

COMPTE-RENDU : TP LAB

Installation du poste de travail virtuel : Cyber Ops

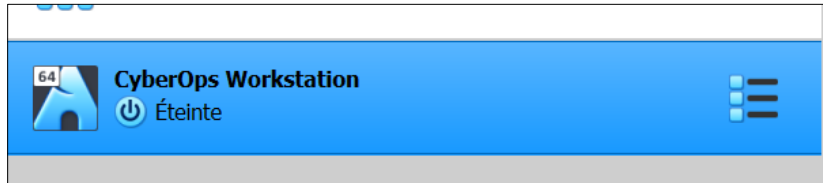
Partie 1 :

Tâches effectuées :

- Mise à jour d'Oracle VirtualBox (version 6.1.14)
- Téléchargement de la machine virtuelle, avec le lien fourni dans l'énoncé

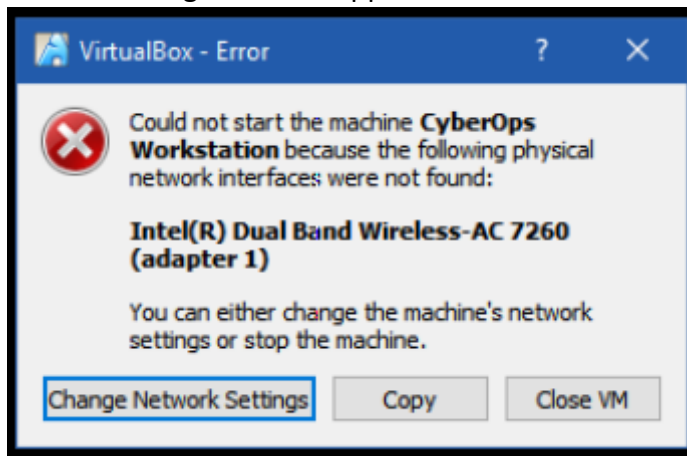
Partie 2 :

- Importer la machine virtuelle dans VirtualBox



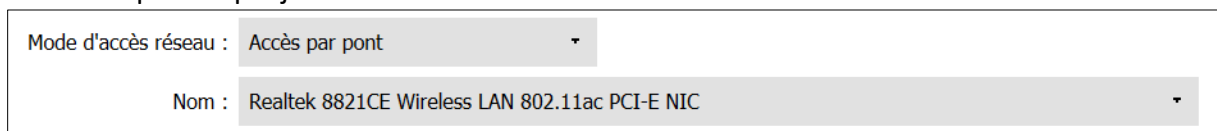
Après avoir importé la machine, il faut l'ouvrir :

Puis ce message d'erreur apparaît :

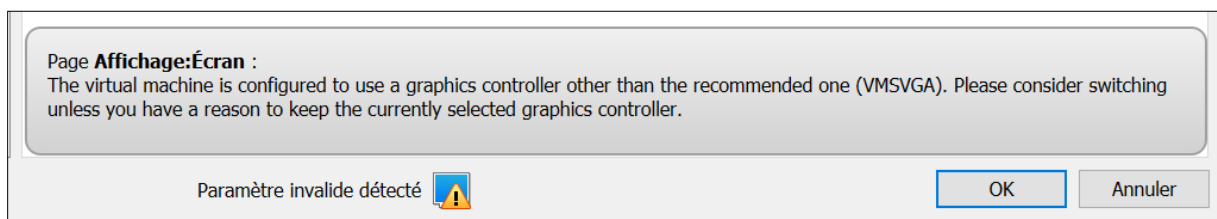


Pour le résoudre il faut changer l'adaptateur au réseau dans Configuration > Réseau

Voilà les options que j'ai choisies :



En regardant dans l'onglet Configuration j'ai vu également cette erreur s'afficher en bas de la fenêtre :



Cette erreur explique la machine virtuelle est configuré pour utiliser a contrôleur graphique autre que celui recommandé.

COMPTE-RENDU : TP LAB

Donc j'ai suivi les instructions et choisis le contrôleur graphique VMSVGA :

Avant :

Contrôleur graphique : VBoxVGA ▼

Et j'ai sélectionné celui-ci et le message d'erreur a disparu :

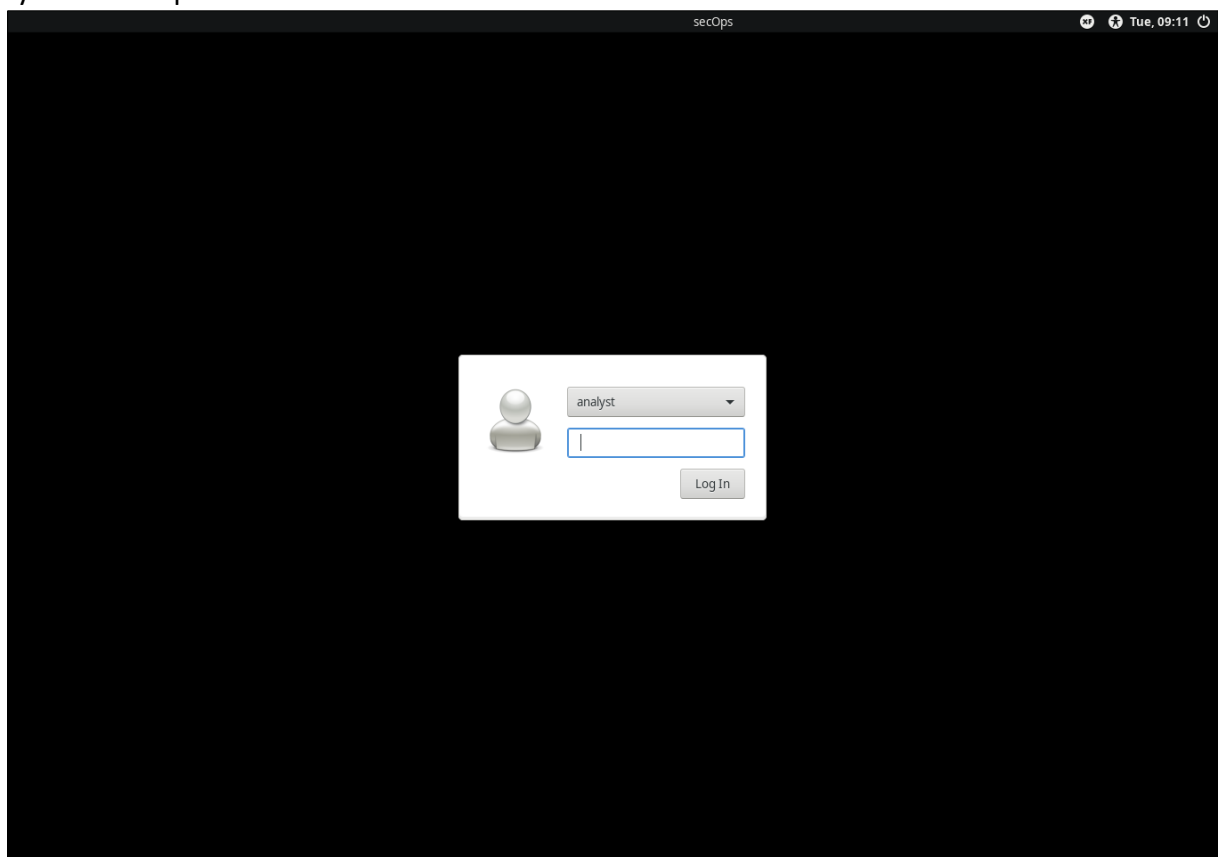
Contrôleur graphique : VMSVGA ▼

Puis le message situé dans la machine virtuelle a disparu

Le processus de démarrage est terminé, il faut nous connecter à la machine virtuelle avec le nom d'utilisateur et le mot de passe donné dans l'énoncé :

Remarque :

Il faut toucher la touche CTRL pour basculer le fonctionnement du clavier et de la souris sur le système d'exploitation de l'hôte



COMPTE-RENDU : TP LAB

Ensuite la machine virtuelle se lance :



