Ulrich Noéjules OGO NDAYEGUIA
Etudiant en Réseaux et Systèmes / Virtualisation et Cloud Computing
ogo.ulrich@gmail.com / jules.ndayeguia@gmail.com
LinkedIn: https://www.linkedin.com/in/jules-ndayeguia-2b0098306

18/10/2024

Projet : Mise en place d'un système de détection d'intrusion avec Snort (IDS/IPS)

Fait par :

Ulrich Noéjules OGO NDAYEGUIA

Méthodologie

1. **Installation de Snort** sur un serveur (UBUNTU).
2. **Configuration des interfaces réseau** pour capturer le trafic.
3. **Mise en place des règles Snort** (détection d'attaques, etc).
4. **Test du fonctionnement de l'IDS/IPS**
5. **Analyse des alertes générées** et ajustement des règles pour une meilleure protection

Environnement Technique

- **Système d'exploitation :** UBUNTU 22.04
- **Outil principal** : Snort (version 2.9.15.1)
- **Outils complémentaires** : Wireshark (pour l'analyse des logs)
- **Matériel requis :** Une machine virtuelle installée sous VMware Workstation, GNS3, un hyperviseur du type 2 VMware Workstation ou autre.

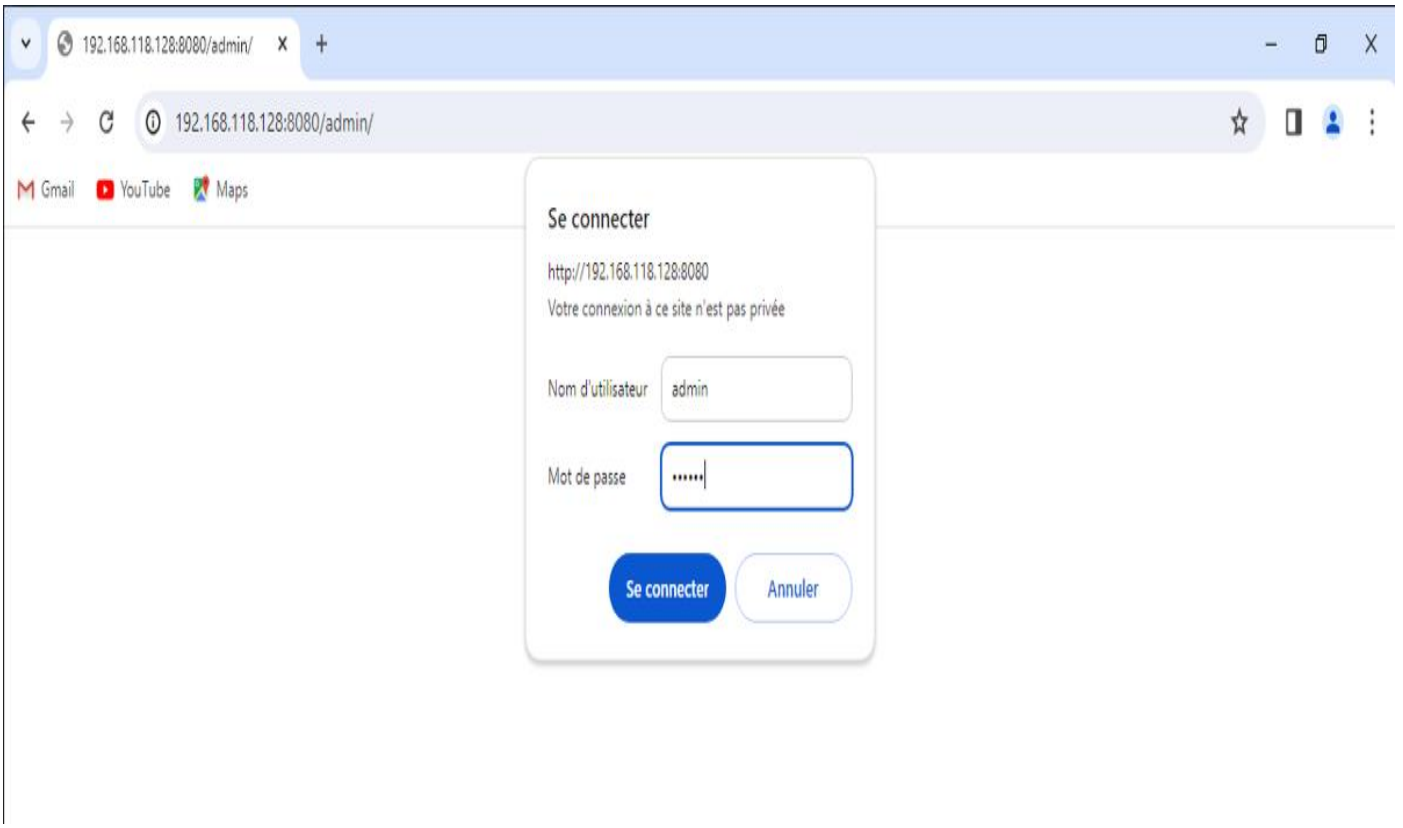
INTRODUCTION

Snort est un système de détection d'intrusion réseau (IDS) open-source, écrit en langage C. Développé en 1998 par Martin Roesch, il est aujourd'hui maintenu par Cisco. Cet outil permet aux administrateurs réseau d'analyser le trafic en temps réel afin d'identifier et de bloquer les paquets potentiellement malveillants.

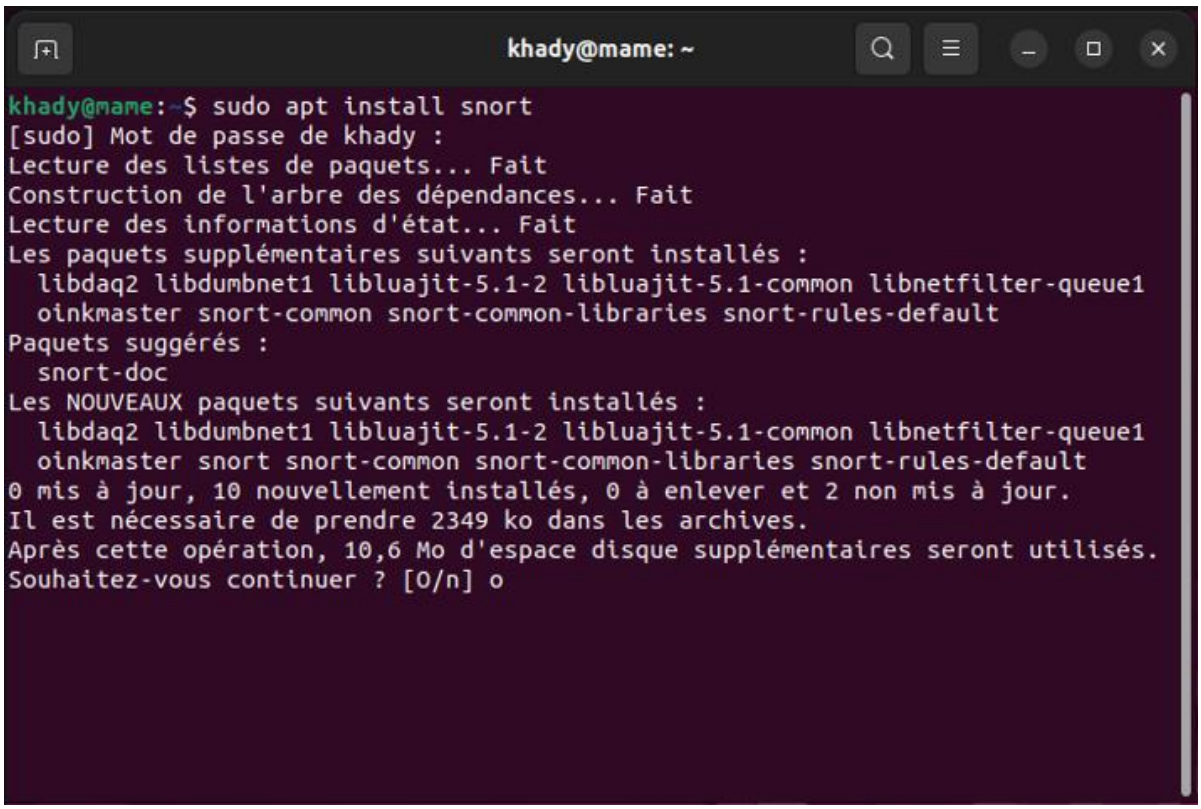
Dans ce projet, nous nous focaliserons sur la configuration pratique de Snort pour la détection d'intrusions et son intégration dans un environnement réseau. Nous détaillerons les différentes étapes de son installation, la personnalisation de ses règles et son déploiement dans un contexte réel. Enfin, nous réaliserons une simulation sous GNS3 afin de tester et d'évaluer son efficacité face à des scénarios d'attaques réels.

