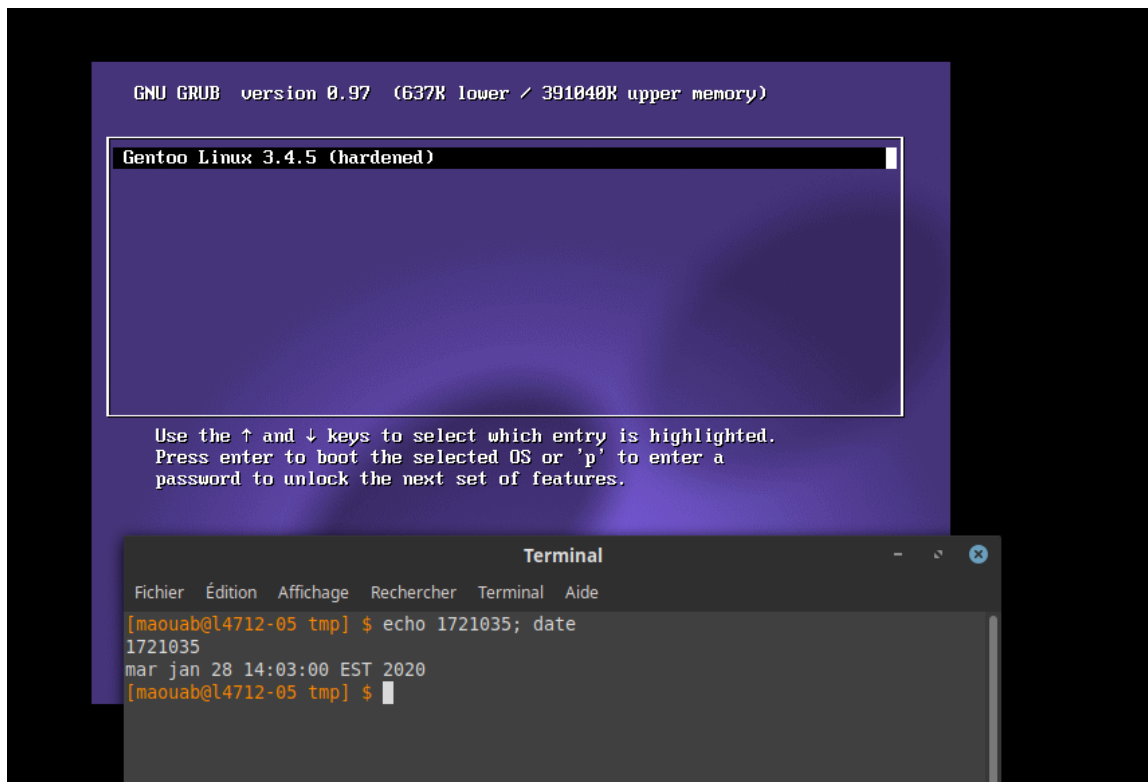


Figure 3 : Menu GRUB

4. Il n'est pas possible d'éditer la ligne de commande correspondante en appuyant sur la touche "e" car comme nous l'avons vu précédemment nous n'avons pas accès ni au bios, ni au systèmes d'exploitation et Grub est protégé par un mot de passe. Ainsi, nous ne sommes pas autorisés à changer les options de BOOT.

Réalisation de l'attaque

Machine LocalOwnLinux

1. Pour commencer, nous devons tout d'abord avoir accès au BIOS afin de booter sur Backtrack. Afin de pouvoir contourner le BIOS, nous devons retirer la pile pendant quelques secondes afin de réinitialiser ce dernier. Pour pouvoir contourner le BIOS et pour Booter sur le lecteur CD-ROM, nous devons d'abord supprimer le fichier *.nvram qui est l'équivalent de retirer la pile pour une VM puis ainsi on pourra redémarrer la VM et rentrer sur BIOS pour faire passer celui-ci vers le lecteur CD-ROM au-dessus du disque dur afin de pouvoir booter sur le lecteur CD-ROM et non sur le disque dur. Les deux figures ci-dessous montres ces deux étapes.

```

[maouab@l4712-05 1721035] $ cd Clone\ of\ LocalOwnLinux/
[maouab@l4712-05 Clone of LocalOwnLinux] $ ls
Clone of LocalOwnLinux.nvram'      Gentoo_amd64_template-cl3-s004.vmdk
Clone of LocalOwnLinux.vmxrest.lck' Gentoo_amd64_template-cl3-s005.vmdk
Clone of LocalOwnLinux.vmsd'       Gentoo_amd64_template-cl3-s006.vmdk
Clone of LocalOwnLinux.vmx'        Gentoo_amd64_template-cl3.vmdk
Clone of LocalOwnLinux.vmx'        Gentoo_amd64_template-cl3.vmdk
Clone of LocalOwnLinux.vmx'        Gentoo_amd64_template-cl3.vmdk
Clone of LocalOwnLinux.vmx'        Gentoo_amd64_template-cl3.vmdk
Clone of LocalOwnLinux.vmx'        Gentoo_amd64_template-cl3.vmdk
Clone of LocalOwnLinux.vmx'        Gentoo_amd64_template-cl3.vmdk
Clone of LocalOwnLinux.vmx'        Gentoo_amd64_template-cl3.vmdk
Gentoo_amd64_template-cl3-s001.vmdk vmware-0.log
Gentoo_amd64_template-cl3-s002.vmdk vmware-1.log
Gentoo_amd64_template-cl3-s003.vmdk vmware-2.log
Gentoo_amd64_template-cl3-s003.vmdk vmware.log
[maouab@l4712-05 Clone of LocalOwnLinux] $ ls *.nvram
Clone of LocalOwnLinux.nvram'
[maouab@l4712-05 Clone of LocalOwnLinux] $ rm Clone\ of\ LocalOwnLinux.nvram
[maouab@l4712-05 Clone of LocalOwnLinux] $ echo 1721035; date
1721035
mar jan 28 14:36:52 EST 2020

```

Figure 4 : Copie de la machine virtuelle

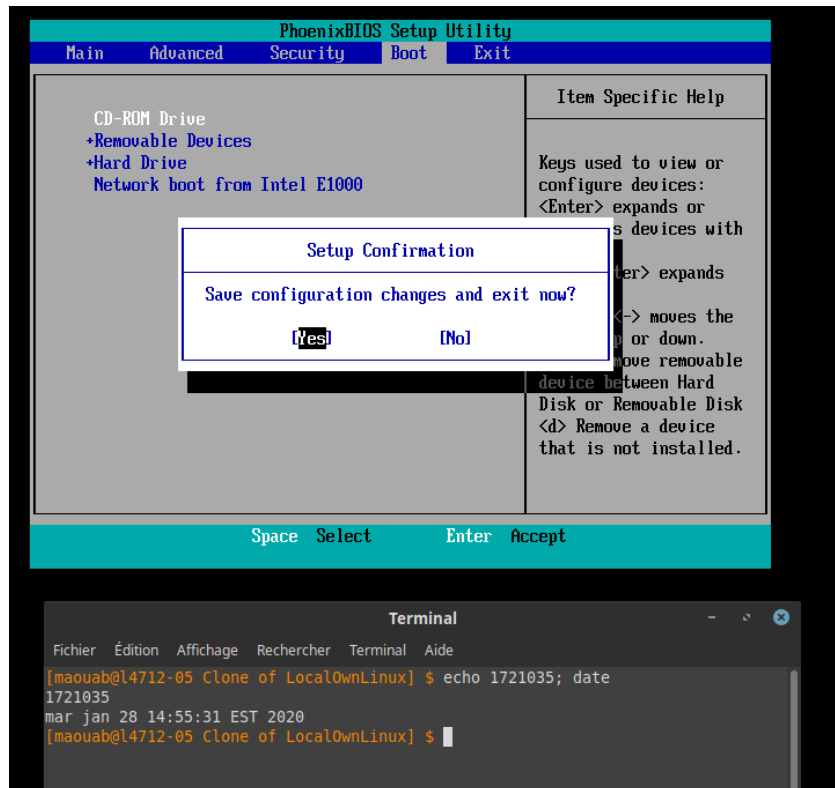


Figure 5 : Accès au BIOS de la machine virtuelle

2. Pour continuer l'attaque, nous devons insérer un livecd de backtrack dans le lecteur CD-ROM afin de pouvoir booter dessus. Pour cela, nous allons placer une image ISO de backtrack sur le lecteur CD de notre VM.

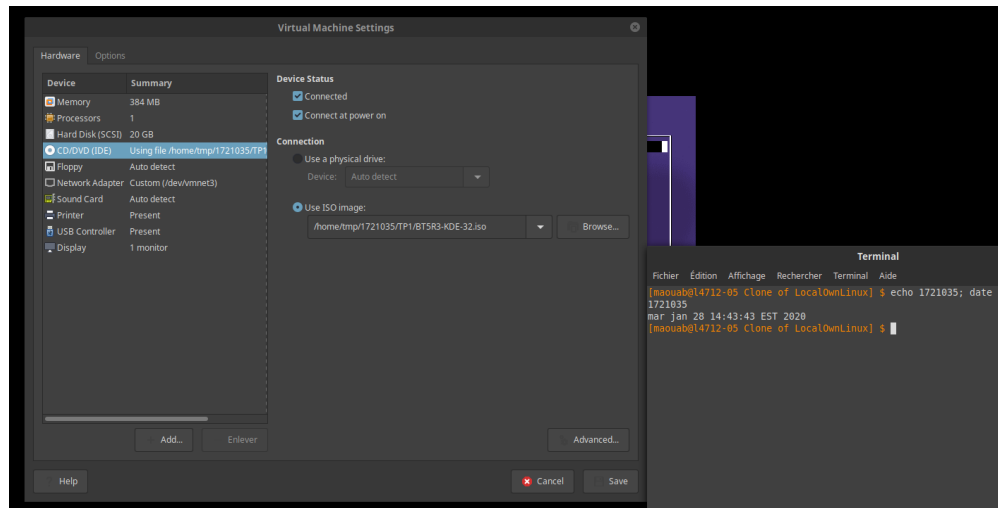


Figure 6 : Réglage de la machine virtuelle pour démarrer à partir d'une image

3. Ensuite, nous devons redémarrer la VM afin de pouvoir "booter" sur Backtrack.

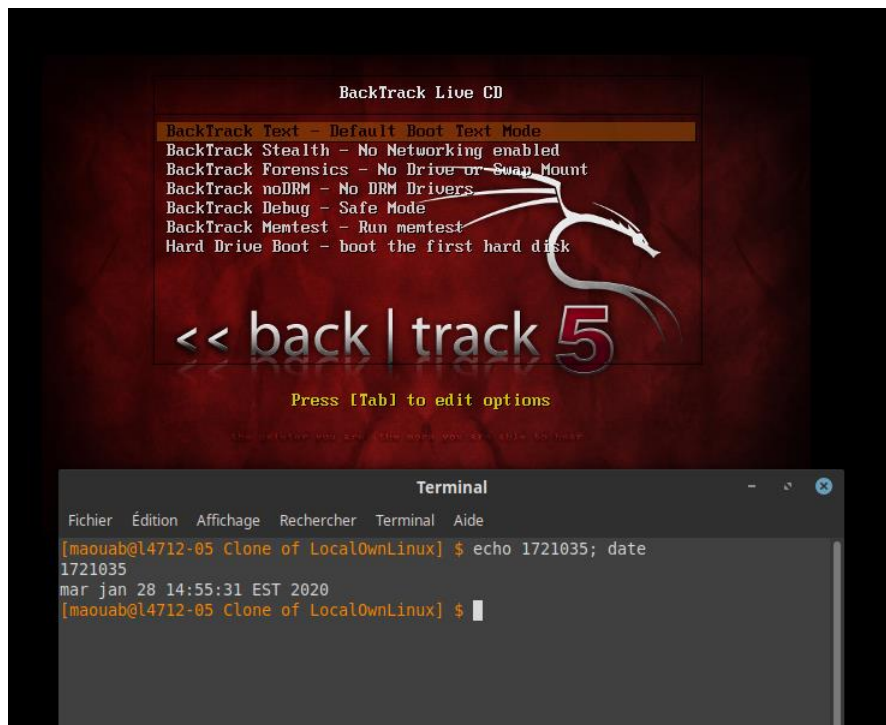
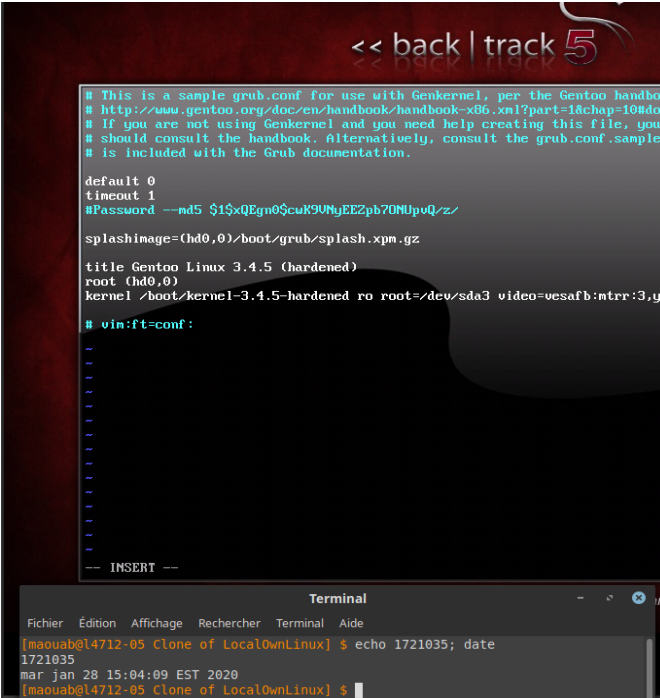