↑Affichage du bureau

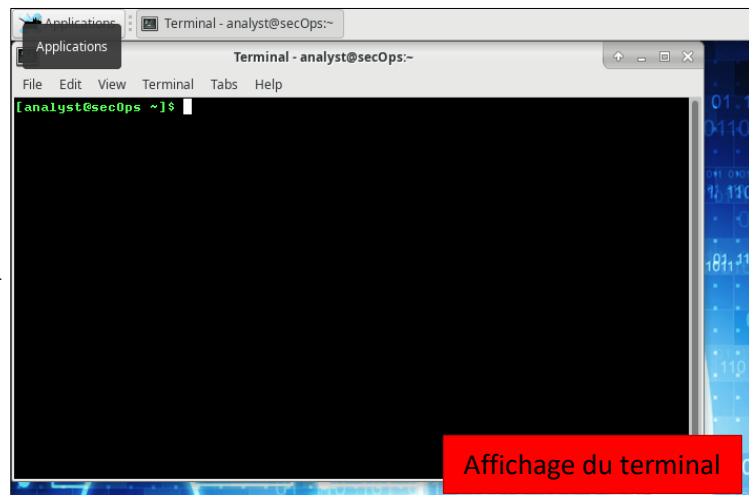
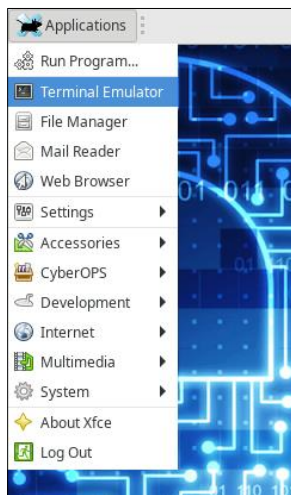
Étape 3 : Pour se familiariser avec la machine virtuelle

>> Application du terminal

On peut cliquer sur la barre de tâche en bas du bureau :

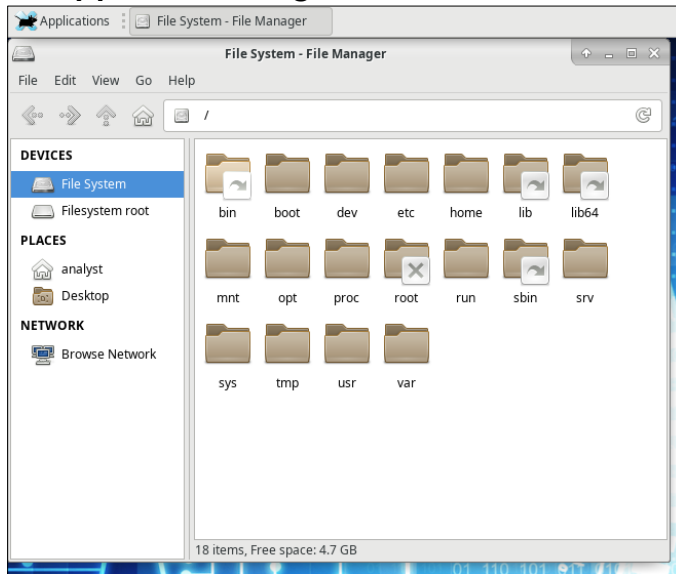


Ou



COMPTE-RENDU : TP LAB

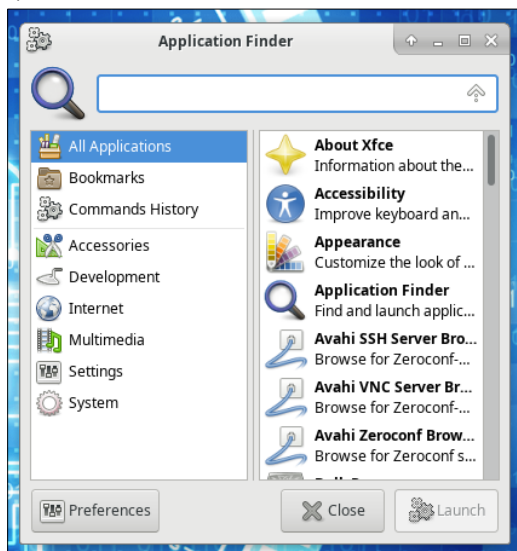
>> Application de gestion des fichiers



>> Application de navigateur

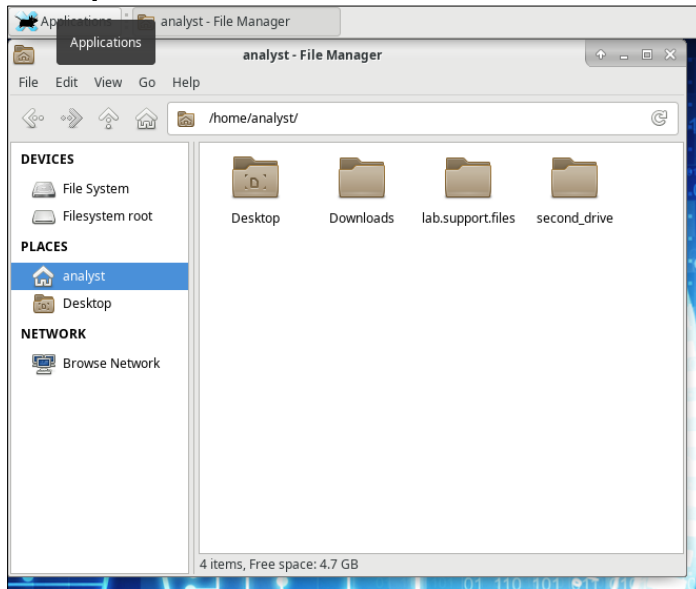


>> Outil de recherche de fichiers

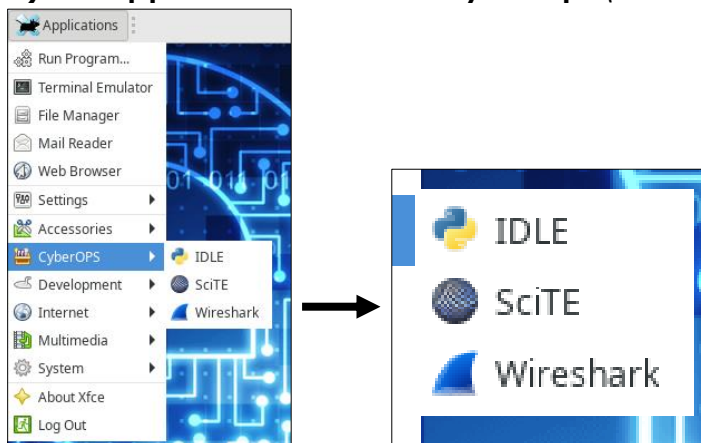


COMPTE-RENDU : TP LAB

>> Répertoire de base de l'utilisateur actuel



A) Les application du menu CyberOps (voir l'énoncé)



B) Les adresses IP attribuées à la machine virtuelle

```
[analyst@secOps ~]$ ip adress
Object "adress" is unknown, try "ip help".
[analyst@secOps ~]$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:07:4d:8a brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.27/24 brd 192.168.1.255 scope global dynamic enp0s3
        valid_lft 85872sec preferred_lft 85872sec
    inet6 2a01:cb20:4079:d600:a00:27ff:fe07:4d8a/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 1774sec preferred_lft 574sec
    inet6 fe80::a00:27ff:fe07:4d8a/64 scope link
        valid_lft forever preferred_lft forever
```

COMPTE-RENDU : TP LAB

C) Accès au navigateur web *(voir l'énoncé)*

À noter : Comment arrêter la machine virtuelle

- 1) Enregistrer l'état de la machine puis OK
- 2) Éteindre la machine
- 3) Mettre la machine hors tension

Mais également :

- 4) Arrêter et Redémarrer à l'aide de ligne de commande

Remarque :

Quels sont les avantages et les inconvénients de l'utilisation d'un ordinateur virtuel ?

Avantages :

- Utilisation de plusieurs système d'exploitation sur différentes machines virtuelles sur une même machine.

En cas d'erreur d'exécution, on peut les restaurer.