Pour le moment, l'accent sera mis sur l'installation et la détection d'intrusions. Pour aller plus loin, nous pourrions également implémenter Barnyard2 pour traiter les alertes de Snort et les stocker dans une base de données telle que MySQL ou PostgreSQL. Une interface web telle que BASE pourrait ensuite être utilisée pour analyser et visualiser les alertes générées par Snort et stockées dans la base de données via Barnyard2.

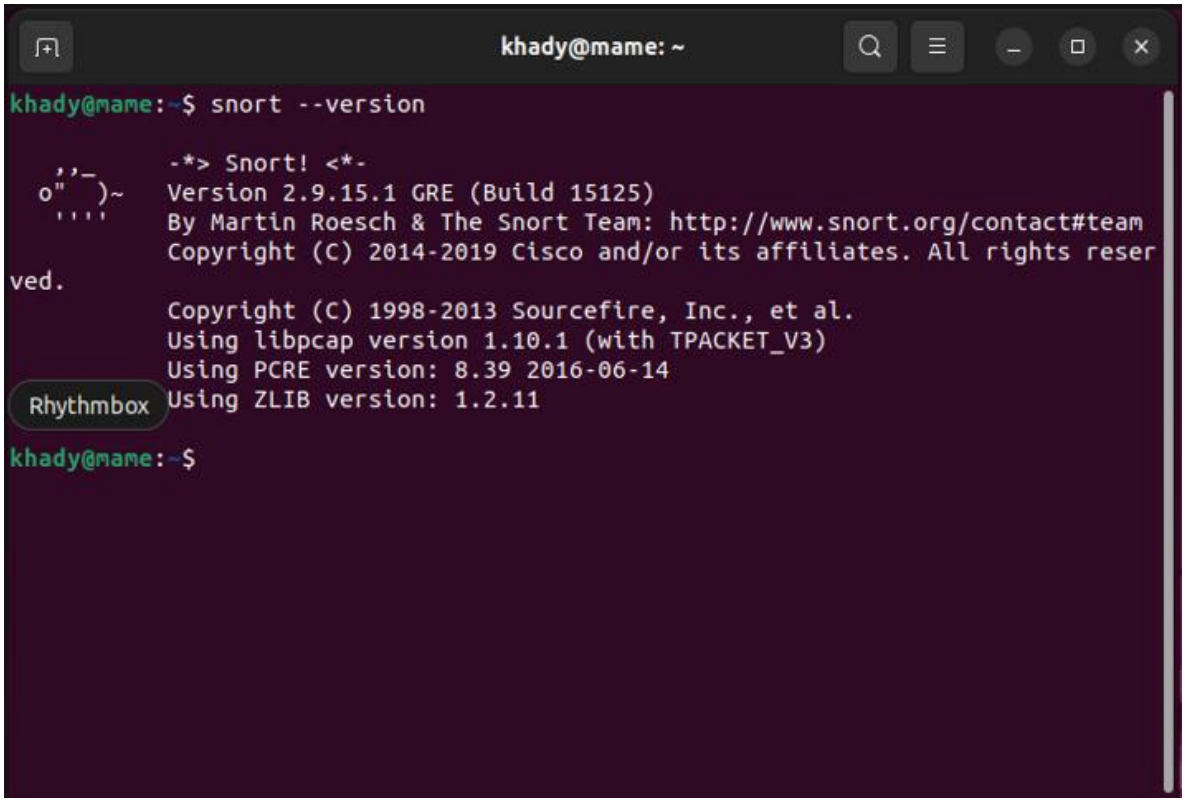