Figure 7 : Démarrage de Backtrack

4. Notre objectif actuellement est de pouvoir obtenir l'accès root sur la machine. Conséquemment, nous allons supprimer le mot de passe de Grub dans le fichier .conf afin de pouvoir faire des modifications sur le lancement du système d'exploitation de la machine. Les étapes à suivre sont les suivantes:
- mkdir /mnt/tmp afin de créer un dossier tmp pour pouvoir stocker les fichiers du disque dur.
 - mount /dev/sda1 /mnt/tmp afin de monter la partition de boot dans le dossier tmp. Cela nous permet l'accès de tous les fichiers du disque dur sda1.
 - Supprimer le password dans le fichier grub.conf.



```
<< back | track 5

# This is a sample grub.conf for use with Genkernel, per the Gentoo handbook
# http://www.gentoo.org/doc/en/handbook/handbook-x86.xml?part=1&chap=10#doc
# If you are not using Genkernel and you need help creating this file, you
# should consult the handbook. Alternatively, consult the grub.conf.sample
# is included with the Grub documentation.

default 0
timeout 1
#Password --md5 $1$xQEgn0$cuK9UHyEEZpb70MUpvQ/z/

splashimage=(hd0,0)/boot/grub/splash.xpm.gz

title Gentoo Linux 3.4.5 (hardened)
root (hd0,0)
kernel /boot/kernel-3.4.5-hardened ro root=/dev/sda3 video=vesafb:ntrr:3,yw

# vim:ft=conf :

-- INSERT --

Terminal
Fichier  Édition  Affichage  Recherche  Terminal  Aide
[maouab@l4712-05 Clone of LocalOwnLinux] $ echo 1721035; date
1721035
mar jan 28 15:04:09 EST 2020
[maouab@l4712-05 Clone of LocalOwnLinux] $
```

Figure 8 : Modification de la configuration GRUB

5. Après, avoir refait passer le disque dur au-dessus du lecteur CD-ROM dans le BIOS nous relançons la VM et arrivons sur Grub ou nous pouvons maintenant appuyer sur “e” pour éditer la commande.

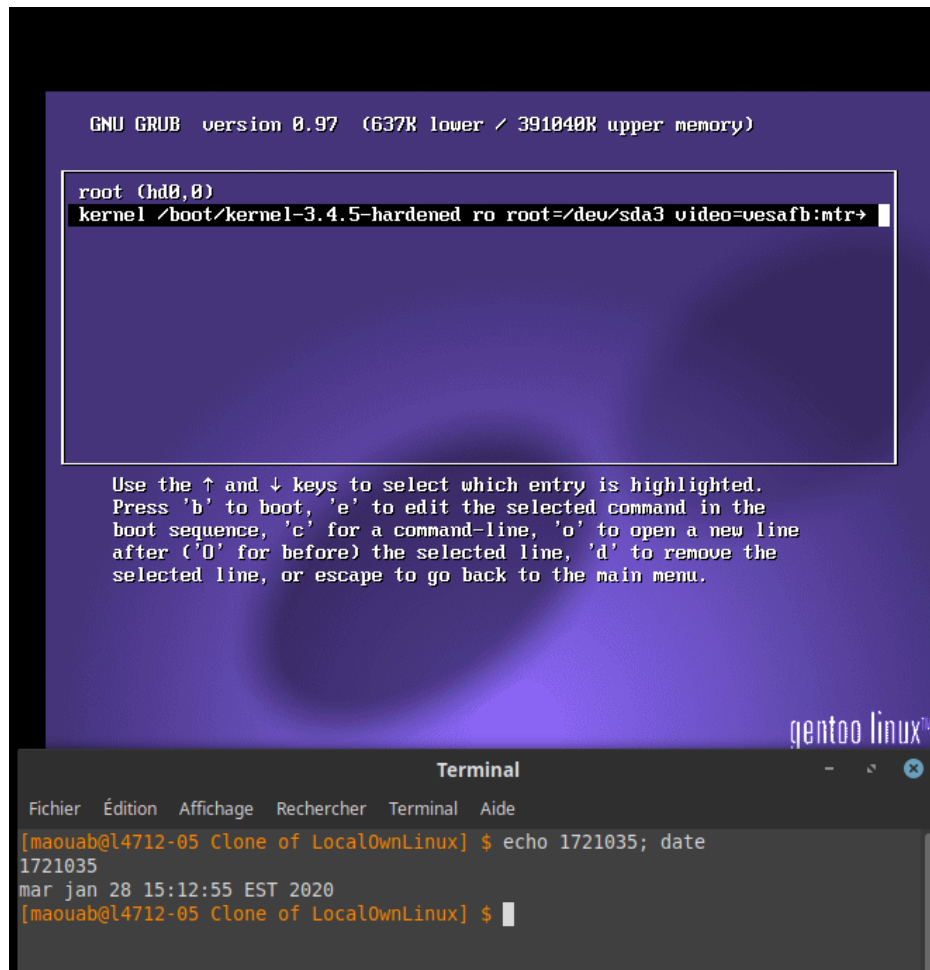


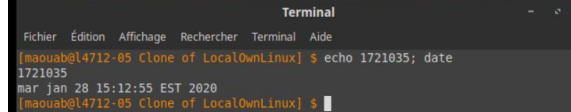
Figure 9 : Accès à l'éditeur sur GRUB

Nous allons maintenant avoir accès aux configurations du kernel. Lorsque l'ordinateur démarre, il exécute un programme appelé "init", généralement trouvé dans /bin/init ou /sbin/init. Ce programme est responsable de tout le démarrage du système et de la création d'un environnement utilisable. Spécifier init = /bin/bash indique au Kernel d'exécuter /bin/bash à la place(qui est un shell), nous donnant ainsi accès à la machine.

```
[ Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename. ESC at any time exits. ]

Co=vesafb; mtrr=3,ywrap oga=Bx341 init=/bin/bash

gentoo linux
```



```
Terminal
Fichier Édition Affichage Rechercher Terminal Aide
[maouab@l4712-05 Clone of LocalOwnLinux] $ echo 1721035; date
1721035
mar jan 28 15:12:55 EST 2020
[maouab@l4712-05 Clone of LocalOwnLinux] $
```

Figure 10 : Modification de la commande à exécuter

6. Une fois appuyé sur “b” pour boot nous avons accès à un shell.

```
[ 1.634975] generic-usb 0003:0E0F:0003.0002: input,hidraw1: USB HID v1.10 Mouse [VMware U
:02:00.0-1/input1
[ 1.739829] usb 2-2: new full-speed USB device number 3 using uhci_hcd
[ 1.868264] usb 2-2: New USB device found, idVendor=0e0f, idProduct=0002
[ 1.872593] usb 2-2: New USB device strings: Mfr=0, Product=1, SerialNumber=0
[ 1.874671] usb 2-2: Product: VMware Virtual USB Hub
[ 1.882142] hub 2-2:1.0: USB hub found
[ 1.888901] hub 2-2:1.0: 7 ports detected

(none) / # mount -o remount,rw /_

Terminal
Fichier Édition Affichage Rechercher Terminal Aide
[maouab@l4712-05 Clone of LocalOwnLinux] $ echo 1721035; date
1721035
mar jan 28 15:12:55 EST 2020
```

Figure 11 : Montage de la partition

Et nous remontons la partition root avec tous les droits afin de pouvoir changer le password de root.

```
(none) / # passwd
New password:
BAD PASSWORD: it is too short
BAD PASSWORD: is too simple
Retype new password:
passwd: password updated successfully
(none) / #
```

Figure 12 : Changement du mot de passe root

7. Finalement nous redémarrons en root !

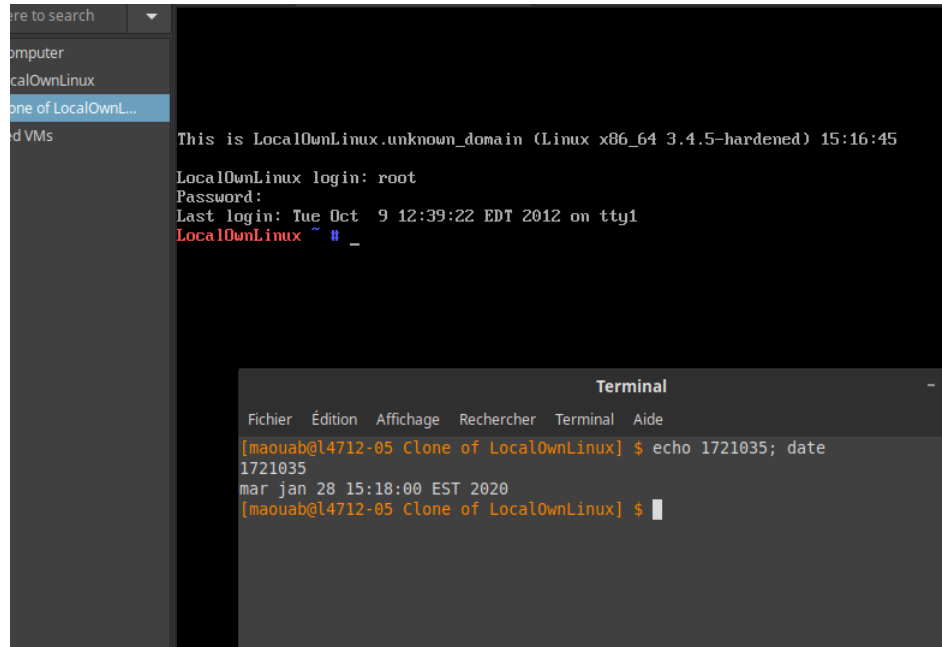


Figure 13 : Accès à la session root

Machine LocalOwnWin

Dans cette partie, nous entamerons la même procédure a quelques différences prêtes sur une VM windows.

1. Cette partie de sur la phase de reconnaissance est la même que celle que nous avons présenté précédemment pour la machine Linux.