- Peut servir comme environnement de test

Inconvénient : La machine virtuelle peut être lente ou ralentir l'hôte lui même

COMPTE-RENDU : TP LAB

La réponse aux diverses questions posées dans le TP sont inscrites directement dans le sujet

Utilisation des fichiers texte dans l'interface de ligne de commande (CLI)

Partie 1 : Éditeurs de texte graphique

Se familiariser avec les fichiers textes sous Linux

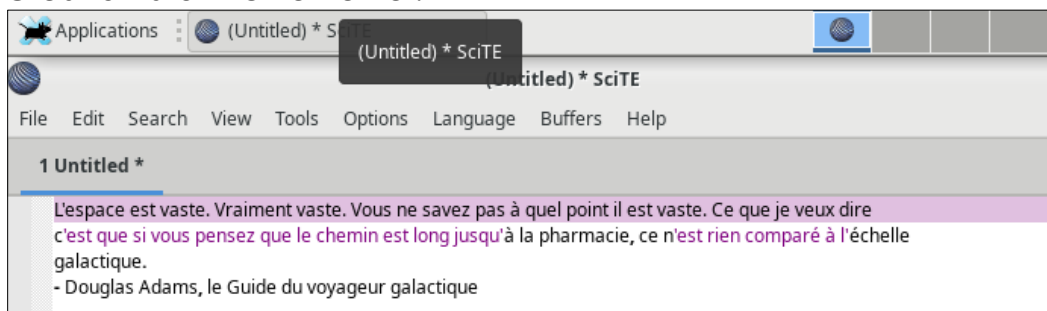
Sur la machine virtuelle **Cisco CyberOPS**, il existe uniquement **SciTE** comme application d'éditeur de texte graphique. Très simple et rapide, pas beaucoup de fonctionnalités avancées donc permet d'effectuer les tâches dans ce cours.

Étape 1 :

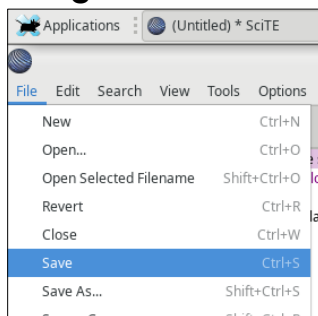
SciTE contient :

- environnement avec des onglets
- syntaxique colorée

Création d'un fichier texte :

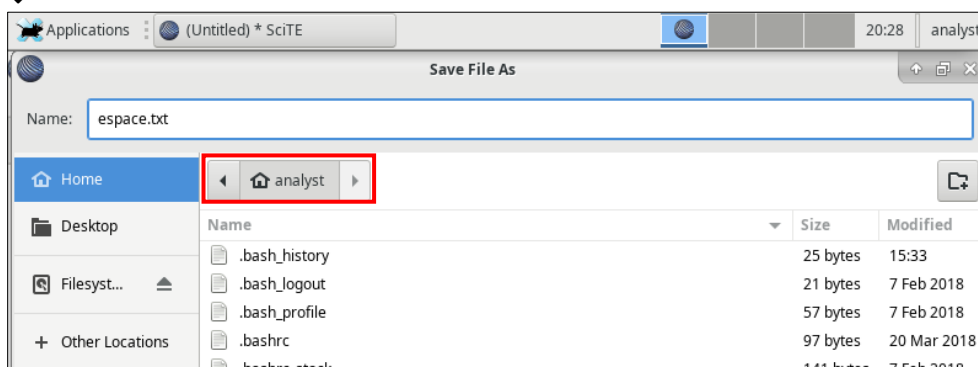


Enregistrement du fichier sous le nom : espace.txt



SciTE enregistre le fichier dans le répertoire de base de l'utilisateur actuel : « **analyst** »

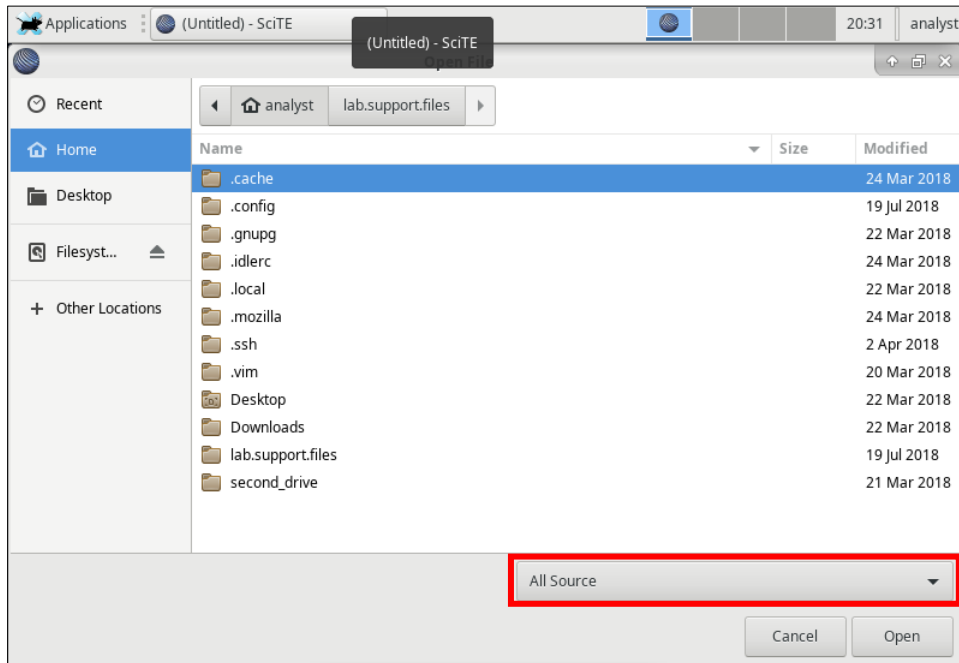
↓



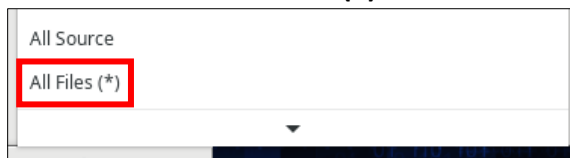
COMPTE-RENDU : TP LAB

Fermer puis réouvert SciTE, recherchez le fichier espace.txt

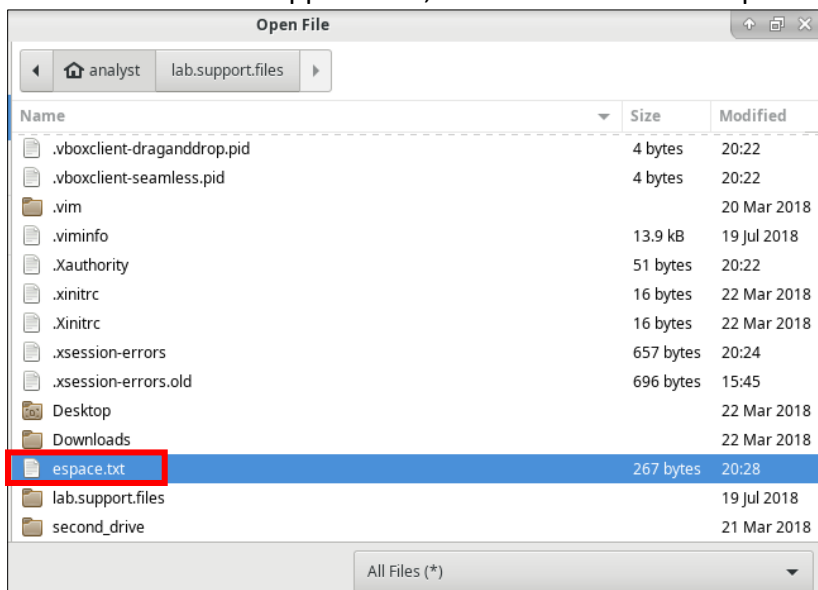
Après avoir cliqué sur **File > Open ...** on observe cette page et le fichier **espace.txt** n'est pas visible



SciTE recherche les extensions connues et **.txt** n'en fait pas partie. Donc afin d'afficher tous les fichiers, on clique sur le menu déroulant en bas à droite (encadré en rouge ↑ sur la capture précédente) Puis sélectionner **All Files (*)**