Test et fonctionnement du Site



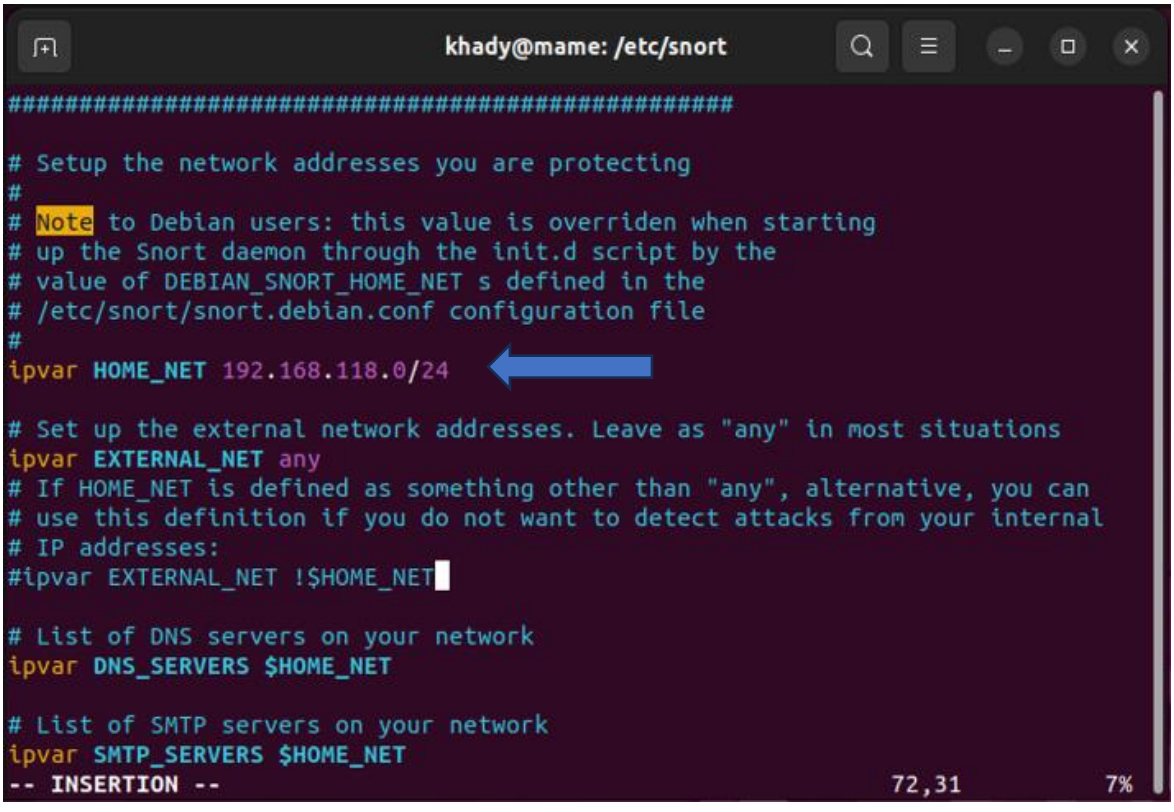
INSTALLATION, CONFIGURATION ET INTEGRATION DE SNORT DANS NOTRE RESEAU



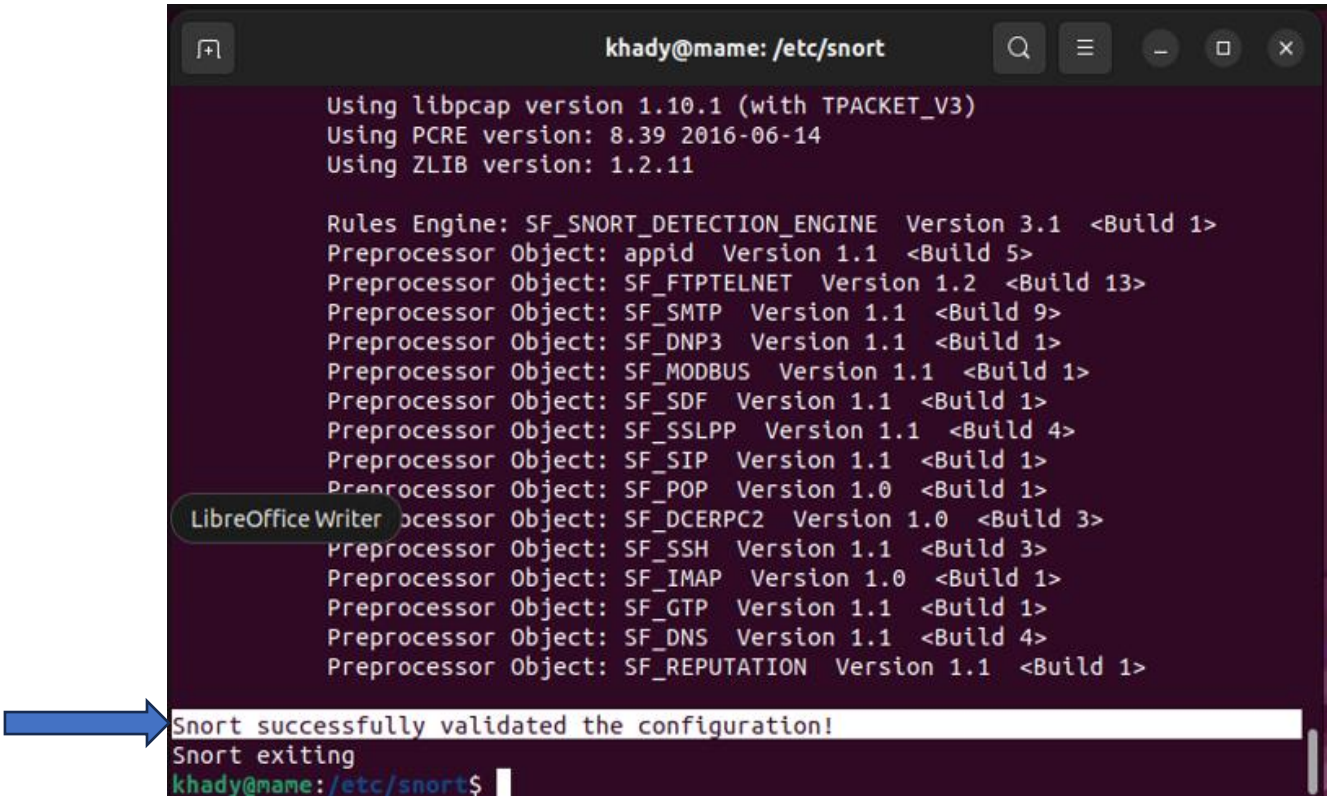
sudo apt install snort : La commande apt = aptitude qui est une commande qui permet d'installer un paquet dans un environnement Linux (Ubuntu, Debian).



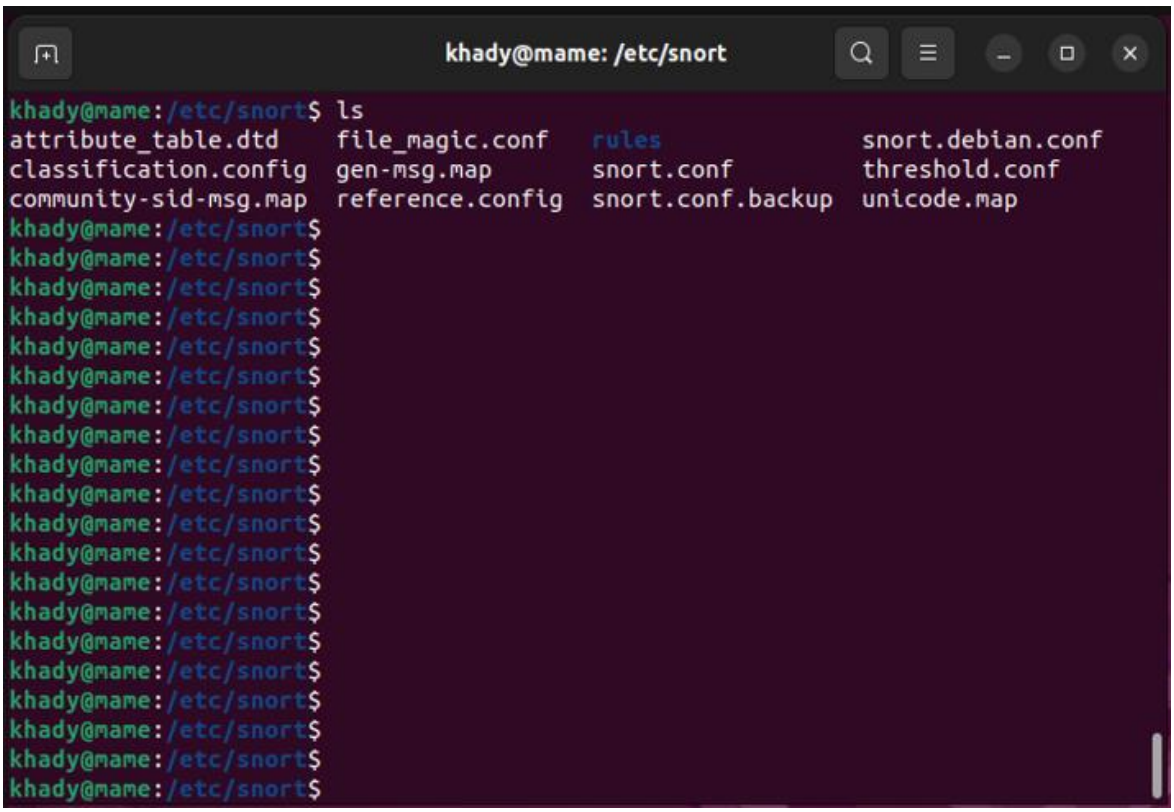
Snort - -version : Pour vérifier la version de notre snort installé