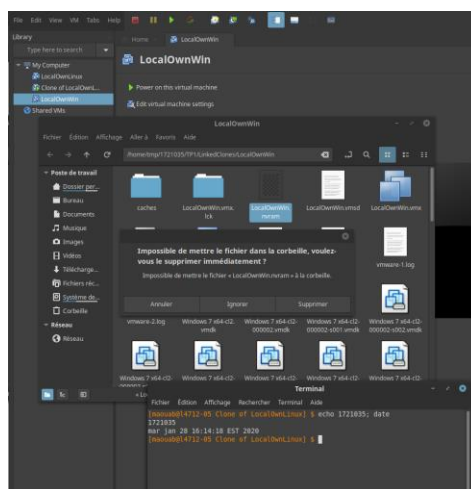


Figure 14 : Copie de la machine virtuelle

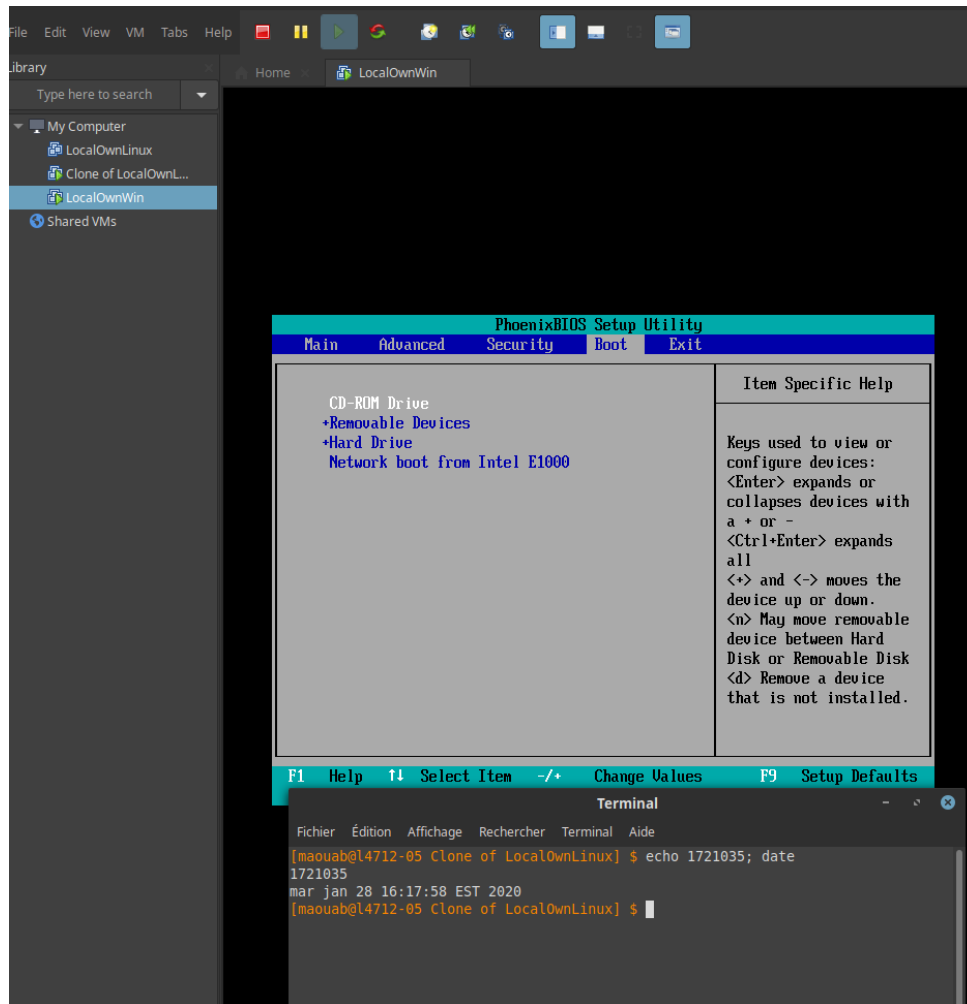


Figure 15 : Accès au BIOS de la machine virtuelle

2. Nous démarrons l'interface graphique de Backtrack.

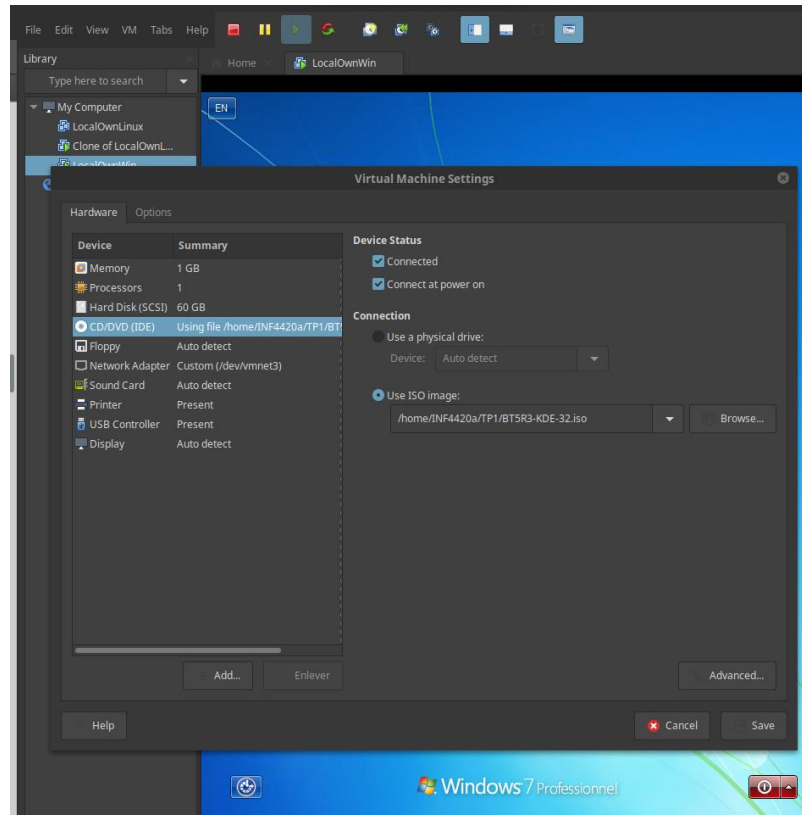


Figure 16 : Configuration de la machine virtuelle pour démarrer Backtrack

On répond "yes" à si une fenêtre s'ouvre concernant la carte son.

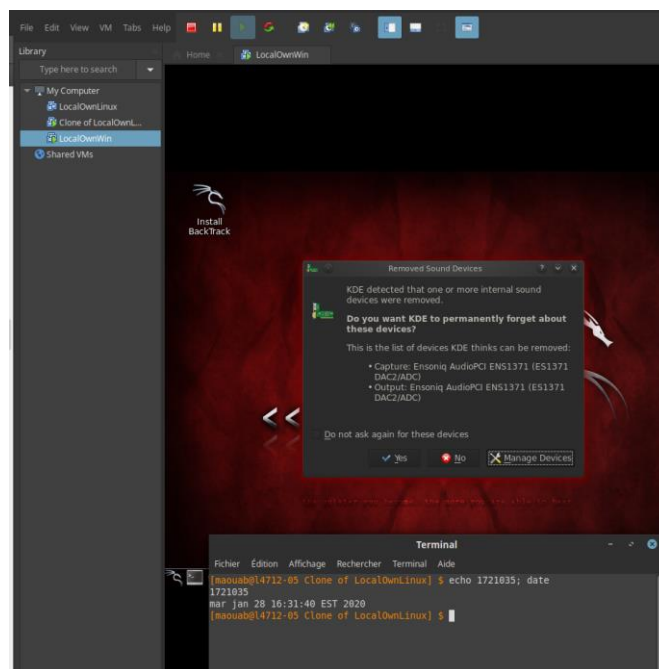


Figure 17 : Démarrage de Backtrack en mode interface graphique

3. N/A
4. La partition est montée.

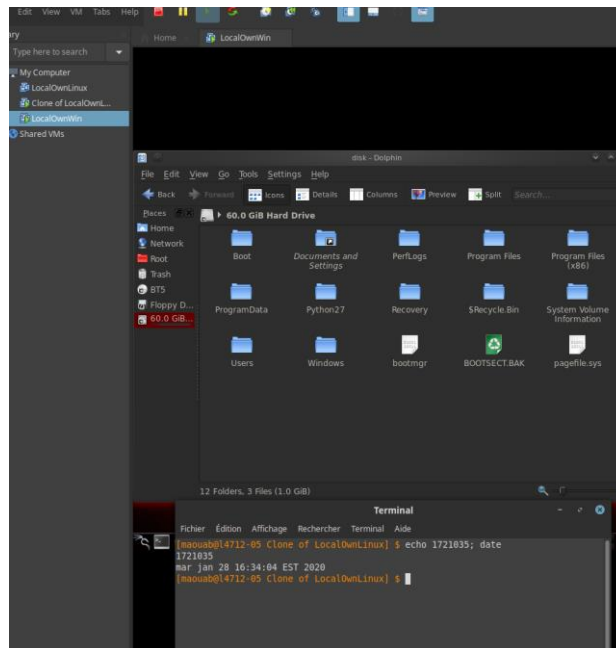


Figure 18 : Montage de la partition sur Backtrack

5. On part le programme chntpw comme on peut le voir dans la prochaine figure (offline attacks puis on sélectionne chntpw).

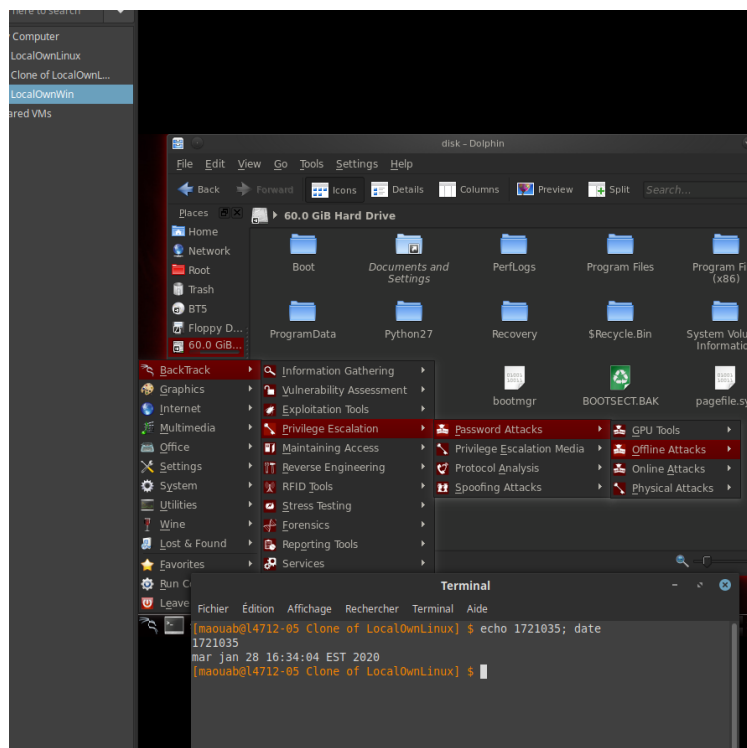


Figure 19 : Démarrage de l'outil d'attaque

6. Le programme chntpw est exécuté avec le fichier SAM comme entrée. L'option i est utilisée pour lister tous les utilisateurs. On utilise le programme pour effacer le mot de passe du compte administrateur.

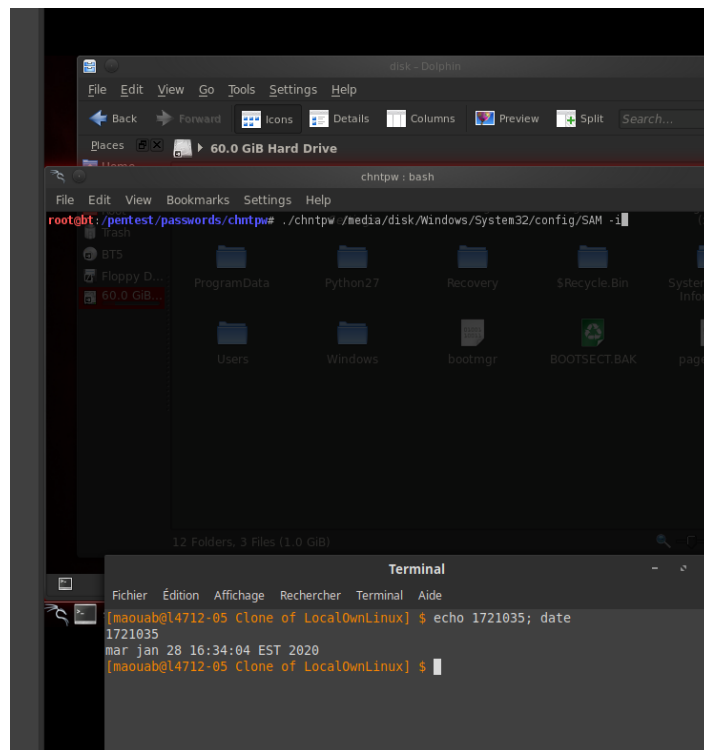


Figure 20 : Chargement du fichier SAM pour modification

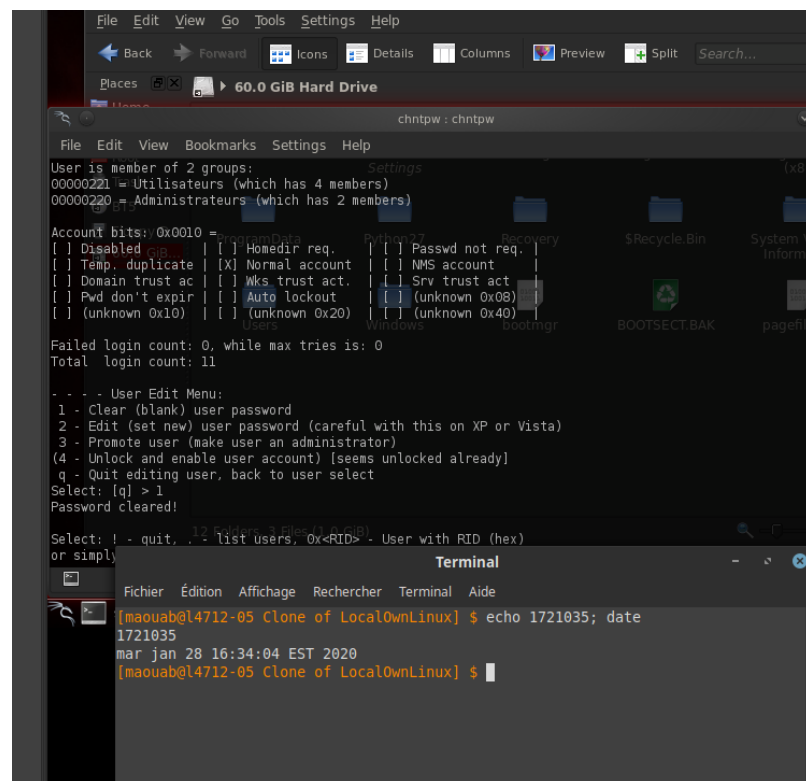


Figure 21 : Modification du fichier SAM

7. Le SAM (Security Account Manager) est une base de données locale présente sur certaines anciennes machine Windows. Il sert à contenir les mots de passes locaux de la machine
8. On utilise le programme pour effacer le mot de passe du compte administrateur.
9. On redémarre ensuite la VM en passant par le BIOS pour remettre les priorités de boot du système comme elle étaient initialement, avec le disque dur en premier. On a alors accès au compte admin sans entrer de mot de passe.