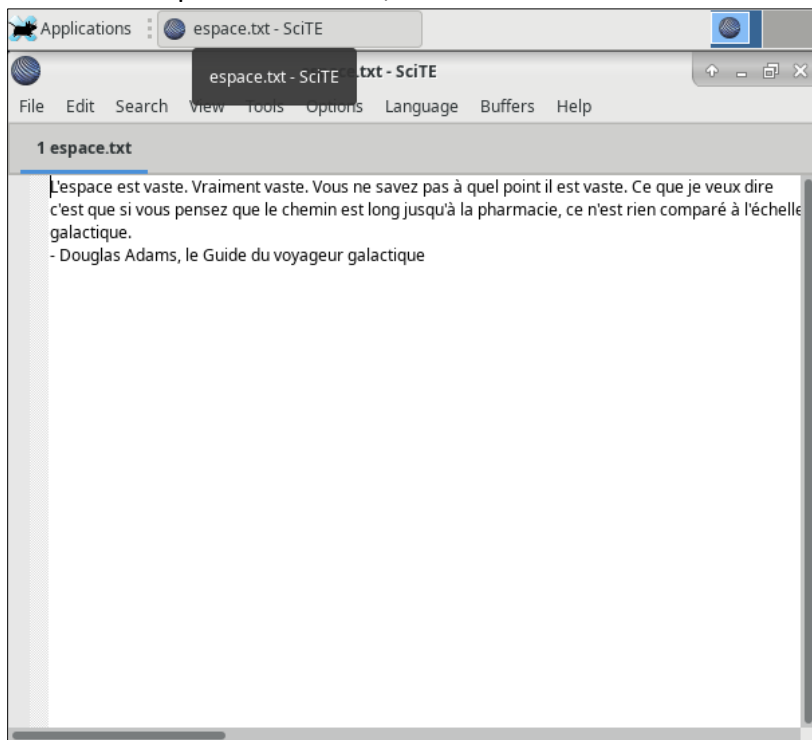


Puis tous les fichiers apparaissent, et retrouve le fichier espace.txt



COMPTE-RENDU : TP LAB

Quand on clique sur le fichier, il s'ouvre correctement :

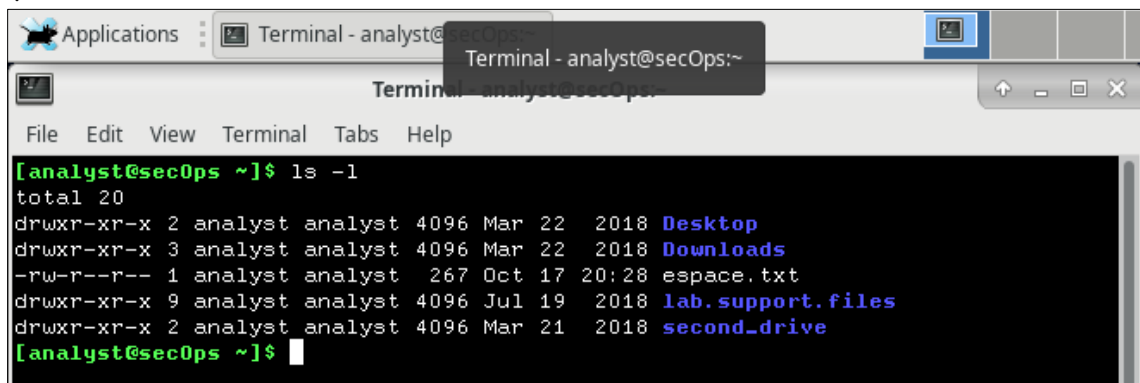


Étape 2 :

Il est possible d'ouvrir SciTE depuis la ligne de commande à l'aide du Terminal

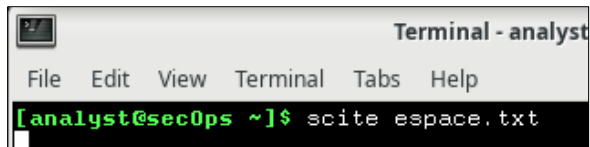
- La commande **ls** :

↓



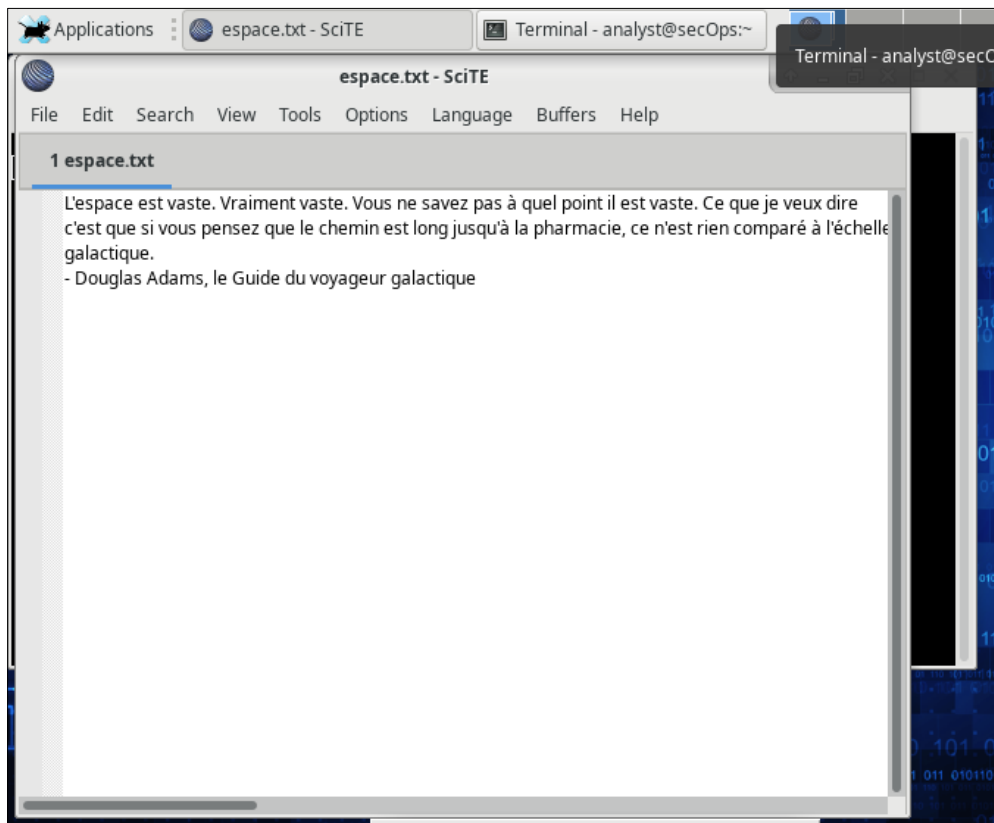
COMPTE-RENDU : TP LAB

- La commande **scite espace.txt** :



```
Terminal - analyst
File Edit View Terminal Tabs Help
[analyst@secOps ~]$ scite espace.txt
```

En cliquant sur la touche Entrer, voici se qui s'affiche :



Par-dessus le terminal, le fichier espace.txt dans SciTE s'ouvre automatiquement.

Remarque :

Il est utile de démarrer SciTE à partir de la ligne de commande lorsque vous souhaitez exécuter SciTE en tant que root. Il suffit de faire précéder **scite** par la commande **sudo** : « **sudo scite** ».

COMPTE-RENDU : TP LAB

Partie 2 : Éditeurs de texte utilisant une ligne de commande

Bien que les éditeurs de texte graphique soient faciles et pratique d'utilisation, les éditeurs de texte utilisant les lignes de commande sont importants également sur Linux.

AVANTAGES :

- Permet de modifier un fichier à partir d'un interpréteur de commande sur un ordinateur distant.

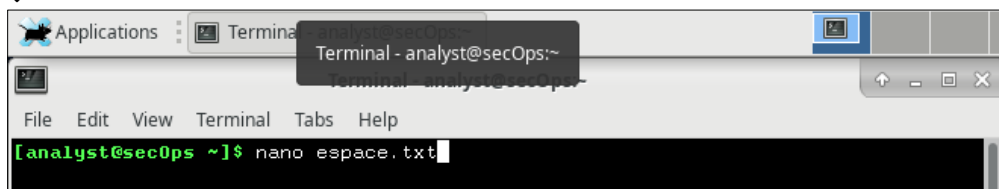
La machine virtuelle **Cisco CyberOPS** comprend quelques éditeurs de texte utilisant une ligne de commande.