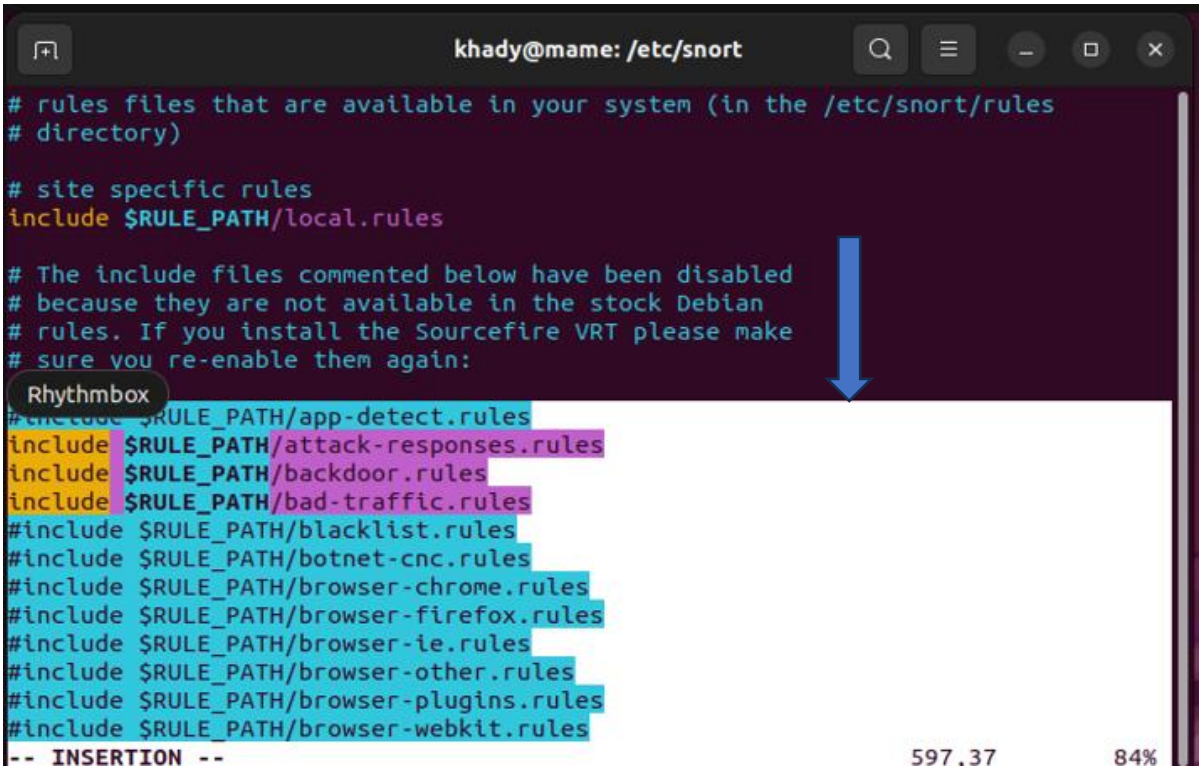


Ajoutons l'adresse réseau au niveau de : ipvar HOME_NET : 192.168.118.0/24

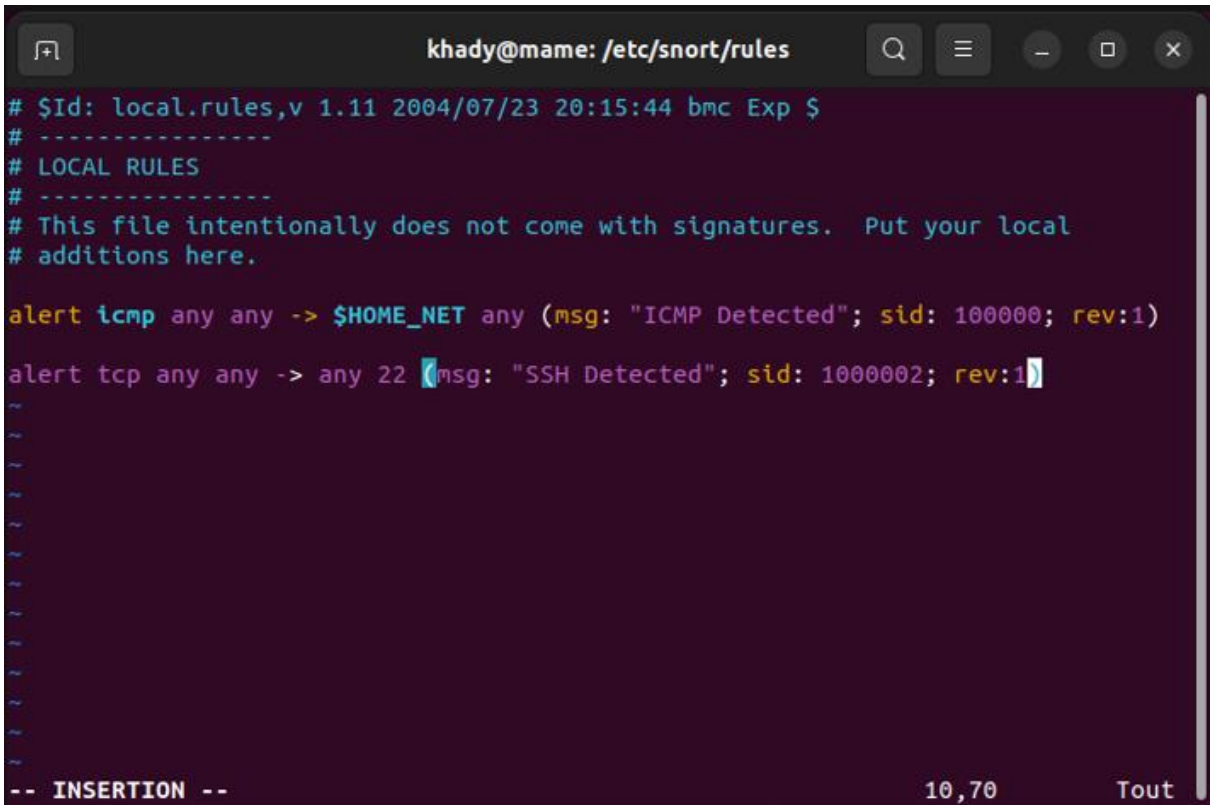


Test de configuration Snort : Configuration de Snort est validé avec succès.