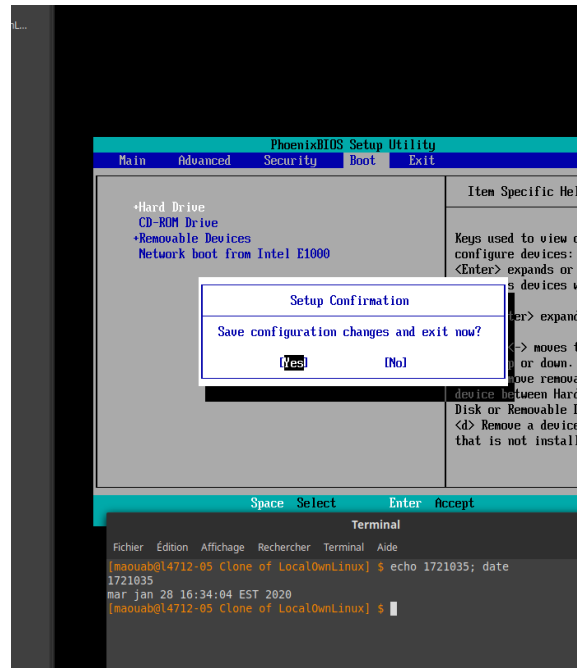


Figure 22 : Modification du BIOS

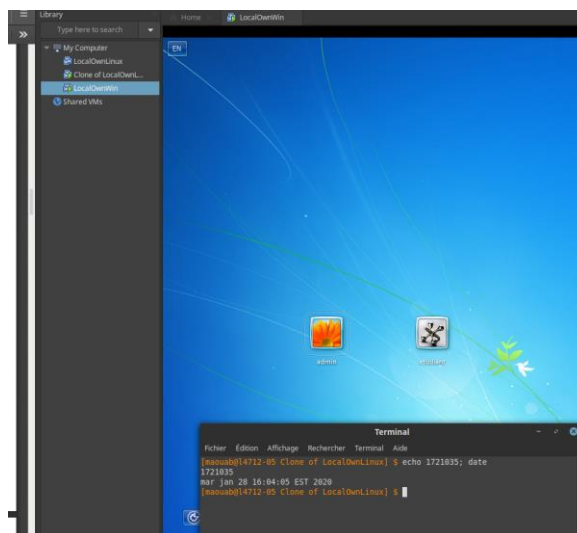
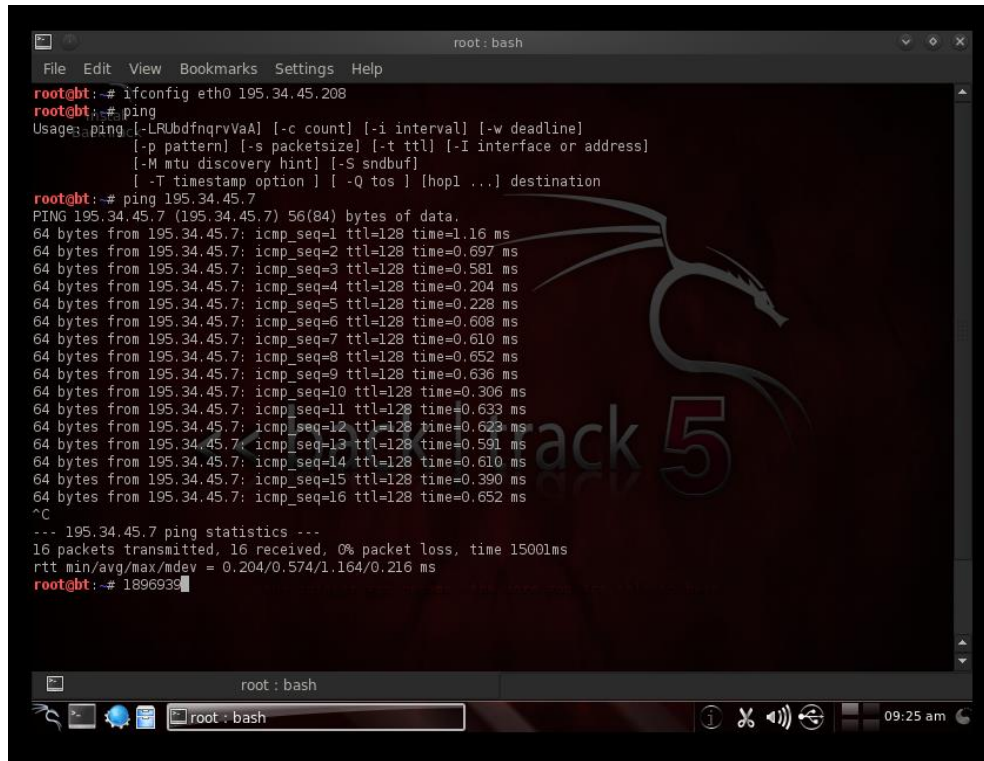


Figure 23 : Connexion au compte admin

Question 2

Phase de reconnaissance

2.

A screenshot of a terminal window titled 'root: bash'. The terminal shows the following commands and output:

```
root@bt:~# ifconfig eth0 195.34.45.208
root@bt:~# ping
Usage: ping [-LRUbfqnrvVaA] [-c count] [-i interval] [-w deadline]
          [-p pattern] [-s packetsize] [-t ttl] [-I interface or address]
          [-M mtu discovery hint] [-S sndbuf]
          [-T timestamp option] [-Q tos] [hop1 ...] destination
root@bt:~# ping 195.34.45.7
PING 195.34.45.7 (195.34.45.7) 56(84) bytes of data.
64 bytes from 195.34.45.7: icmp_seq=1 ttl=128 time=1.16 ms
64 bytes from 195.34.45.7: icmp_seq=2 ttl=128 time=0.697 ms
64 bytes from 195.34.45.7: icmp_seq=3 ttl=128 time=0.581 ms
64 bytes from 195.34.45.7: icmp_seq=4 ttl=128 time=0.204 ms
64 bytes from 195.34.45.7: icmp_seq=5 ttl=128 time=0.228 ms
64 bytes from 195.34.45.7: icmp_seq=6 ttl=128 time=0.608 ms
64 bytes from 195.34.45.7: icmp_seq=7 ttl=128 time=0.610 ms
64 bytes from 195.34.45.7: icmp_seq=8 ttl=128 time=0.652 ms
64 bytes from 195.34.45.7: icmp_seq=9 ttl=128 time=0.636 ms
64 bytes from 195.34.45.7: icmp_seq=10 ttl=128 time=0.306 ms
64 bytes from 195.34.45.7: icmp_seq=11 ttl=128 time=0.633 ms
64 bytes from 195.34.45.7: icmp_seq=12 ttl=128 time=0.623 ms
64 bytes from 195.34.45.7: icmp_seq=13 ttl=128 time=0.591 ms
64 bytes from 195.34.45.7: icmp_seq=14 ttl=128 time=0.610 ms
64 bytes from 195.34.45.7: icmp_seq=15 ttl=128 time=0.390 ms
64 bytes from 195.34.45.7: icmp_seq=16 ttl=128 time=0.652 ms
^C
--- 195.34.45.7 ping statistics ---
16 packets transmitted, 16 received, 0% packet loss, time 1500lms
rtt min/avg/max/mdev = 0.204/0.574/1.164/0.216 ms
root@bt:~#
```

The terminal window has a dark background with a large, stylized 'ack 5' watermark. The bottom of the window shows a taskbar with icons for a file manager, terminal, and system status, along with the time '09:25 am'.

Figure 24 : Configuration du réseau

4. Puisque la machine virtuelle n'est pas connectée à un routeur, il faut spécifier l'adresse publique pour pouvoir se connecter aux autres machines virtuelles. Le routeur se chargerait de faire la redirection du ping vers la machine de l'autre sous-réseau.

7.

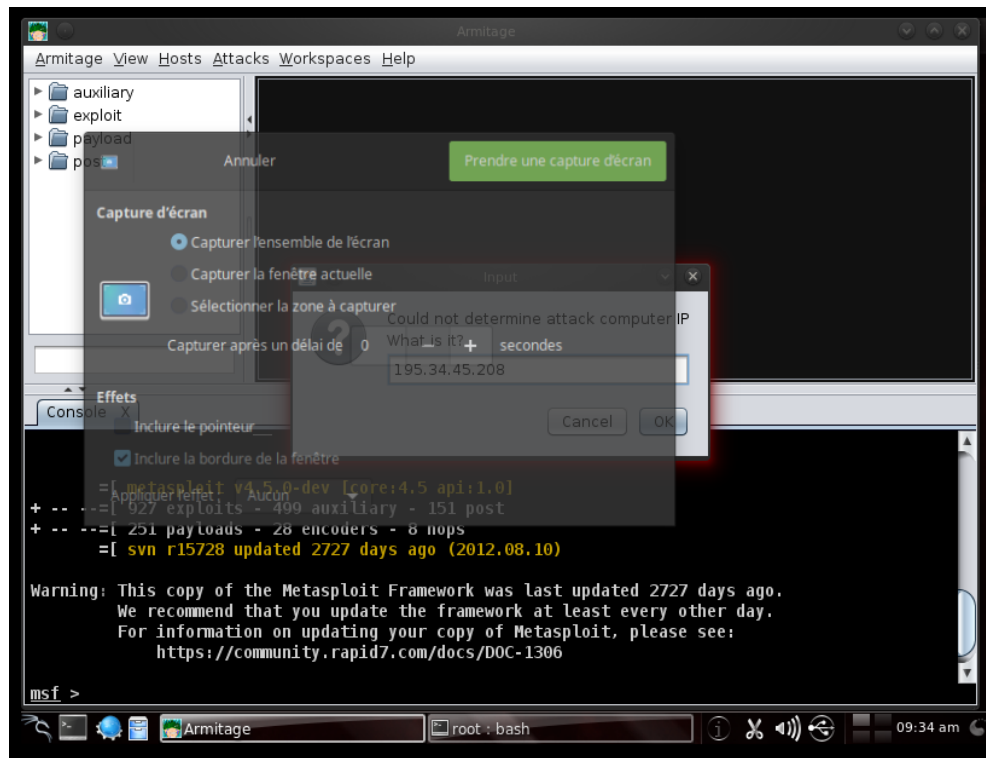


Figure 25 : Démarrage de l'outil d'attaque

8.

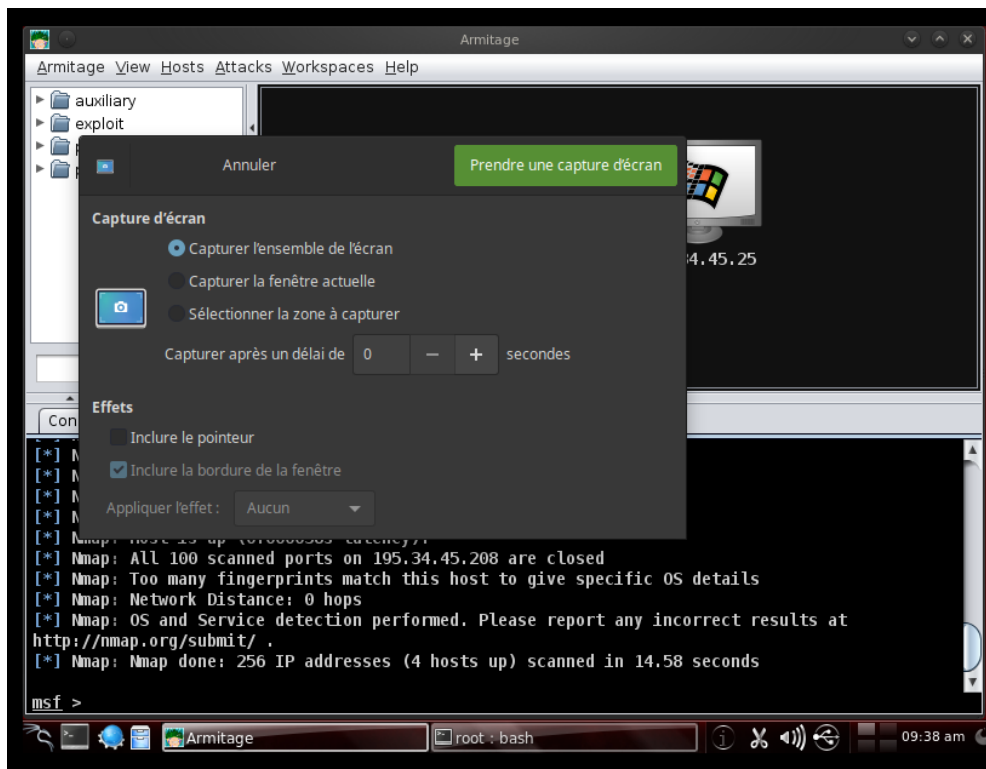


Figure 26 : Scan des machines accessible sur le réseau

9. Nmap fait une carte du réseau pour pouvoir voir quelles machines sont connectés sur celui-ci. Dans notre cas, Nmap nous permet de trouver l'adresse IP ainsi que le système d'exploitation des autres machines virtuelles.