Cette partie est axé sur l'éditeur de texte **nano**.

nano peut être manipulé qu'avec le clavier : **CTRL + O** pour enregistrer un fichier, **CTRL + W** pour ouvrir le menu de recherche etc.

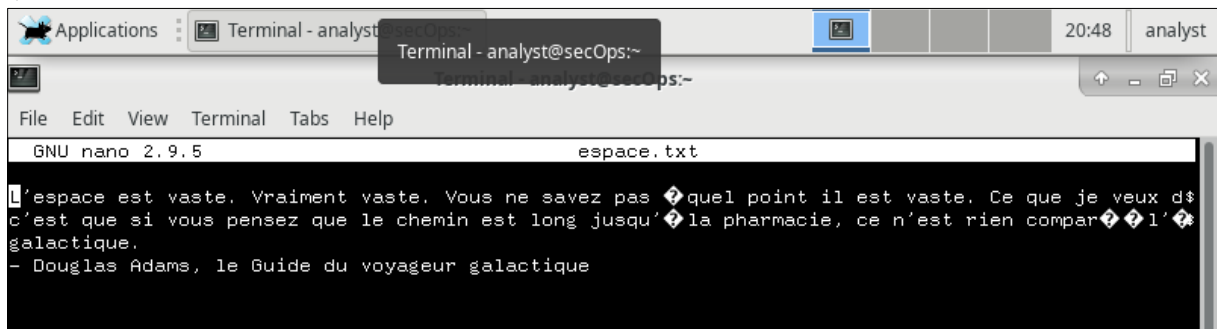
Ouvrir le fichier espace.txt avec nano dans le Terminal :

↓



Voila comment le fichier se présente :

↓



Quel caractère **nano** utilise-t-il pour représenter une ligne qui se prolonge au-delà des limites de l'écran ?

Pour montrer que la ligne se prolonge au-delà des limites de l'écran on observe le caractère " \$ "

COMPTE-RENDU : TP LAB

Partie 3 : Utiliser des fichiers de configurations

Les fichiers de configuration sont en général de fichiers texte dans lesquels les réglages et les paramètres des services et des applications sont stockés.

Presque tous les éléments de linux ont recours aux fichiers de configuration pour fonctionner. Certains services peuvent avoir plusieurs fichiers de configuration.

Étape 1 : Rechercher les fichiers de configuration

Sous Linux les fichiers de configuration qui sont utilisés pour configurer les applications sont souvent placés dans le répertoire de base de l'utilisateur tandis que les fichiers de configuration utilisés pour contrôler les services à l'échelle du système sont placés dans le répertoire **/etc**.

- La commande **ls** répertorie les fichiers du répertoire de base de l'utilisateur « **analyst** »

```
[analyst@secOps ~]$ ls -l
total 20
drwxr-xr-x 2 analyst analyst 4096 Mar 22 2018 Desktop
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 Downloads
-rw-r--r-- 1 analyst analyst 267 Oct 17 20:28 espace.txt
drwxr-xr-x 9 analyst analyst 4096 Jul 19 2018 lab.support.files
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
```

Les fichiers de configuration hébergés dans le répertoire de base sont en général masqués et leur nom est précédé d'un caractère «**.**»

- La commande **ls -la** permet d'afficher les fichiers cachés :

```
[analyst@secOps ~]$ ls -la
total 144
drwx----- 14 analyst analyst 4096 Oct 17 20:28 .
drwxr-xr-x 3 root root 4096 Mar 20 2018 ..
-rw----- 1 analyst analyst 74 Oct 17 20:51 .bash_history
-rw-r--r-- 1 analyst analyst 21 Feb 7 2018 .bash_logout
-rw-r--r-- 1 analyst analyst 57 Feb 7 2018 .bash_profile
-rw-r--r-- 1 analyst analyst 97 Mar 20 2018 .bashrc
-rw-r--r-- 1 analyst analyst 141 Feb 7 2018 .bashrc_stock
drwxr-xr-x 7 analyst analyst 4096 Mar 24 2018 .cache
drwxr-xr-x 10 analyst analyst 4096 Jul 19 2018 .config
drwxr-xr-x 2 analyst analyst 4096 Mar 22 2018 Desktop
-rw-r--r-- 1 analyst analyst 23 Mar 23 2018 .dmrc
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 Downloads
-rw-r--r-- 1 analyst analyst 267 Oct 17 20:28 espace.txt
drwx----- 3 analyst analyst 4096 Mar 22 2018 .gnupg
-rw----- 1 analyst analyst 2520 Oct 17 20:22 .ICEauthority
drwxr-xr-x 2 analyst analyst 4096 Mar 24 2018 .idlerc
drwxr-xr-x 9 analyst analyst 4096 Jul 19 2018 lab.support.files
-rw----- 1 analyst analyst 51 Apr 2 2018 .lessht
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 .local
drwx----- 5 analyst analyst 4096 Mar 24 2018 .mozilla
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
drwx----- 2 analyst analyst 4096 Apr 2 2018 .ssh
-rw-r----- 1 analyst analyst 4 Oct 17 20:22 .vboxclient-clipboard.pid
-rw-r----- 1 analyst analyst 4 Oct 17 20:22 .vboxclient-display.pid
-rw-r----- 1 analyst analyst 4 Oct 17 20:22 .vboxclient-draganddrop.pid
-rw-r----- 1 analyst analyst 4 Oct 17 20:22 .vboxclient-seamless.pid
drwxr-xr-x 3 analyst analyst 4096 Mar 20 2018 .vim
-rw----- 1 analyst analyst 13912 Jul 19 2018 .viminfo
-rw----- 1 analyst analyst 51 Oct 17 20:22 .Xauthority
-rw-r--r-- 1 analyst analyst 16 Mar 22 2018 .xinitrc
-rw-r--r-- 1 analyst analyst 16 Mar 22 2018 .Xinitrc
-rw----- 1 analyst analyst 657 Oct 17 20:24 .xsession-errors
-rw----- 1 analyst analyst 696 Oct 17 15:45 .xsession-errors.old
```

COMPTE-RENDU : TP LAB

Pour afficher le contenu d'un fichier on utilise la commande **cat**

Le fichier **.bashrc** sert à configurer la personnalisation et le comportement du Terminal selon les besoins de l'utilisateur.

↓

Résultat :

```
[analyst@sec0ps ~]$ cat .bashrc
export EDITOR=vim

PS1='\[\e[1;32m\][\u@\h \W]\$'\[\e[0m\] '
alias ls="ls --color"
alias vi="vim"
```

À noter : seul l'utilisateur root a accès au répertoire **/etc** qui stocke les fichiers de configuration relatifs aux services du système.

- Commande **ls** pour afficher le contenu du répertoire **/etc**

```
[analyst@sec0ps ~]$ ls /etc
adjtime          fstab            issue            makepkg.conf     openldap          request-key.conf  sudoers.d
apparmor.d       gai.conf         kernel           man_db.conf      openvswitch       request-key.d     sysctl.d
arch-release     group           krb5.conf        mdadm.conf       os-release        resolv.conf       syslog-ng
avahi            group-          ld.so.cache      mime.types       pacman.conf       resolvconf.conf  systemd
bash.bash_logout grub.d          ld.so.conf       mkinitcpio.conf  pam.d             rpc               tmpfiles.d
bash.bashrc      gshadow         ld.so.conf.d     mkinitcpio.d     passwd            security          trusted-key.key
binfmt.d         gshadow-       libnl            modprobe.d       passw             sensors3.conf    ts.conf
ca-certificates gtk-2.0         lightdm          modules-load.d   passwd-          sensors.d        udev
conf.d           gtk-3.0        locale.conf     motd              pcmcia            services         UPower
crypttab         healthd.conf   locale.gen      mtab              pkcs11            shadow          vbox
dbus-1           host.conf      localtime       nanorc            polkit-1          shadow-         vdpau-wrapper.cfg
default          hostname      login.defs      netconfig        profile           shells          vimrc
depmod.d         hosts         logrotate.conf  netctl           profile.d         skel            vsftpd.conf
dhcpcd.conf      ifplugd       logrotate.d     nginx            protocols        snort           vsftpd.conf_stock
dirc             initcpio      lvm             nscd.conf        pulse            ssh             xdg
environment      iproute2      mailcap         nsswitch.conf    rc.d              ssl             xinetd.d
ethertypes       iptables      mail.rc         ntp.conf         rc-keymaps        sudoers         yaourtc
```