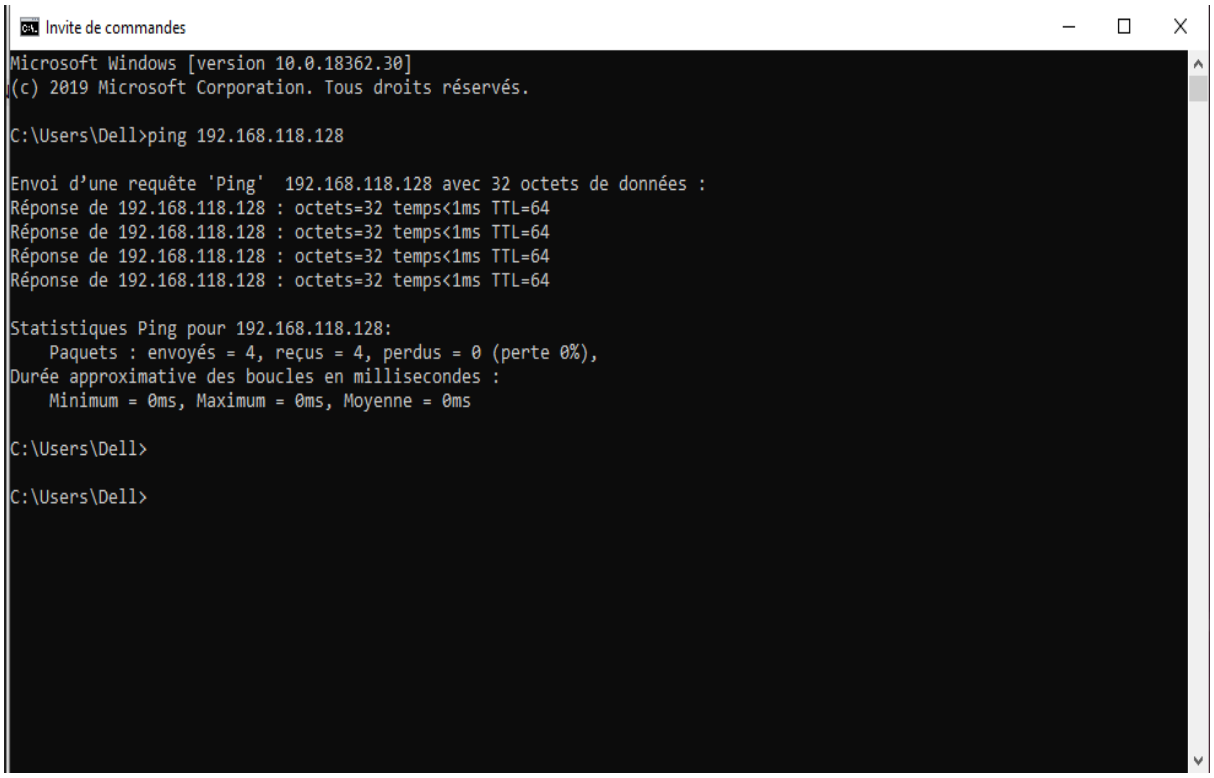


On supprime toutes les règles qui s'y trouvent dans /etc/snort/local.rules

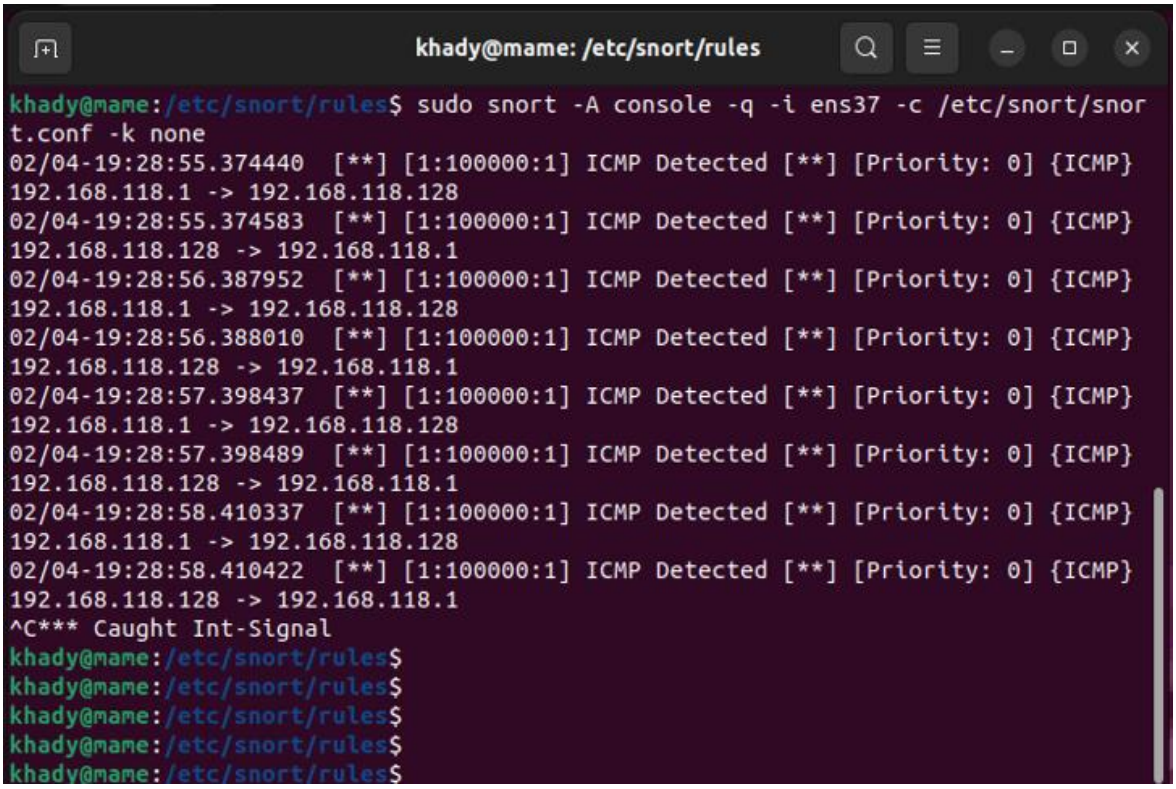


On personnalise nous-même nos règles pour mieux contrôler le fonctionnement de Snort.

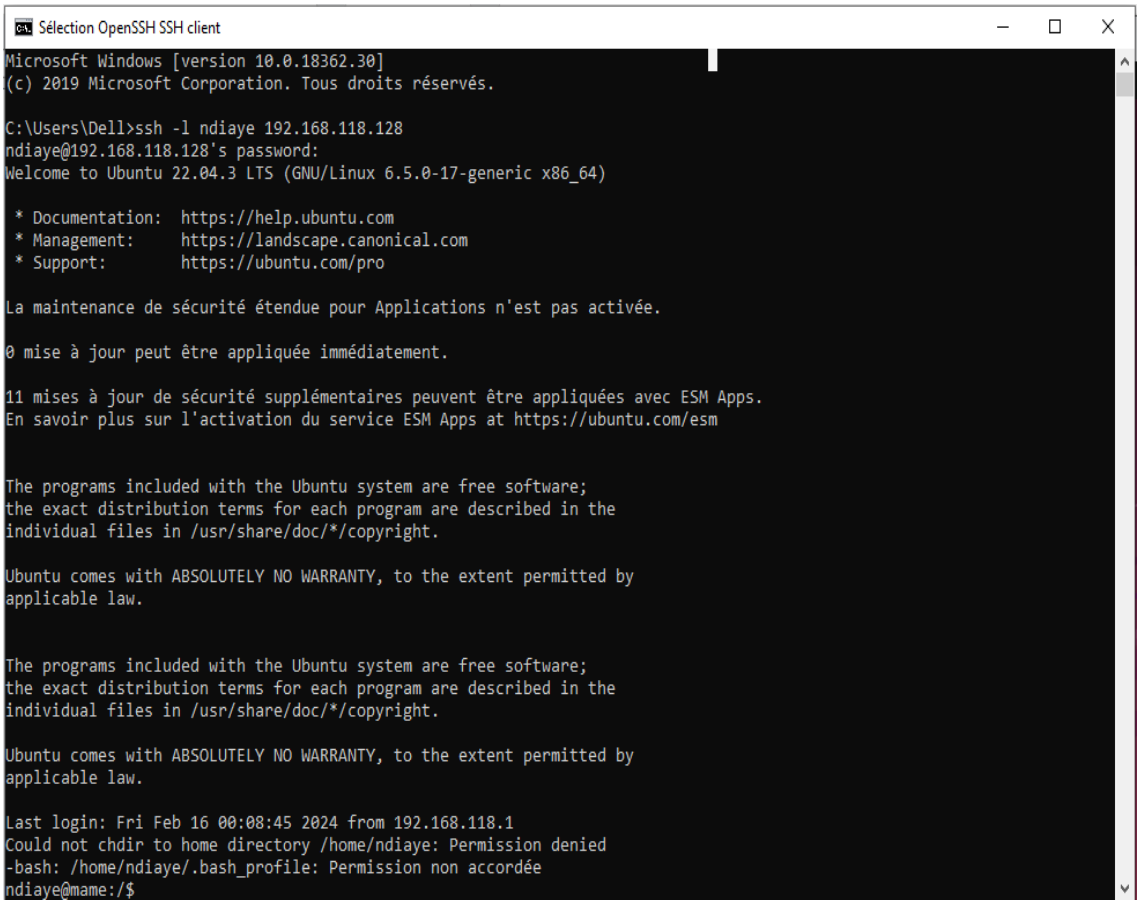
1ère Partie : Test IDS/IPS de l'interface physique vers l'interface virtuelle de Ubuntu avec Snort



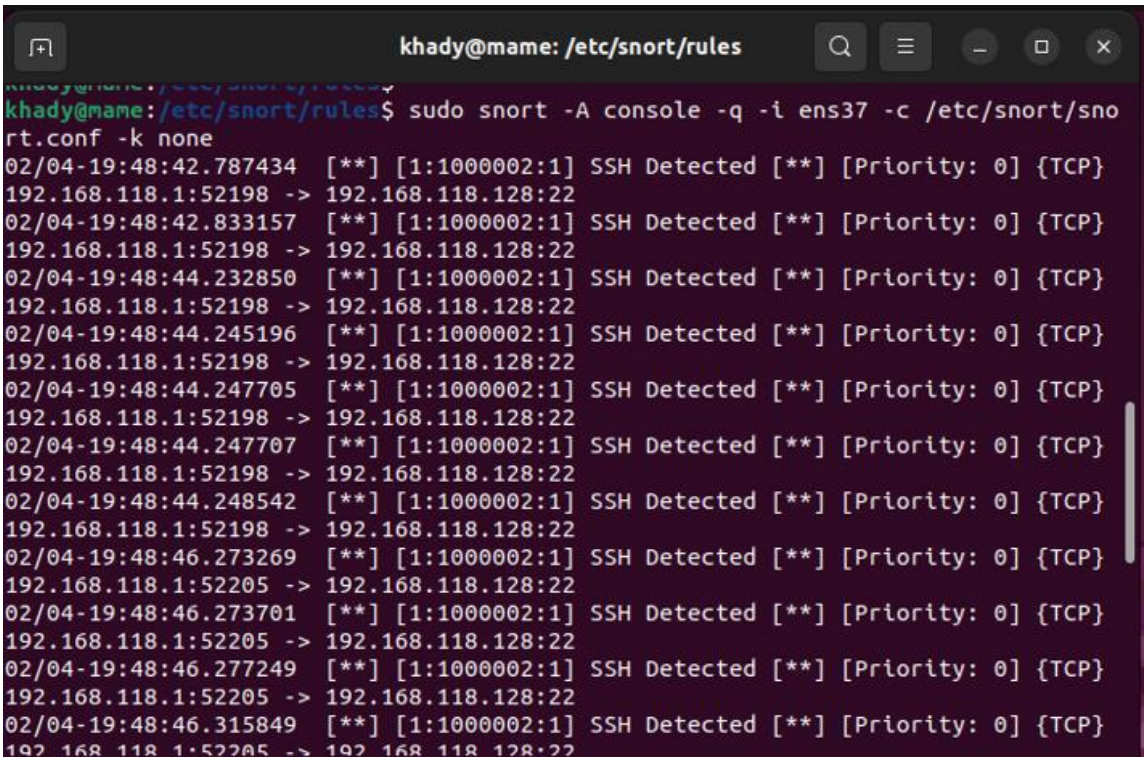
L'envoi des pings à l'adresse de notre serveur 192.168.118.128 à laquelle snort est configuré alors 192.168.118.1 est l'adresse physique de UBUNTU-SERVER puisque les deux adresses IP sont dans le même réseau.