Exploitation des failles de sécurité connues

1.

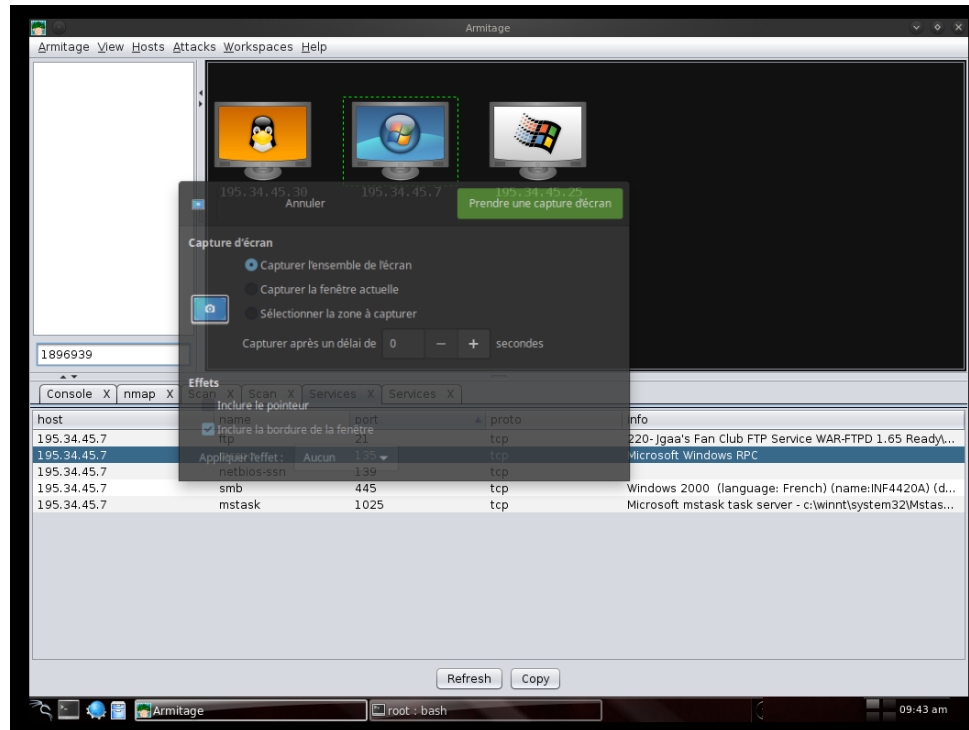


Figure 27 : Recherche du processus rpcdcom

2. Nous avons gagné un accès partiel à la machine. Meterpreter nous donne maintenant accès à différents menus et une des options nous permet d'exécuter des commandes dans l'invite de commande.

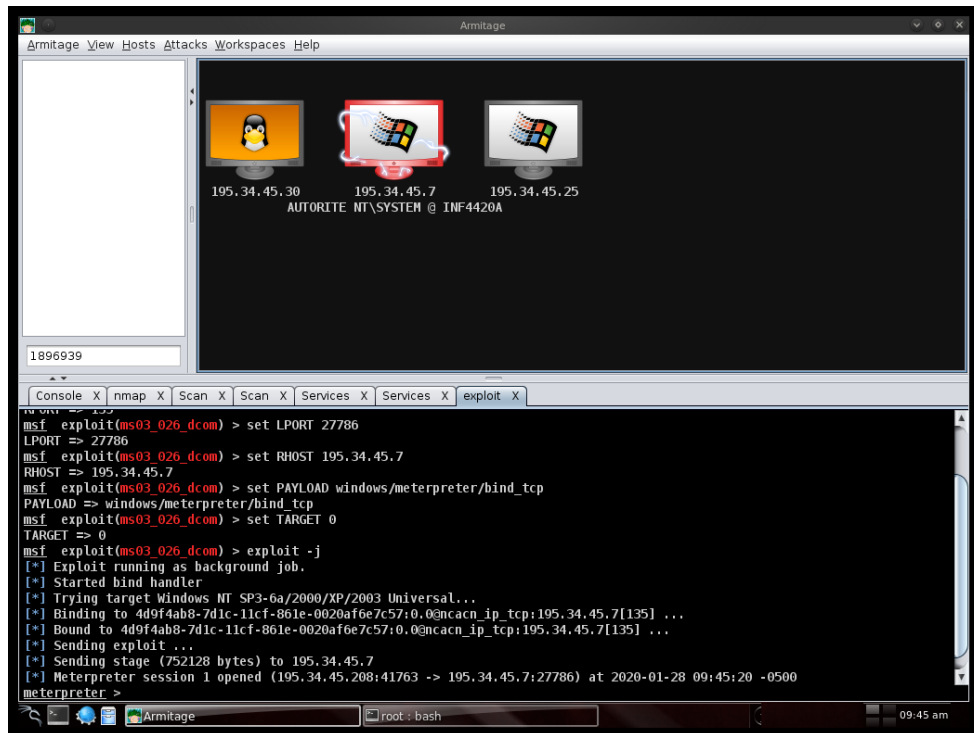


Figure 28 : Gain de contrôle sur une machine

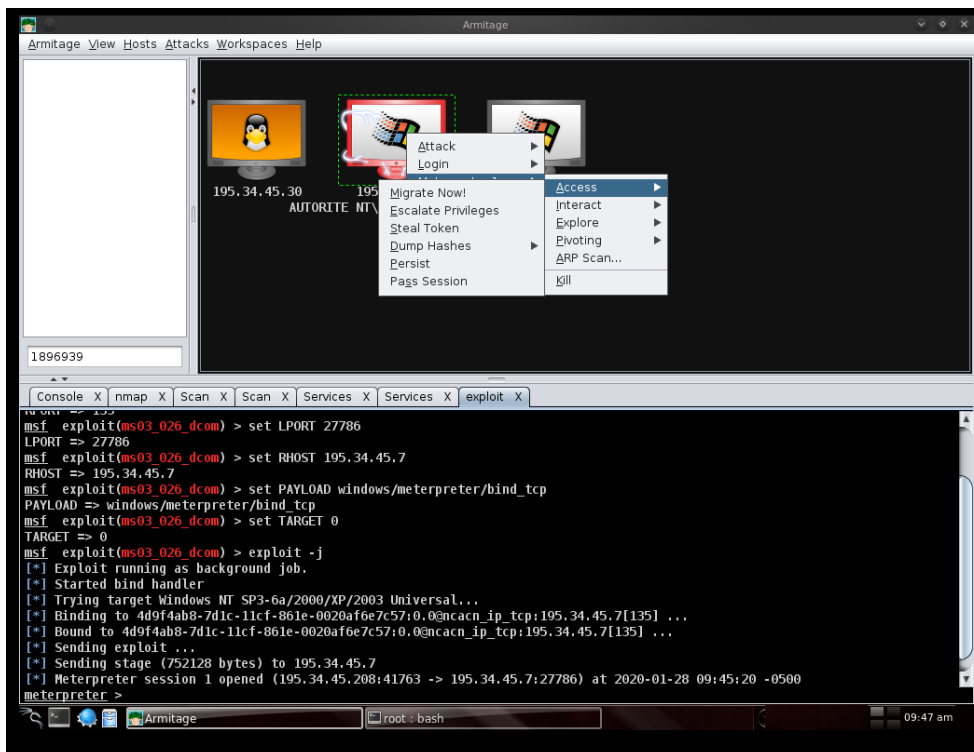


Figure 29 : Démarrage de l'attaque sur la machine

- À l'aide de l'invite de commande fournie par Meterpreter, nous avons simplement créé un nouveau compte utilisateur à l'aide de la commande « net user h4x0r toto /add ». Ensuite, nous nous sommes servi de l'outil Browse Files de Meterpreter pour naviguer au bon répertoire et simplement créer un dossier.

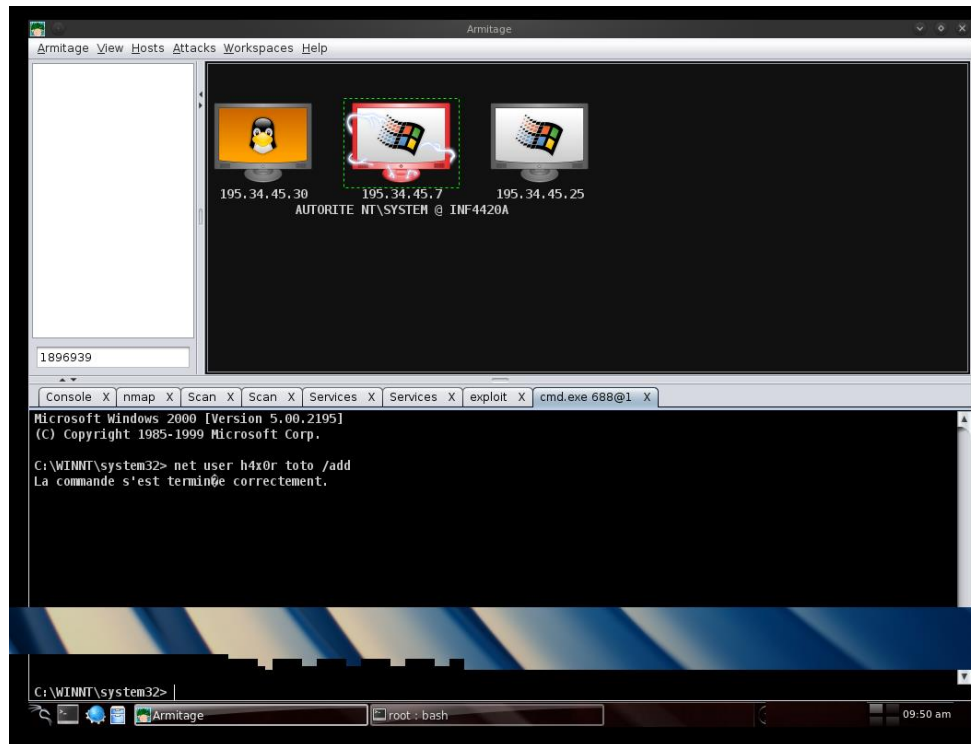


Figure 30 : Ajout d'un utilisateur sur la machine

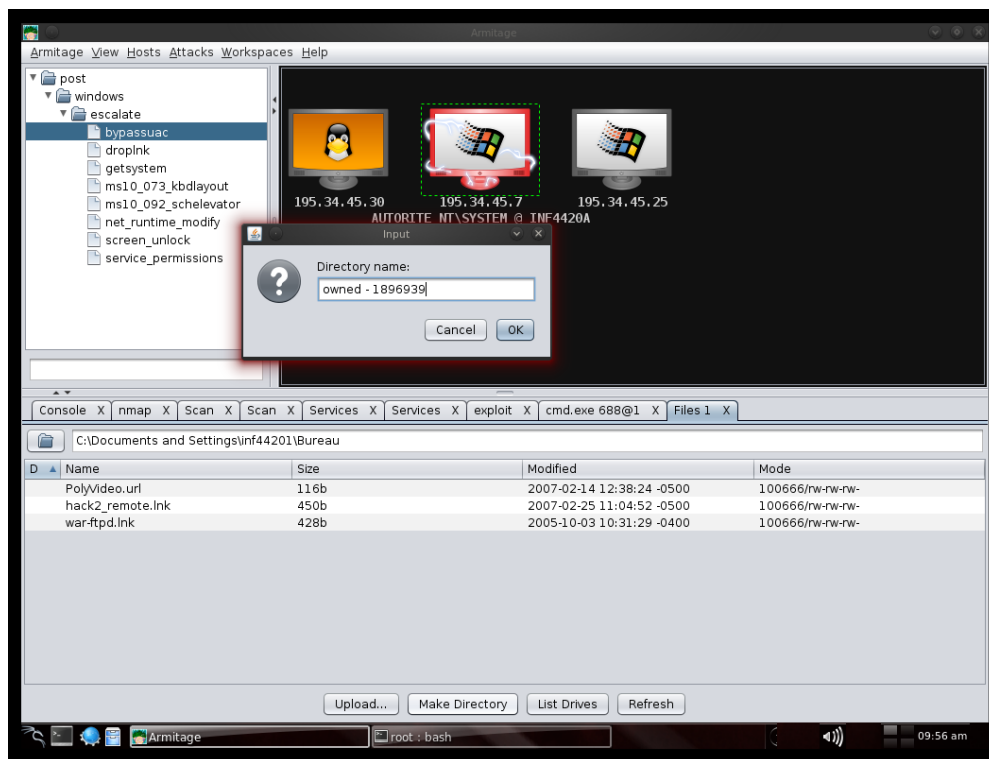


Figure 31 : Ajout d'un répertoire sur la machine

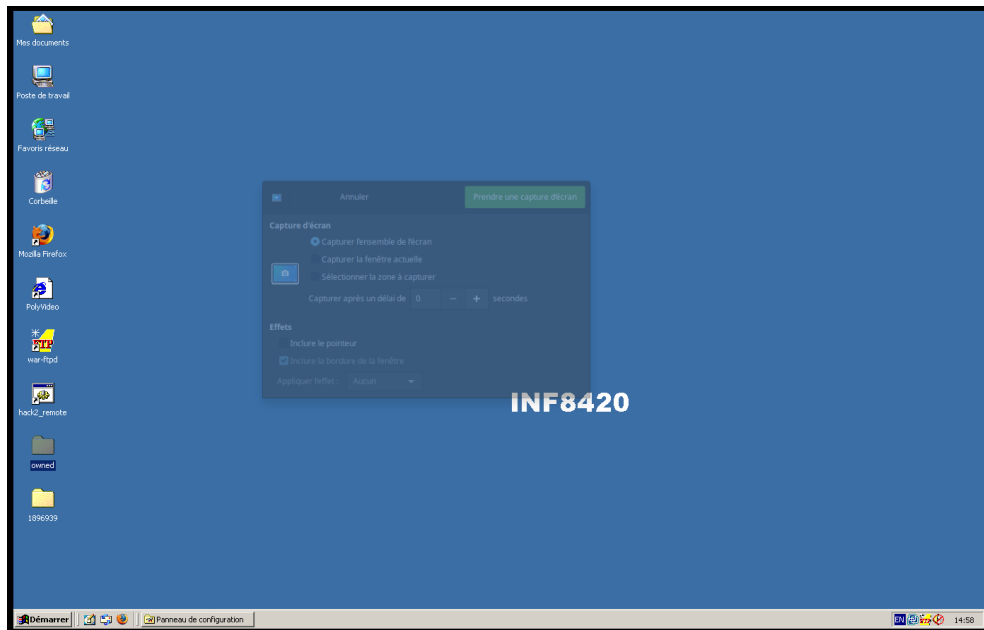


Figure 32 : Confirmation de l'ajout du répertoire

4. Le nom du module utilisé est warftpd_165_pass. Celui-ci nous a permis d'avoir accès un nouvel accès à Meterpreter. Meterpreter nous a ensuite permis de se servir de Browse Files pour créer, comme à la question précédente, un nouveau dossier.