- Afficher le contenu du fichier **bash_bashrc** :

```
[analyst@sec0ps ~]$ cat /etc/bash.bashrc
#
# /etc/bash.bashrc
#

# If not running interactively, don't do anything
[[ $- != *i* ]] && return

[[ $DISPLAY ]] && shopt -s checkwinsize

PS1='\[\u@\h \W\]\$ '

case ${TERM} in
    xterm*|rxvt*|Eterm|aterm|kterm|gnome*)
        PROMPT_COMMAND=${PROMPT_COMMAND:+$PROMPT_COMMAND;} 'printf "\033]0;%s@%s:%s\007" "${USER}" "${HOSTNAME%%.*}" "${PWD/#$HOME}/\~"'
        ;;
    screen*)
        PROMPT_COMMAND=${PROMPT_COMMAND:+$PROMPT_COMMAND;} 'printf "\033_%s@%s:%s\033\\" "${USER}" "${HOSTNAME%%.*}" "${PWD/#$HOME}/\~"'
        ;;
esac

[ -r /usr/share/bash-completion/bash_completion ] && . /usr/share/bash-completion/bash_completion
```

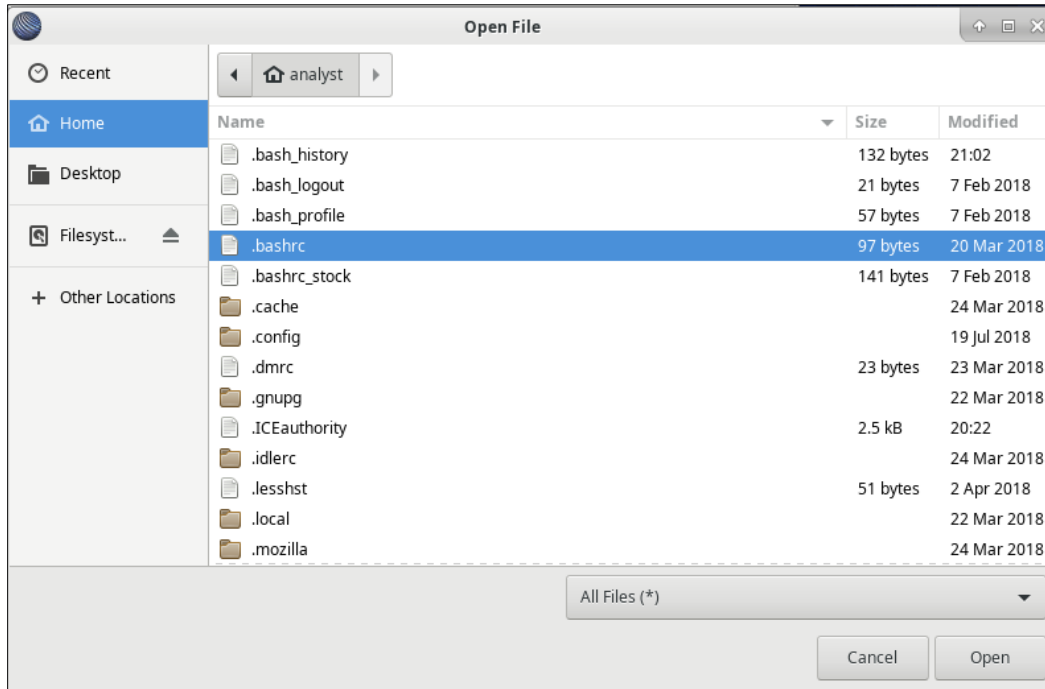
Ce fichier définit le comportement par défaut de l'interpréteur de commande par l'utilisateur. Pour modifier ce fichier, l'utilisateur doit se connecter en tant que **root**.

COMPTE-RENDU : TP LAB

Étape 2 : Modifier et enregistrer les fichiers de configuration

Modification du fichier **.bashrc** pour changer la couleur de l'invite de commandes du vert au rouge pour l'utilisateur **analyst**

Ouvrir le fichier **.bashrc** à l'aide de SciTE :



Remplacer le nombre 32 (code pour le vert) par 31 (code pour le rouge)

```
1 .bashrc
export EDITOR=vim

PS1="\[\e[1;32m\][\u@\h \W]\$\[\e[0m\] '
alias ls="ls --color"
alias vi="vim"
```

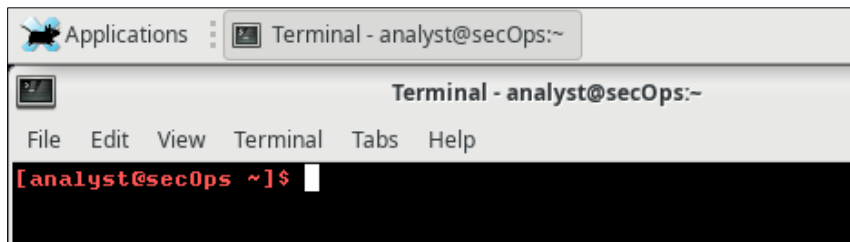


```
1 .bashrc *
export EDITOR=vim

PS1="\[\e[1;31m\][\u@\h \W]\$\[\e[0m\] '
alias ls="ls --color"
alias vi="vim"
```

COMPTE-RENDU : TP LAB

Résultat :



```
Applications  Terminal - analyst@secOps:~  
Terminal - analyst@secOps:~  
File Edit View Terminal Tabs Help  
[analyst@secOps ~]$
```

Le même changement peut être effectué à partir de la ligne de commande avec un éditeur de texte tel que nano :



```
[analyst@secOps ~]$ nano .bashrc
```


↓



```
GNU nano 2.9.5 .bashrc  
export EDITOR=vim  
PS1='\[\e[1;31m\][\u@\h \W]\$[\e[0m\] ' '  
alias ls="ls --color"  
alias vi="vim"
```

Remplacer 31 (code du rouge) par 33 (code du jaune)

↓



```
GNU nano 2.9.5 .bashrc Modified  
export EDITOR=vim  
PS1='\[\e[1;33m\][\u@\h \W]\$[\e[0m\] ' '  
alias ls="ls --color"  
alias vi="vim"
```

Après avoir enregistré le changement voila le résultat :



```
[analyst@secOps ~]$
```

COMPTE-RENDU : TP LAB

Étape 3 : Modifier les fichiers de configuration des services

nginx est un serveur web qui est installé sur la machine virtuelle **Cisco CyberOPS**.

Il est personnalisable. Son fichier se trouve dans le répertoire **/etc**

A l'aide **nano**, on ouvre le fichier de configuration nginx

Le commutateur **-l** permet d'activer la numérotation des lignes

↓

```
[analyst@secOps ~]$ sudo nano -l /etc/nginx/custom_server.conf
```

Il nous demande de saisir le mot de passe de l'utilisateur analyst

```
[analyst@secOps ~]$ sudo nano -l /etc/nginx/custom_server.conf
[sudo] password for analyst:
```

Quand le mot de passe est bien saisi, il nous affiche le fichier de configuration de nginx nommé **custom_server.conf**

↓

```
GNU nano 2.9.5 /etc/nginx/custom_server.conf
1
2 #user html;
3 worker_processes 1;
4
5 #error_log logs/error.log;
6 #error_log logs/error.log notice;
7 #error_log logs/error.log info;
8
9 #pid logs/nginx.pid;
10
11
12 events {
13     worker_connections 1024;
14 }
15
16
17 http {
18     include mime.types;
19     default_type application/octet-stream;
20
21     #log_format main '$remote_addr - $remote_user [$time_local] "$request" '
22     # '$status $body_bytes_sent "$http_referer" '
23     # '"$http_user_agent" "$http_x_forwarded_for"';
24
25     #access_log logs/access.log main;
26
27     sendfile on;
28     #tcp_nopush on;
29
30     #keepalive_timeout 0;
31     keepalive_timeout 65;
32
33     #gzip on;
34
35     types_hash_max_size 4096;
36     server_names_hash_bucket_size 128;
37 }
```

Remarque : En général, les extensions **.conf** sont utilisés pour identifier les fichiers de configuration