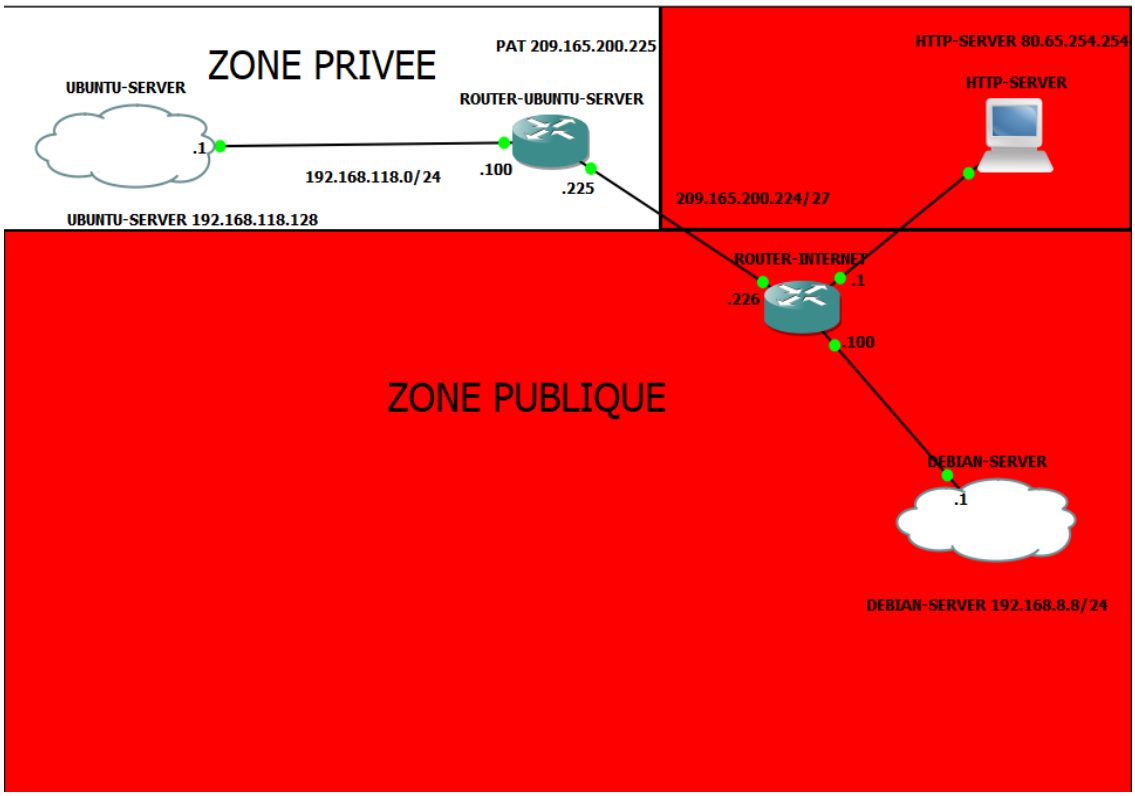
Détection d'intrusion à l'adresse 192.168.118.1 de l'interface physique vers le serveur Ubuntu à l'adresse 192.168.118.128. 192.168.118.1 pings provenant de l'interface physique.



Connexion ssh : du PC physique 192.168.118.1 vers le serveur Ubuntu 192.168.118.128 vers le shell de l'utilisateur ndiaye (serveur Ubuntu)



2ème PARTIE : TEST SNORT avec le simulateur réseau GNS3



ROUTER-INTERNET : Configuration et adressage IP sur les interfaces : f0/0, f0/1, f3/0

ROUTER-INTERNET

```
inistratively down
*Feb 14 16:10:00.015: %LINK-5-CHANGED: Interface Serial2/3, changed state to adm
inistratively down
ROUTER-INTERNET#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ROUTER-INTERNET(config)#int f0/0
ROUTER-INTERNET(config-if)#no sh
ROUTER-INTERNET(config-if)#ip add
*Feb 14 16:13:03.747: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Feb 14 16:13:04.747: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
ROUTER-INTERNET(config-if)#ip add 192.168.8.100 255.255.255.0
ROUTER-INTERNET(config-if)#ex
ROUTER-INTERNET(config)#int f0/0
ROUTER-INTERNET(config-if)#int f0/1
ROUTER-INTERNET(config-if)#no sh
ROUTER-INTERNET(config-if)#ip add
*Feb 14 16:13:36.059: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Feb 14 16:13:37.059: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
ROUTER-INTERNET(config-if)#ip add 80.65.254.1 255.255.255.0
ROUTER-INTERNET(config-if)#ex
```

IP route 0.0.0.0 0.0.0.0 209.165.200.225 -> route statique par défaut vers ROUTER-UBUNTU SERVER.

ROUTER-INTERNET

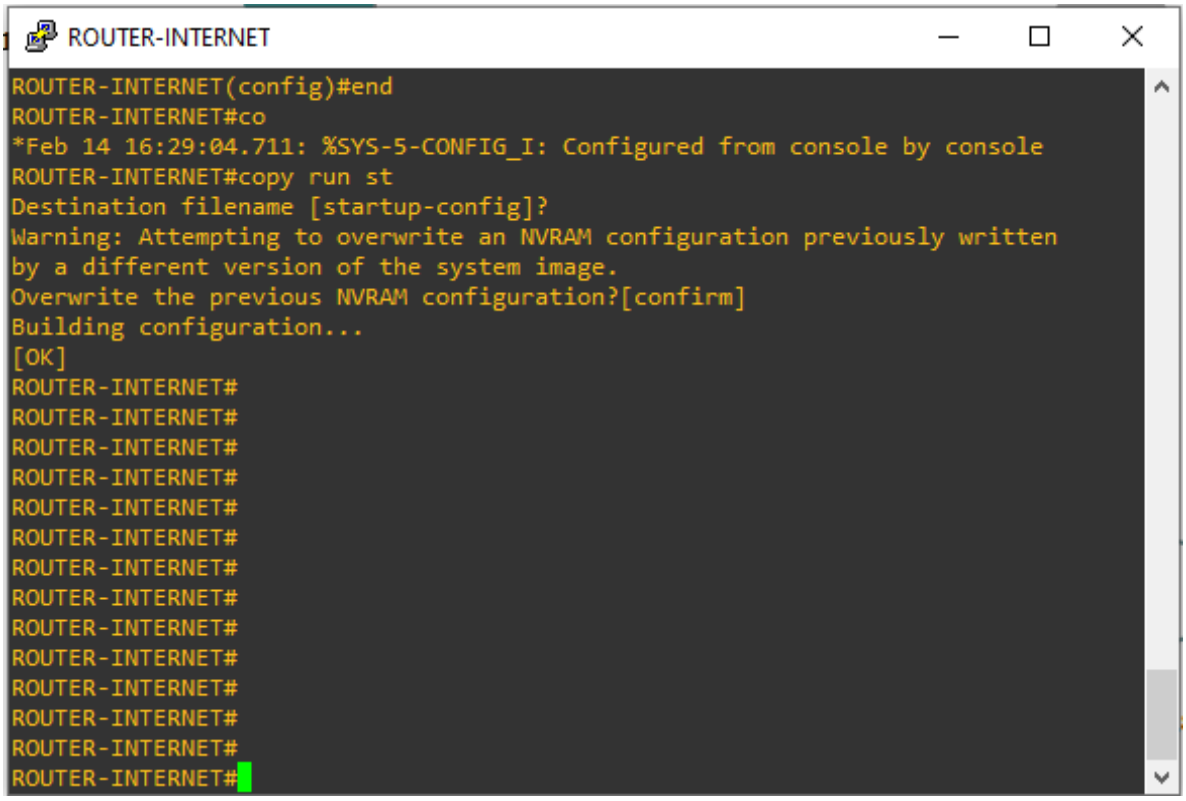
```
ROUTER-INTERNET(config)#int f3/0
ROUTER-INTERNET(config-if)#no sh
ROUTER-INTERNET(config-if)#ip add 209.
*Feb 14 16:13:55.059: %LINK-3-UPDOWN: Interface FastEthernet3/0, changed state t
o up
*Feb 14 16:13:56.059: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et3/0, changed state to up
ROUTER-INTERNET(config-if)#ip add 209.165.200.226 255.255.255.0
ROUTER-INTERNET(config-if)#ip add 209.165.200.226 255.255.255.224
ROUTER-INTERNET(config-if)#
ROUTER-INTERNET(config-if)#ex
ROUTER-INTERNET(config)#
ROUTER-INTERNET(config)#
*Feb 14 16:15:41.859: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et3/0, changed state to down
ROUTER-INTERNET(config)#
*Feb 14 16:16:20.859: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et3/0, changed state to up
ROUTER-INTERNET(config)#
ROUTER-INTERNET(config)#
ROUTER-INTERNET(config)#
ROUTER-INTERNET(config)#
ROUTER-INTERNET(config)#
ROUTER-INTERNET(config)#ip route 0.0.0.0 0.0.0.0 209.165.200.225
```