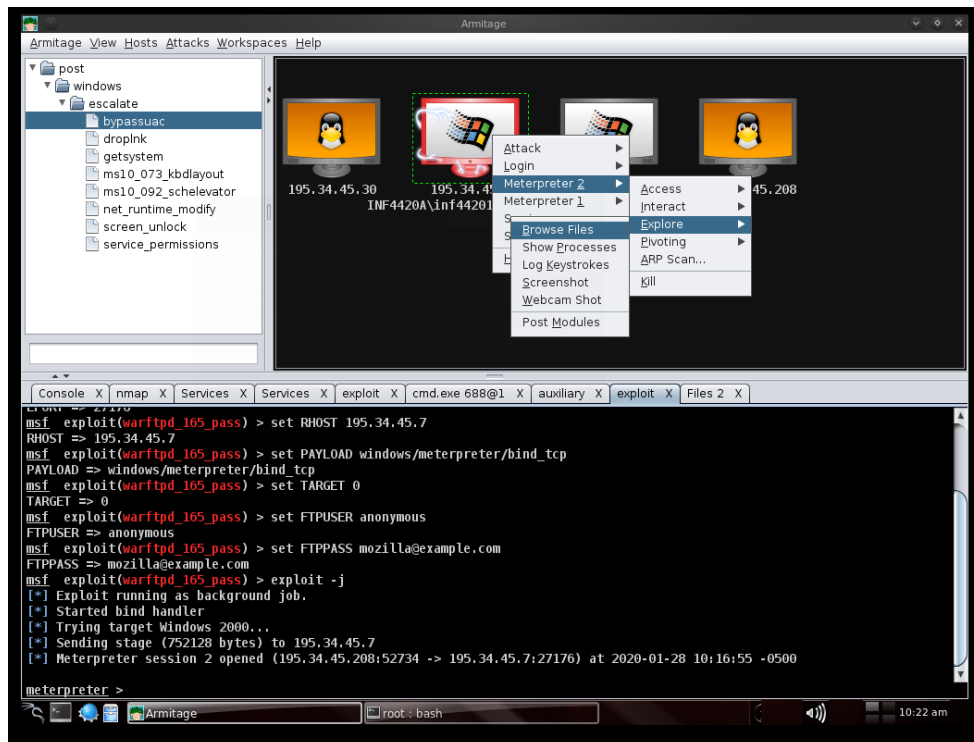


Figure 33 : Recherche d'une autre faille sur la machine

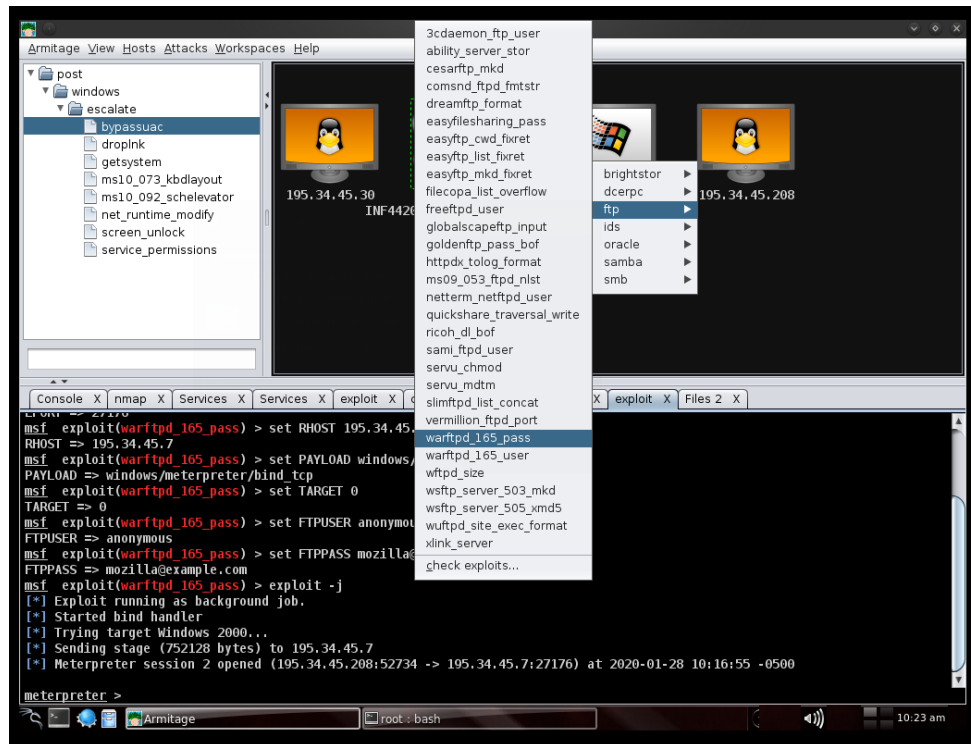


Figure 34 : Découverte du module warftpd_165_pass

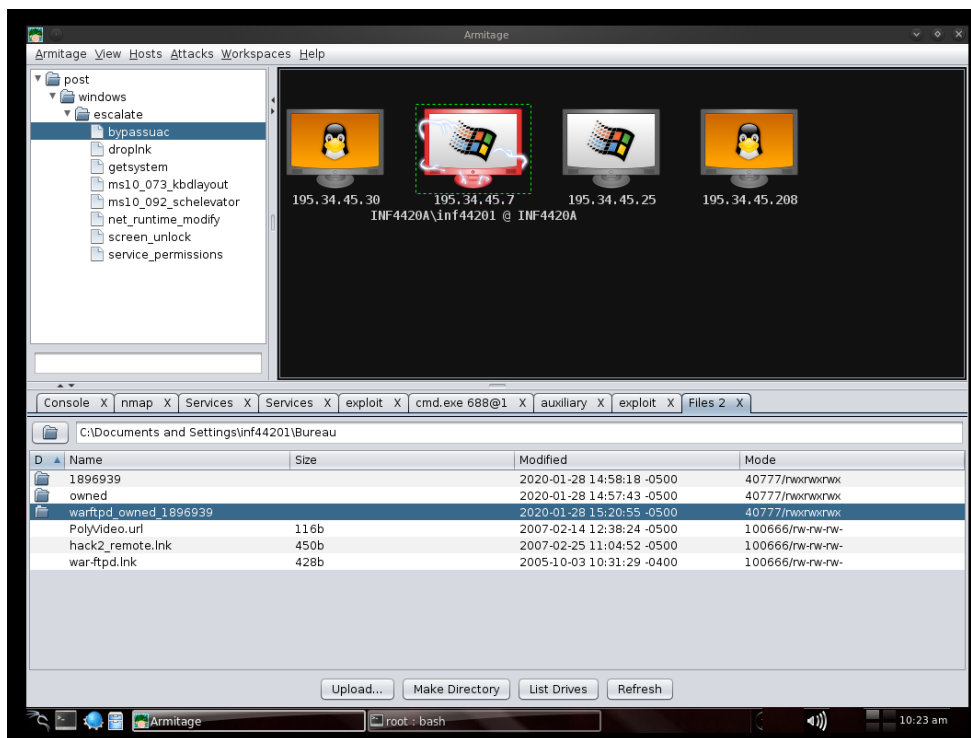


Figure 35 : Utilisation du module

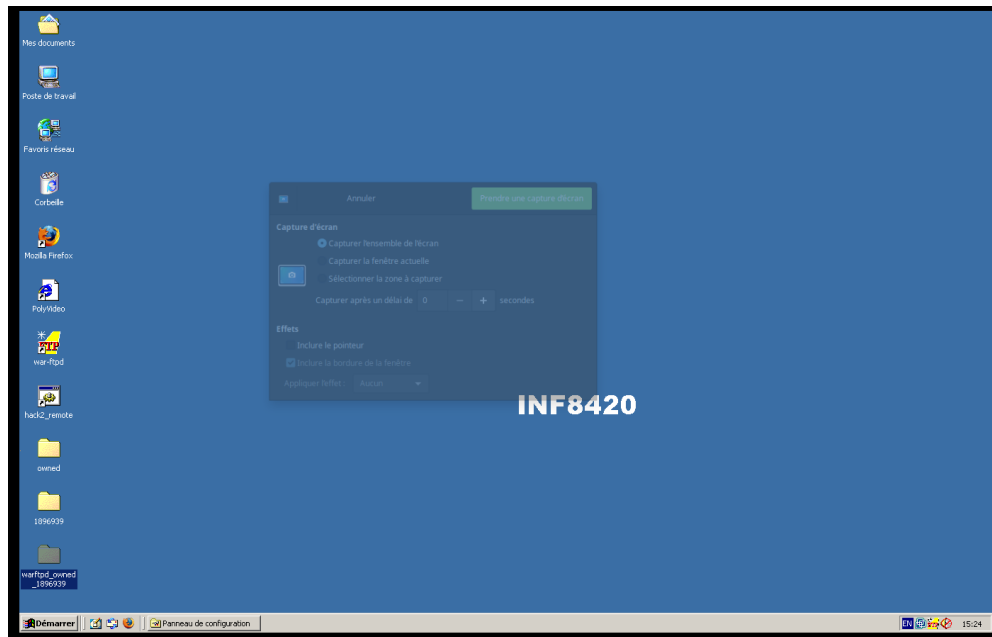


Figure 36 : Confirmation de l'exécution de l'attaque

5.