COMPTE-RENDU : TP LAB

A la ligne 39, on va changer le numéro du port en remplaçant 81 par 8080 qui correspond aux requêtes http sur le port TCP 8080

```
37
38     server {
39         listen      81;
40         server_name localhost;
41     }
```

```
37
38     server {
39         listen      8080;
40         server_name localhost;
41     }
```

A la ligne 47 changer le chemin :

```
46     location / {
47         root      /usr/share/nginx/html;
48         index     index.html index.htm;
49     }
```

```
45
46     location / {
47         root      /usr/share/nginx/html/text_ed_lab;
48         index     index.html index.htm;
49     }
50
```

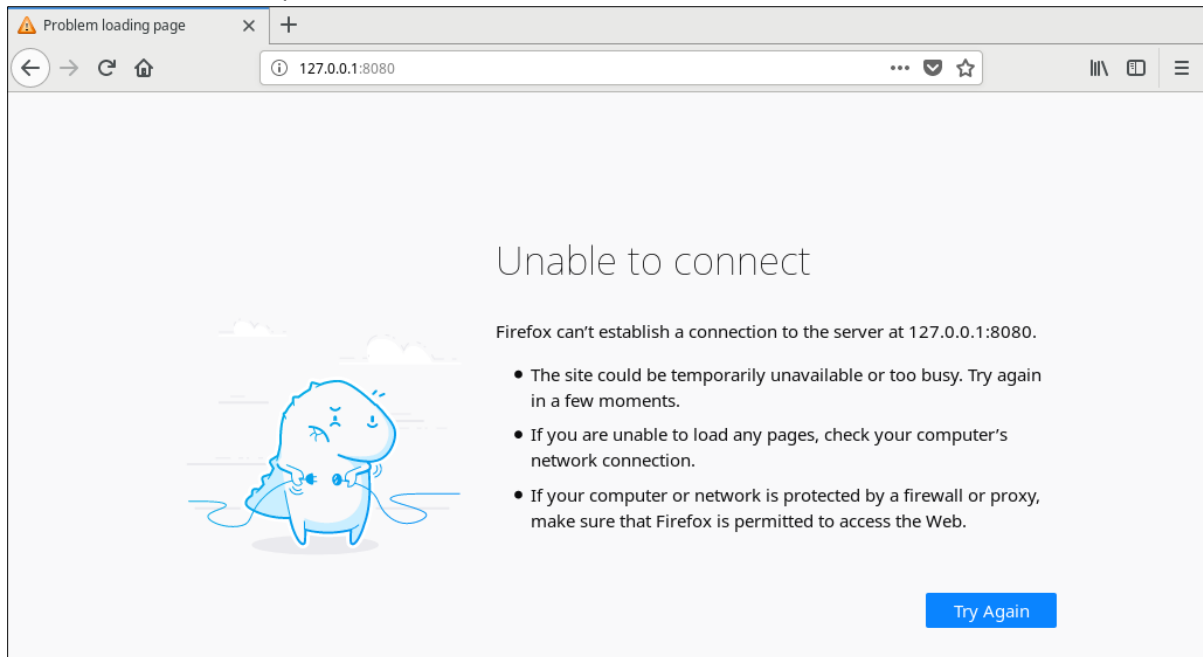
On va exécuter le fichier de configuration modifié précédemment avec la comande suivante :

```
[analyst@secOps ~]$ sudo nginx -c custom_server.conf -g "pid/var/run/nginx_v.pid;"
[sudo] password for analyst:
```

La section -g "pid /var/run/nginx_v.pid;" est nécessaire pour que nginx sache quel fichier utiliser lors du stockage de l'ID de processus qui identifie cette instance de nginx.

COMPTE-RENDU : TP LAB

Dans le navigateur, on tape **127.0.0.1:8080** pour se connecter à un serveur web hébergé sur l'ordinateur local via le port 8080.



On arrête le serveur web nginx puis dans l'invite de commande on écrit la commande suivante :

```
[analyst@secOps ~]$ sudo pkill nginx
```

COMPTE-RENDU : TP LAB

La réponse aux diverses questions posées dans le TP sont inscrites directement dans le sujet

Se familiariser avec le shell Linux

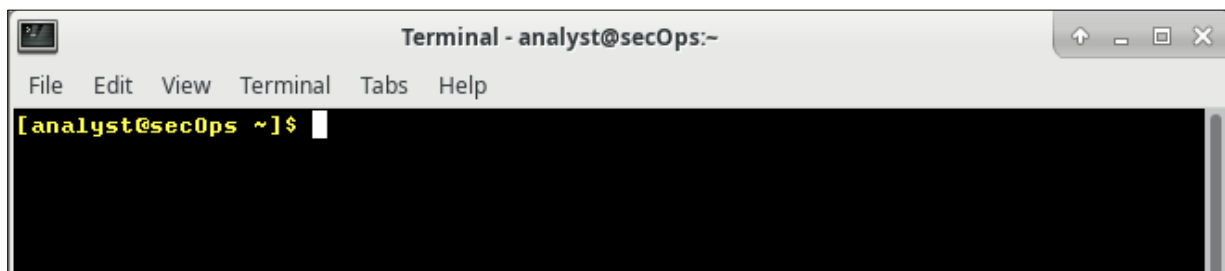
Utiliser la ligne de commande Linux pour gérer les fichiers et les répertoires, et pour effectuer certaines tâches d'administration de base.

Partie 1 : Notion de base sur le shell

Le « **shell** » désigne l'interpréteur de commande sous Linux, aussi nommé Terminal.

Étape 1 : Accéder à la ligne de commande

Afficher le Terminal.



Étape 2 : Afficher les pages de manuel à partir de la ligne de commande

La page **man** (abréviation pour manuel) est une documentation sur les commandes Linux, leurs syntaxes, leurs utilisations et leurs options.

Étape 3 : Créer et modifier des répertoires

- Pour le changement de répertoire on utilise la commande **cd**
- Pour la création de répertoire (ou dossiers) : la commande **mkdir**
- Pour lister les répertoires : la commande **ls**

- Liste des dossiers et fichiers dans le répertoire actif (c'est-à-dire analyst)

Grâce à la commande **ls -l** on a :

- La taille des fichiers
- Ses autorisations
- Ses paramètres de propriétés
- Sa date de création ...

```
[analyst@secOps ~]$ ls -l
total 20
drwxr-xr-x 2 analyst analyst 4096 Mar 22  2018 Desktop
drwxr-xr-x 3 analyst analyst 4096 Mar 22  2018 Downloads
-rw-r--r-- 1 analyst analyst  267 Oct 17 20:28 espace.txt
drwxr-xr-x 9 analyst analyst 4096 Jul 19  2018 lab.support.files
drwxr-xr-x 2 analyst analyst 4096 Mar 21  2018 second_drive
```

En **bleu** : les répertoires (= dossiers)

En blanc/gris : les fichiers

COMPTE-RENDU : TP LAB

- Création de 3 nouveaux dossiers puis vérification qu'ils sont bien créés :

```
[analyst@secOps ~]$ mkdir cyops_folder1
[analyst@secOps ~]$ mkdir cyops_folder2
[analyst@secOps ~]$ mkdir cyops_folder3
[analyst@secOps ~]$ ls -l
total 32
drwxr-xr-x 2 analyst analyst 4096 Oct 20 15:09 cyops_folder1
drwxr-xr-x 2 analyst analyst 4096 Oct 20 15:09 cyops_folder2
drwxr-xr-x 2 analyst analyst 4096 Oct 20 15:09 cyops_folder3
drwxr-xr-x 2 analyst analyst 4096 Mar 22 2018 Desktop
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 Downloads
-rw-r--r-- 1 analyst analyst 267 Oct 17 20:28 espace.txt
drwxr-xr-x 9 analyst analyst 4096 Jul 19 2018 lab.support.files
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
```

Le symbole `~` représente le répertoire de base de l'utilisateur actuel

(Ici le répertoire actuel est : `/home/analyst`)

La commande `cd` permet d'accéder au répertoire `cyops_folder2` qui devient alors à l'exécution de cette commande le répertoire actuel de l'utilisateur :

```
[analyst@secOps ~]$ cd /home/analyst/cyops_folder3
[analyst@secOps cyops_folder3]$
```