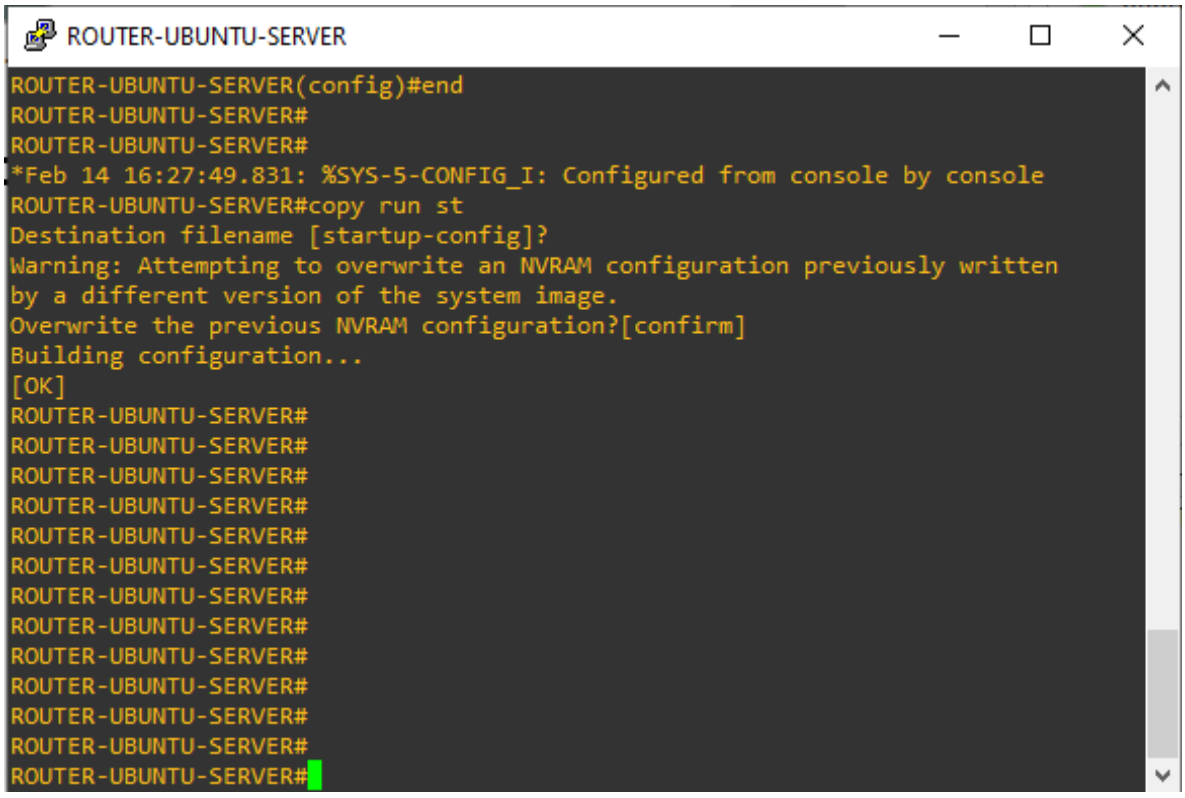
IP route 0.0.0.0 0.0.0.0 209.165.200.226 -> route statique par défaut vers ROUTER-INTERNET

ROUTER-UBUNTU-SERVER

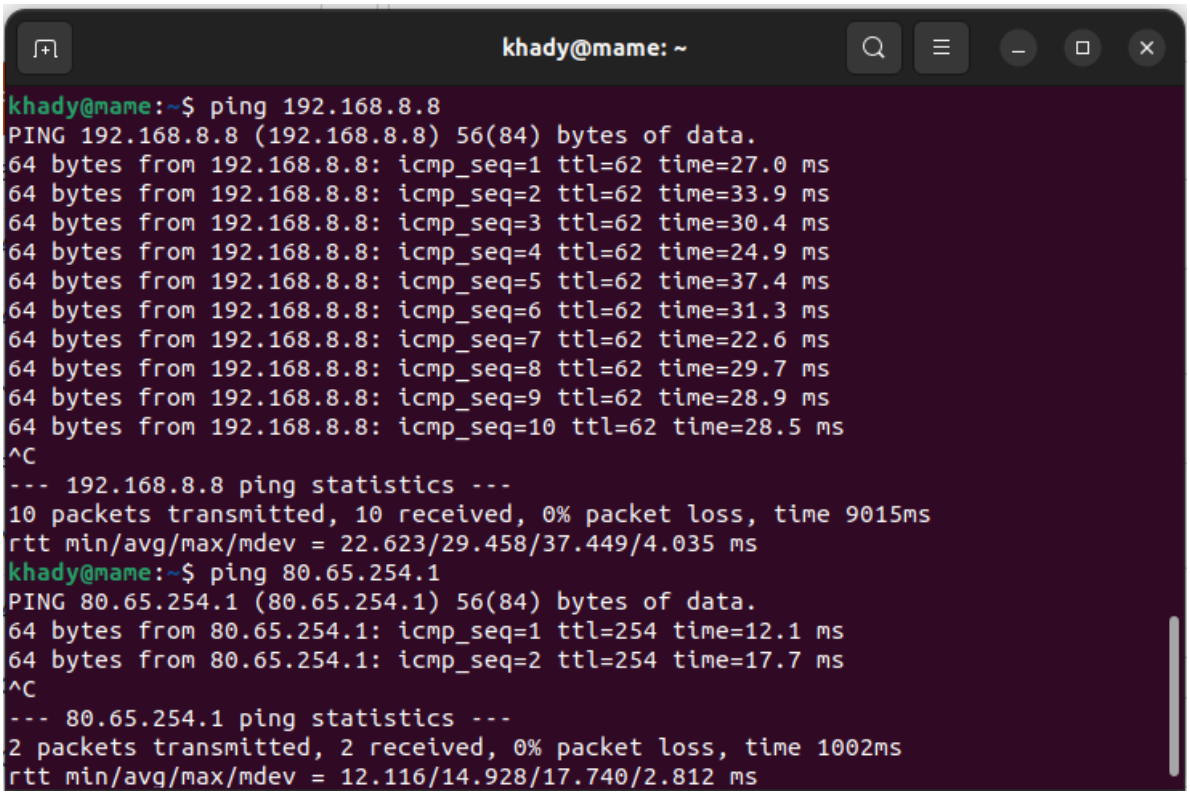
```
inistratively down
ROUTER-UBUNTU-SERVER#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ROUTER-UBUNTU-SERVER(config)#int f0/0
ROUTER-UBUNTU-SERVER(config-if)#no sh
ROUTER-UBUNTU-SERVER(config-if)#ip add 192.
*Feb 14 16:23:26.987: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
ROUTER-UBUNTU-SERVER(config-if)#ip add 192.168.
*Feb 14 16:23:27.987: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
ROUTER-UBUNTU-SERVER(config-if)#ip add 192.168.118.100 255.255.255.0
ROUTER-UBUNTU-SERVER(config-if)#ex
ROUTER-UBUNTU-SERVER(config)#int f3/0
ROUTER-UBUNTU-SERVER(config-if)#ip add 209.165.200.225 255.255.255.224
ROUTER-UBUNTU-SERVER(config-if)#no sh
ROUTER-UBUNTU-SERVER(config-if)#exit
ROUTER-UBUNTU-SERVER(config)#
*Feb 14 16:24:10.927: %LINK-3-UPDOWN: Interface FastEthernet3/0, changed state t
o up
*Feb 14 16:24:11.927: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et3/0, changed state to up
ROUTER-UBUNTU-SERVER(config)#ip route 0.0.0.0 0.0.0.0 209.165.200.226
ROUTER-UBUNTU-SERVER(config)#
```