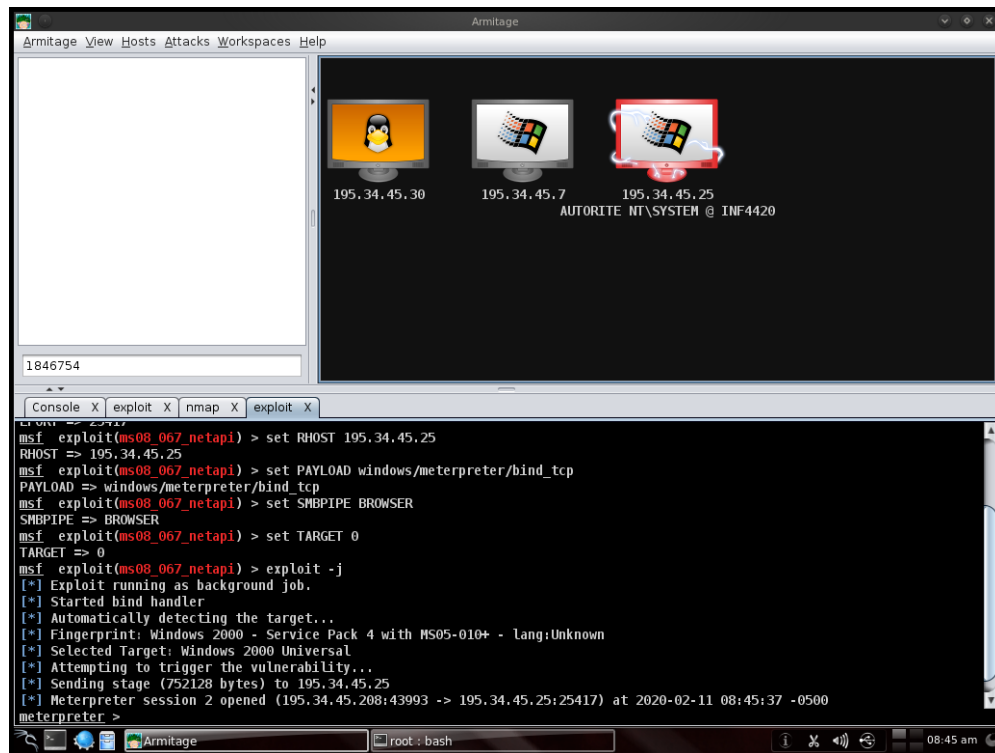


Figure 37 : Accès à une autre machine

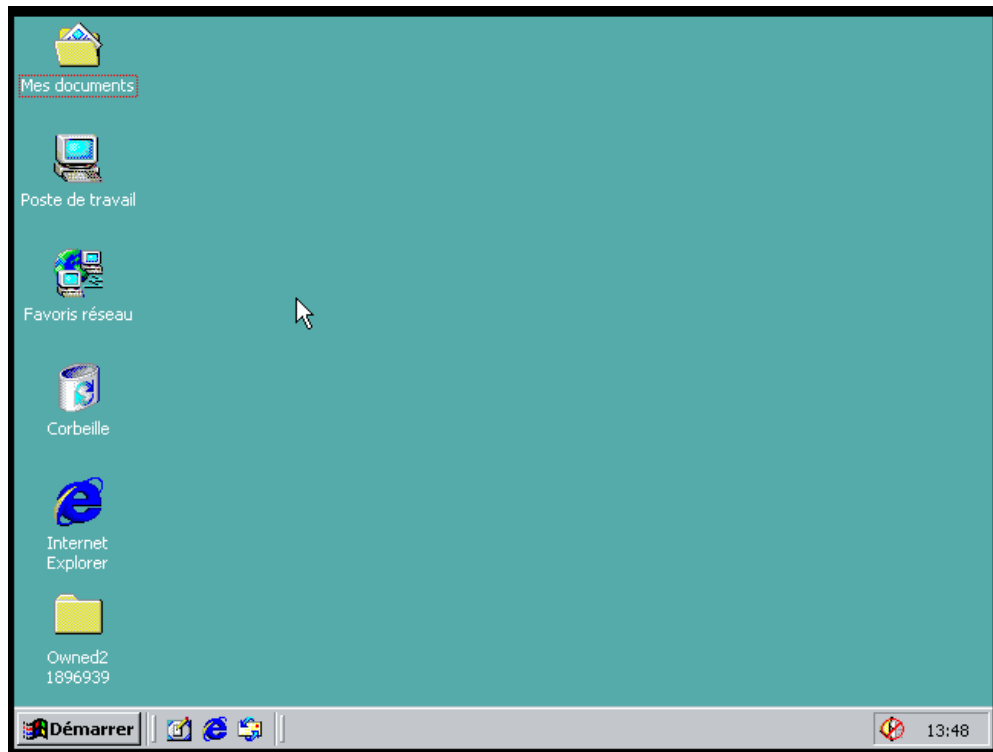


Figure 38 : Confirmation d'exécution de l'attaque

6.

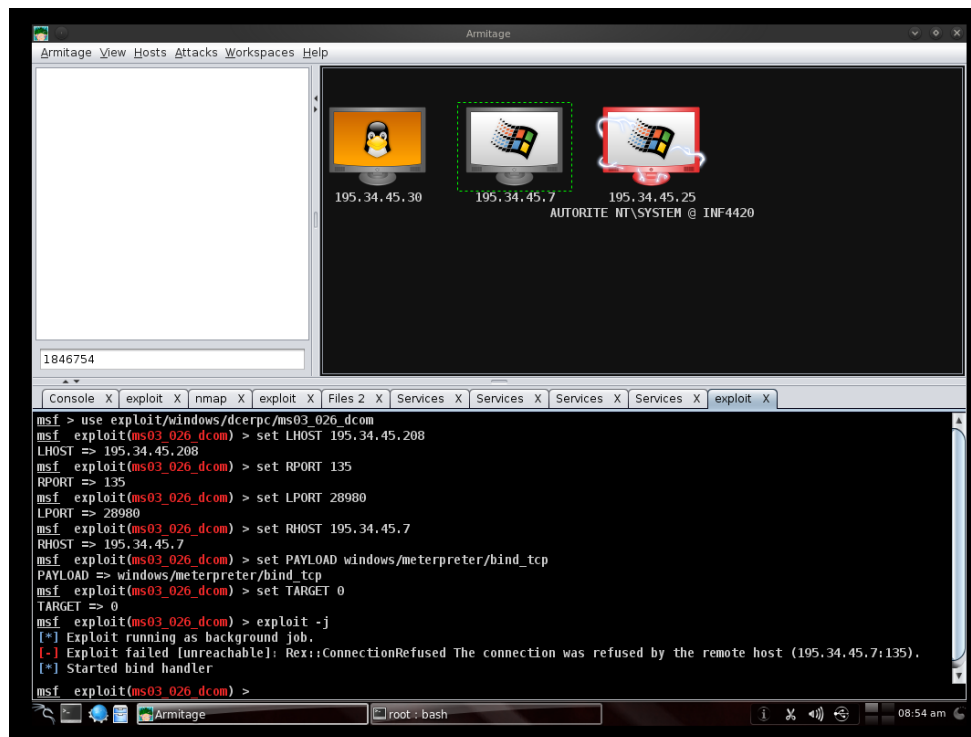


Figure 39 : Exécution de l'attaque

7.

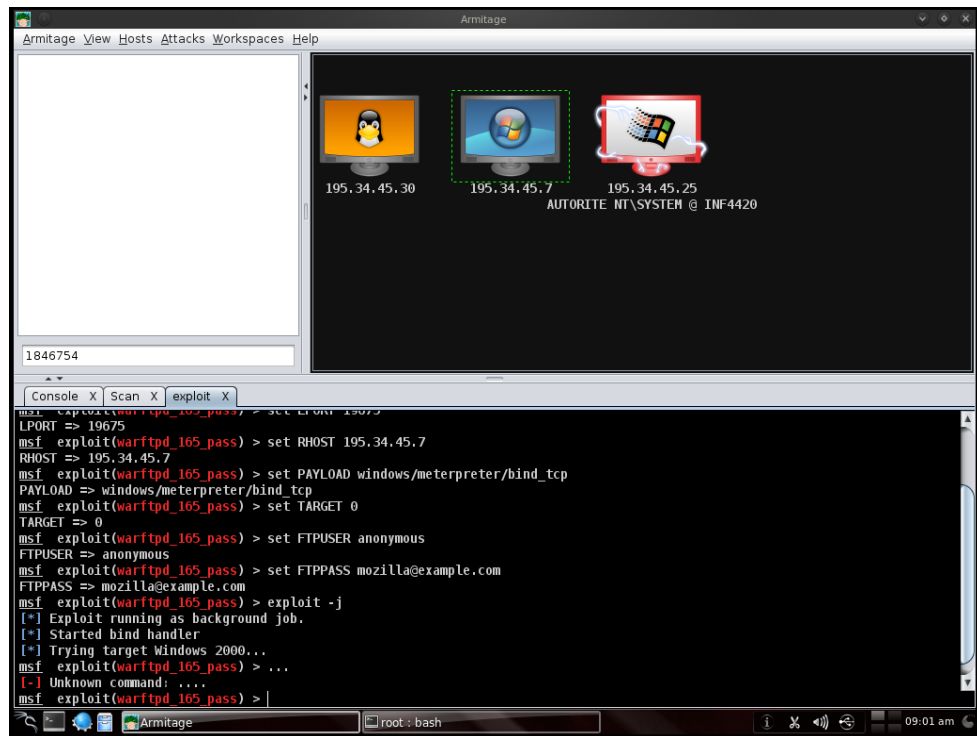


Figure 40 : Confirmation de la non-fonctionnalité de l'attaque suite à la mise à jour

Question 3

1. La connexion d'un utilisateur requiert un nom d'utilisateur et un mot de passe. Aussitôt ces informations sont entrées et soumises pour une tentative de connexion, une requête de type SQL est exécutée afin de rechercher le nom d'utilisateur et le mot de passe dans la base de données. Voici un format générique du type de requête SQL: "**\$req = "select mem_codefrom MEMBRES wheremem_login = '\$login' and mem_pwd = '\$pass'";**". Celle-ci recherche dans la table des membres, le nom login, et le mot de passe, pass. Si la paire de données est trouvée, l'utilisateur est alors authentifié. Une méthode qu'on peut utiliser afin de contourner ce processus permettant de se logger est de contourner la recherche dans la base de données. En effet, en injectant du code SQL dans les champs permettant d'entrer le nom d'utilisateur et mot de passe, on peut faire ce qu'on appelle de l'injection de SQL (SQLi). Dans le cas de cette question, le nom d'utilisateur nous est déjà fourni: **gigi**. Afin de pouvoir contourner le mot de passe, nous pouvons entrer dans le champ mot de passe la condition **mem_pwd = 'or'1=1**. Cette condition est toujours vraie, donc la vérification du mot de passe ne se produit tout simplement pas. Ainsi, on peut se connecter avec le compte du membre gigi. Évidemment, cette méthode ne fonctionne que si aucun système de permettant de se protéger de cette condition SQLi n'est implanté.

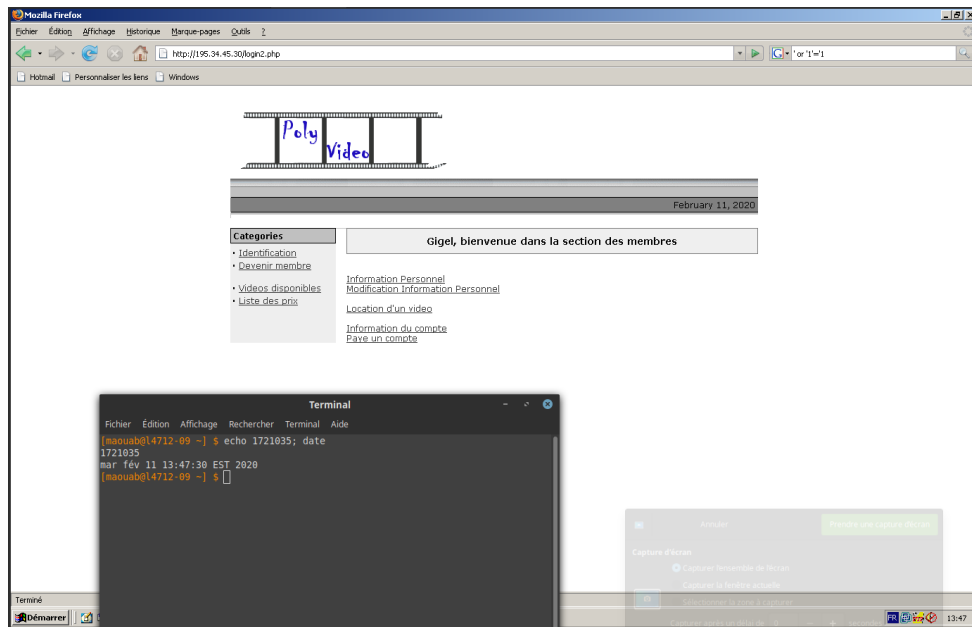


Figure 41 : Confirmation de l'exécution de l'attaque

2. Encore une fois, nous devons utiliser de l'injection SQL (SQLi) afin de contourner le fait de ne posséder ni nom d'utilisateur et de mot de passe. 1. La requête SQL **\$req = "select mem_codefrom MEMBRES wheremem_login = '\$login' and mem_pwd = '\$pass'"** qui communique avec la base de données démontre qu'il faut contourner les vérifications de nom d'utilisateur et de mot de passe. Ainsi, il suffit de mettre **'or'1=1** dans le champ login et password pour se connecter. Comme ceci est expliqué précédemment a

query 'or'1=1 dans le champ Login interroge en fait tous les utilisateurs de la base de données et contourne ainsi la sécurité.