Remarque :

« `$` » indique des privilèges d'utilisateur standard

« `#` » indique des privilèges élevés (*utilisateur root*)

- Création d'un dossier dans le dossier `cyops_folder3` et vérification :

```
[analyst@secOps ~]$ mkdir /home/analyst/cyops_folder3/cyops_folder4
[analyst@secOps ~]$ ls -l /home/analyst/cyops_folder3
total 4
drwxr-xr-x 2 analyst analyst 4096 Oct 20 15:16 cyops_folder4
```

La commande `ls -la` permet d'afficher tous les fichiers pour le répertoire `cyops_folder3`

```
[analyst@secOps ~]$ ls -la /home/analyst/cyops_folder3
total 12
drwxr-xr-x 3 analyst analyst 4096 Oct 20 15:16 .
drwx----- 17 analyst analyst 4096 Oct 20 15:09 ..
drwxr-xr-x 2 analyst analyst 4096 Oct 20 15:16 cyops_folder4
```

COMPTE-RENDU : TP LAB

Étape 4 : Rediriger les sorties

(Voir les captures d'écran des tests dans le dossier LAB2screen > Partie 1)

Résumé :

- La commande **echo** permet d'envoyer un message.
- Le symbole « > » est une opération qui redirige la sortie d'une commande vers un emplacement.
- La commande **cat** permet d'afficher le contenu d'un fichier

Étape 5 : Rediriger un fichier et y ajouter des données

(Voir les captures d'écran des tests dans le dossier LAB2screen > Partie 1)

Résumé :

- Le symbole « >> » permet de rediriger les données vers un fichiers ou l'ajout des données à la fin du fichier ciblé sans modifier le contenu actuel

Test :

```
[analyst@secOps ~]$ echo Il s'agit d'une autre ligne de texte. Elle sera AJOUTEE au fichier de sortie. >> fichier_texte.txt
[analyst@secOps ~]$ cat fichier_texte.txt
Il s'agit dun message DIFFERENT envoye de nouveau au terminal par echo
Il s'agit dune autre ligne de texte. Elle sera AJOUTEE au fichier de sortie.
```

Étape 6 : Utiliser des fichiers cachés dans Linux

(Voir les captures d'écran des tests dans le dossier LAB2screen > Partie 1)

Dans Linux, les fichiers dont le nom commence par un « . » (point) ne sont pas affichés par défaut. Ils sont appelés fichiers cachés.

↓

- Utilisation de la commande **ls -l** :

```
[analyst@secOps ~]$ ls -l
total 36
drwxr-xr-x 2 analyst analyst 4096 Oct 20 15:09 cyops_folder1
drwxr-xr-x 2 analyst analyst 4096 Oct 20 15:09 cyops_folder2
drwxr-xr-x 3 analyst analyst 4096 Oct 20 15:16 cyops_folder3
drwxr-xr-x 2 analyst analyst 4096 Mar 22 2018 Desktop
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 Downloads
-rw-r--r-- 1 analyst analyst 267 Oct 17 20:28 espace.txt
-rw-r--r-- 1 analyst analyst 146 Oct 20 15:39 fichier_texte.txt
drwxr-xr-x 9 analyst analyst 4096 Jul 19 2018 lab.support.files
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
```

COMPTE-RENDU : TP LAB

- Commande **ls -la** pour afficher tous les fichiers du répertoire analyst :

```
[analyst@secOps ~]$ ls -la
total 164
drwx----- 17 analyst analyst 4096 Oct 20 15:30 .
drwxr-xr-x  3 root    root    4096 Mar 20 2018 ..
-rw-----  1 analyst analyst  369 Oct 20 14:56 .bash_history
-rw-r--r--  1 analyst analyst   21 Feb  7 2018 .bash_logout
-rw-r--r--  1 analyst analyst   57 Feb  7 2018 .bash_profile
-rw-r--r--  1 analyst analyst   97 Oct 17 21:32 .bashrc
-rw-r--r--  1 analyst analyst   97 Oct 17 21:26 .bashrc.save
-rw-r--r--  1 analyst analyst  141 Feb  7 2018 .bashrc_stock
drwxr-xr-x  7 analyst analyst 4096 Mar 24 2018 .cache
drwxr-xr-x 10 analyst analyst 4096 Jul 19 2018 .config
drwxr-xr-x  2 analyst analyst 4096 Oct 20 15:09 cyops_folder1
drwxr-xr-x  2 analyst analyst 4096 Oct 20 15:09 cyops_folder2
drwxr-xr-x  3 analyst analyst 4096 Oct 20 15:16 cyops_folder3
drwxr-xr-x  2 analyst analyst 4096 Mar 22 2018 Desktop
-rw-r--r--  1 analyst analyst   23 Mar 23 2018 .dmrc
drwxr-xr-x  3 analyst analyst 4096 Mar 22 2018 Downloads
-rw-r--r--  1 analyst analyst  267 Oct 17 20:28 espace.txt
-rw-r--r--  1 analyst analyst  146 Oct 20 15:39 fichier_texte.txt
drwx-----  3 analyst analyst 4096 Mar 22 2018 .gnupg
-rw-----  1 analyst analyst 3462 Oct 20 14:25 .ICEauthority
drwxr-xr-x  2 analyst analyst 4096 Mar 24 2018 .idlerc
drwxr-xr-x  9 analyst analyst 4096 Jul 19 2018 lab.support.files
-rw-----  1 analyst analyst   51 Apr  2 2018 .lessht
drwxr-xr-x  3 analyst analyst 4096 Mar 22 2018 .local
drwx-----  5 analyst analyst 4096 Mar 24 2018 .mozilla
drwxr-xr-x  2 analyst analyst 4096 Mar 21 2018 second_drive
drwx-----  2 analyst analyst 4096 Apr  2 2018 .ssh
-rw-r-----  1 analyst analyst    4 Oct 20 14:25 .vboxclient-clipboard.pid
-rw-r-----  1 analyst analyst    4 Oct 20 14:25 .vboxclient-display.pid
-rw-r-----  1 analyst analyst    4 Oct 20 14:25 .vboxclient-draganddrop.pid
-rw-r-----  1 analyst analyst    4 Oct 20 14:25 .vboxclient-seamless.pid
drwxr-xr-x  3 analyst analyst 4096 Mar 20 2018 .vim
-rw-----  1 analyst analyst 13912 Jul 19 2018 .viminfo
-rw-----  1 analyst analyst   51 Oct 20 14:25 .Xauthority
-rw-r--r--  1 analyst analyst   16 Mar 22 2018 .xinitrc
-rw-r--r--  1 analyst analyst   16 Mar 22 2018 .Xinitrc
-rw-----  1 analyst analyst   654 Oct 20 14:56 .xsession-errors
-rw-----  1 analyst analyst   891 Oct 20 14:01 .xsession-errors.old
```

COMPTE-RENDU : TP LAB

Partie 2 : Copier, supprimer et déplacer des fichiers

(Voir les captures d'écran des tests dans le dossier LAB2screen > Partie 2)

Étape 1 : Copier des fichiers

Résumé :

- La commande **cp** sert à copier des fichiers dans le système de fichiers local.
Après utilisation de la commande, une nouvelle copie est créée et placée dans l'emplacement indiqué, sans modifier le contenu de celui-ci
Paramètre de la commande : cp [fichier source] [sa destination (le dossier)]
- La commande **ls** pour vérifier que le fichier a bien été copier dans le dossier indiqué

Étape 2 : Supprimer des fichiers et des répertoires

Résumé :

- La commande **rm** permet de supprimer des fichiers
- La commande **ls** pour vérifier que le fichier a bien été supprimé

Étape 3 : Déplacer des fichiers et des répertoires

Résumé :

- La commande **mv** permet de déplacer des fichiers dans le système de fichiers local.
Paramètre de la commande :
mv [fichier source] [répertoire du fichier source] [sa destination]
- La commande **ls** pour vérifier que le fichier a bien été déplacé

A noter : on peut aussi déplacer des répertoires dans un autre

Conclusion :

Ce Tp m'a permis de comprendre comment fonctionnait le système d'exploitation Linux et l'importance des lignes de commande qui permettent une meilleur gestion des données (fichiers, répertoires) présents dans la machine.