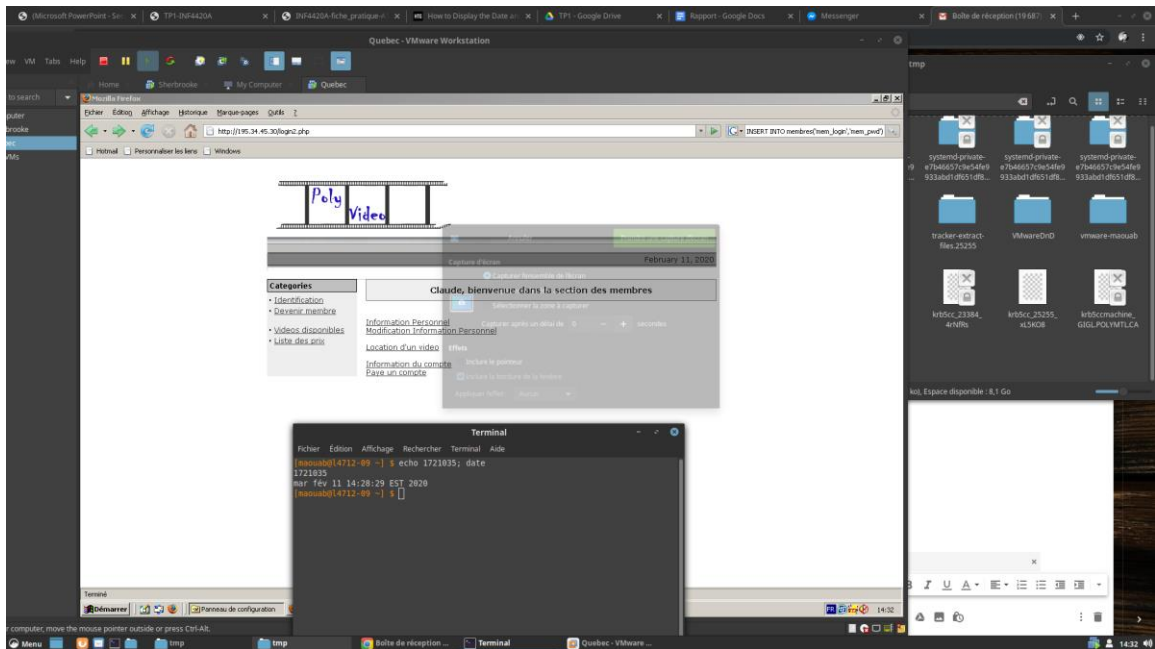


Figure 42 : Confirmation de l'exécution de l'attaque

3. Pour les deux attaques, le message d'erreur retourné lorsqu'on rentre des faux identifiants retourne un message d'erreur sur ou l'erreur s'est produite. Ceci permet donc de guider un utilisateur malveillant vers l'utilisation de l'injection SQL. Par ailleurs, Le PHP parle aux bases de données SQL à l'aide de pilotes de base de données. Un pilote permet à une application de construire et d'exécuter des instructions SQL sur une base de données, d'extraire et de manipuler des données selon les besoins. Donc, les requêtes paramétrées s'assurent que les paramètres (c'est-à-dire les entrées login et mot de passe) passés dans les instructions SQL sont traités de manière sûre. Or, ici aucune requête paramétrée n'est utilisée. Enfin, les développeurs doivent toujours s'efforcer de rejeter les entrées qui semblent suspectes et ce n'est pas le cas ici.

4. Comme il s'agit d'une base de données MySQL, les requêtes doivent être correctement créées pour ne pas permettre l'exécution de code arbitraire. En passant les paramètres donnés par l'utilisateur à la fonction `mysql_real_escape_string`, certains caractères comme les guillemets sont transformés d'une façon à ne pas affecter la requête.

```
extract($_POST);
```

```
$escaped_login = mysql_real_escape_string($login);
```

```
$escaped_password = mysql_real_escape_string($pass)
```

```
$req = "select mem_code from MEMBRES where mem_login = '$escaped_login' and mem_pwd = '$escaped_password'";
```

```
$result = mysql_query($req) or
```

```
die("Error : the SQL request ".$req." is not valid: ".mysql_error());
```

```
list($mem_code) = mysql_fetch_array($result);
```

```
if (empty($mem_code)) { //verifier que la requete a retourne une reponse positive
```

Question 4

1. Nous fournissons au système la chaîne de caractères suivante pour le *Username* : `aa` (soit le même caractère répété 61 fois), et pour le *Password* : `a` (soit le dernier caractère de la chaîne précédente). Puisque l'espace alloué à la variable *username* et *password* est de 20 octets chacun et que l'espace alloué pour un utilisateur est de 40 octets, écrire un nom d'utilisateur d'une longueur équivalente à la somme de son espace, de l'espace du mot de passe et de l'espace du premier attribut d'un utilisateur nous permet de d'écraser le premier utilisateur dans la liste d'utilisateurs enregistrés. Par la suite, en donnant le mot de passe contenu après les 60 caractères du nom d'utilisateur permet de positionner un *null terminator* au bon endroit dans la mémoire pour couper le nom d'utilisateur et valider le *strcmp*. À la suite de notre expérimentation, nous avons réalisé que le *Username* pouvait simplement être 60 caractères identiques au lieu de 61 et que le *Password* pouvait être laissé vide puisqu'un *null terminator* serait automatiquement ajouté.

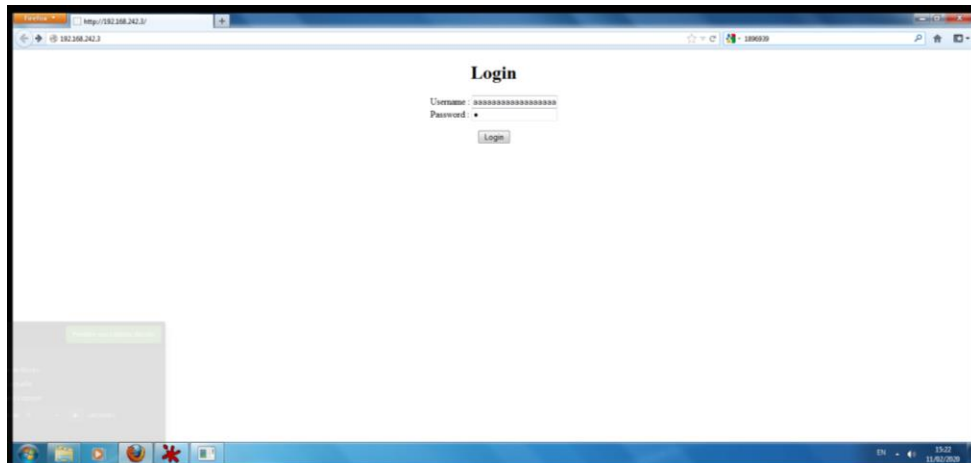


Figure 43 : Page à attaquer

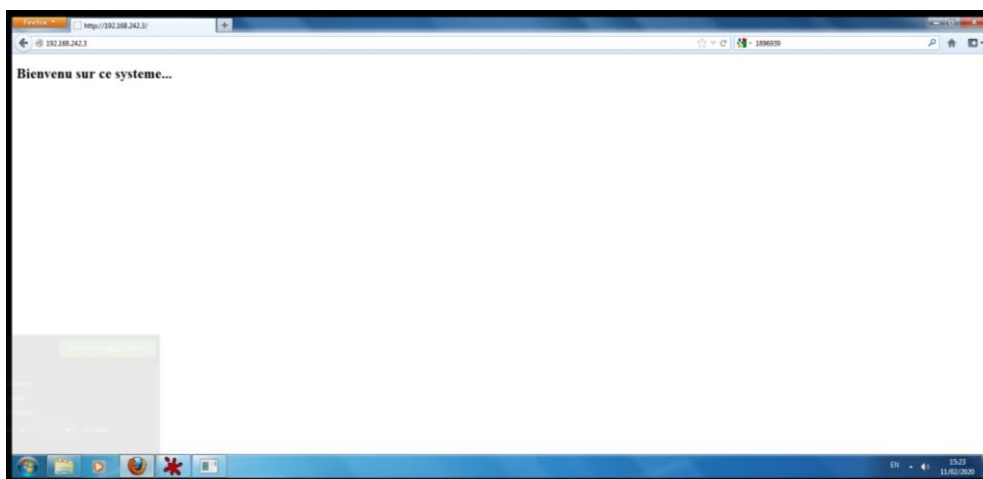


Figure 44 : Confirmation de l'exécution de l'attaque

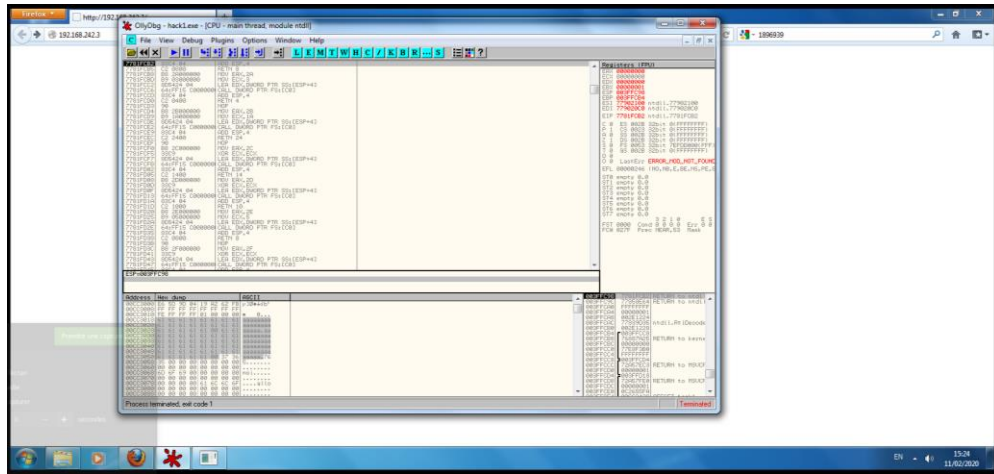


Figure 45 : Outil utilisé pour découvrir l'attaque

2. Pour corriger cette faille de sécurité, nous pourrions simplement changer les appels à la fonction `gets` par des appels à la fonction `fgets` en spécifiant la longueur de la chaîne de caractères.

Question 5

1. La faiblesse se trouve dans le choix numéro 2 du programme en effet quand on nous demande le nom du fichier un simple scanf sans vérification ou limite de caractère nous permet de déborder du stack. Sachant qu'après 24 caractères on sort de la variable locale du stack et qu'après 4 caractères de plus on arrive dans l'adresse de retour de la méthode. Il faut rentrer l'adresse de retour voulu pour la méthode logon ou l'adresse de son call dans le stack après 28 caractères.

Donc : aaaaaaaaaaaaaaaaaaaaaaaaaaaaaa[...]

L'adresse est en hexadécimal, mais on ne peut entrer que des caractères ascii donc il faut entrer des caractères non-imprimables (sur windows avec alt + le code décimal de l'hexadécimal que l'on veut). Les caractères doivent être entré dans l'ordre inverse, car le logiciel est en little endian et l'adresse sera lu à l'envers octet par octet.

L'adresse finale de logon est 0x004010E0

Donc: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaa(alt+224)(alt+16)(alt+64)

Ou encore: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaÓ►@

Et celle de son call dans le switch 0x0040135C

Donc: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaa(alt+92)(alt+19)(alt+64)

Ou encore: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaa\!!@

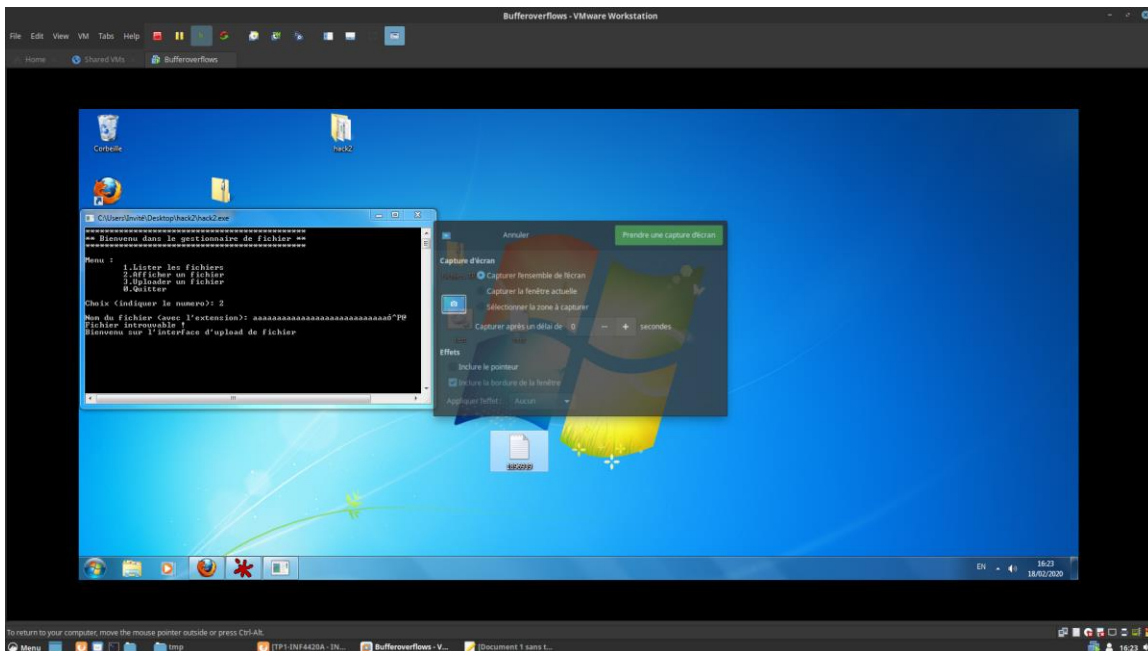


Figure 46 : Confirmation de l'exécution de l'attaque

2. Il suffirait de changer l'appel du clavier de l'utilisateur pour une fonction protégée dans la longueur, comme ça a été fait pour la demande de mot de passe du choix trois, upload. L'idée étant d'empêcher l'utilisateur de rentrer plus de 24 caractères pour qu'il ne sorte pas des variables locales.