- Sauvegarde de la configuration sur le ROUTER-INTERNET et ROUTER-UBUNTU-SERVER

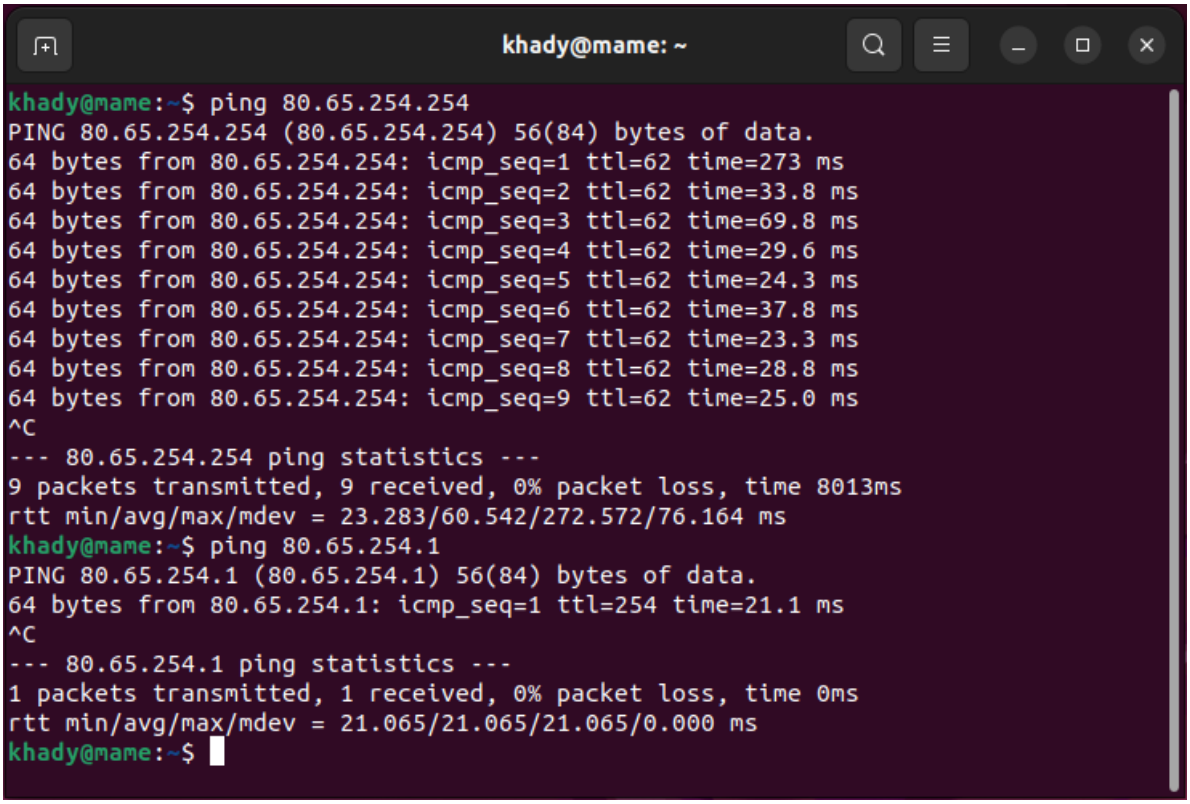




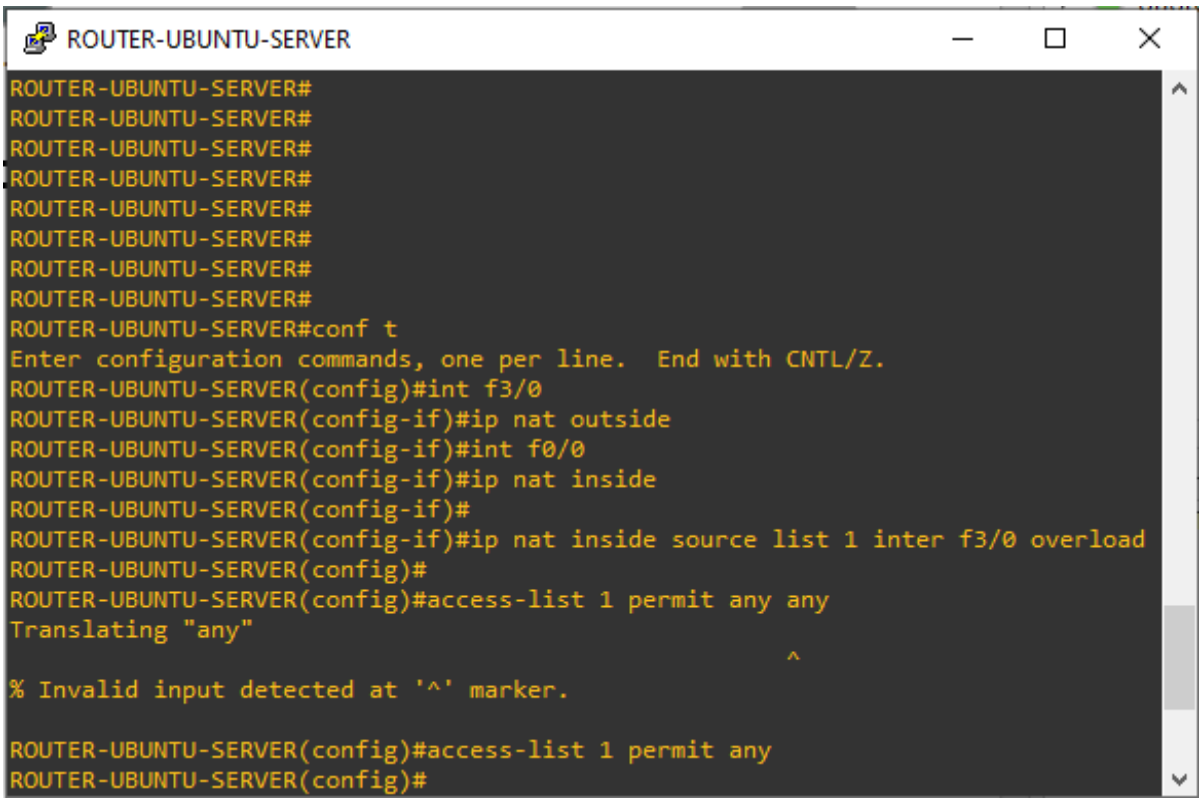
TEST PING VERS LE SERVEUR DEBIAN



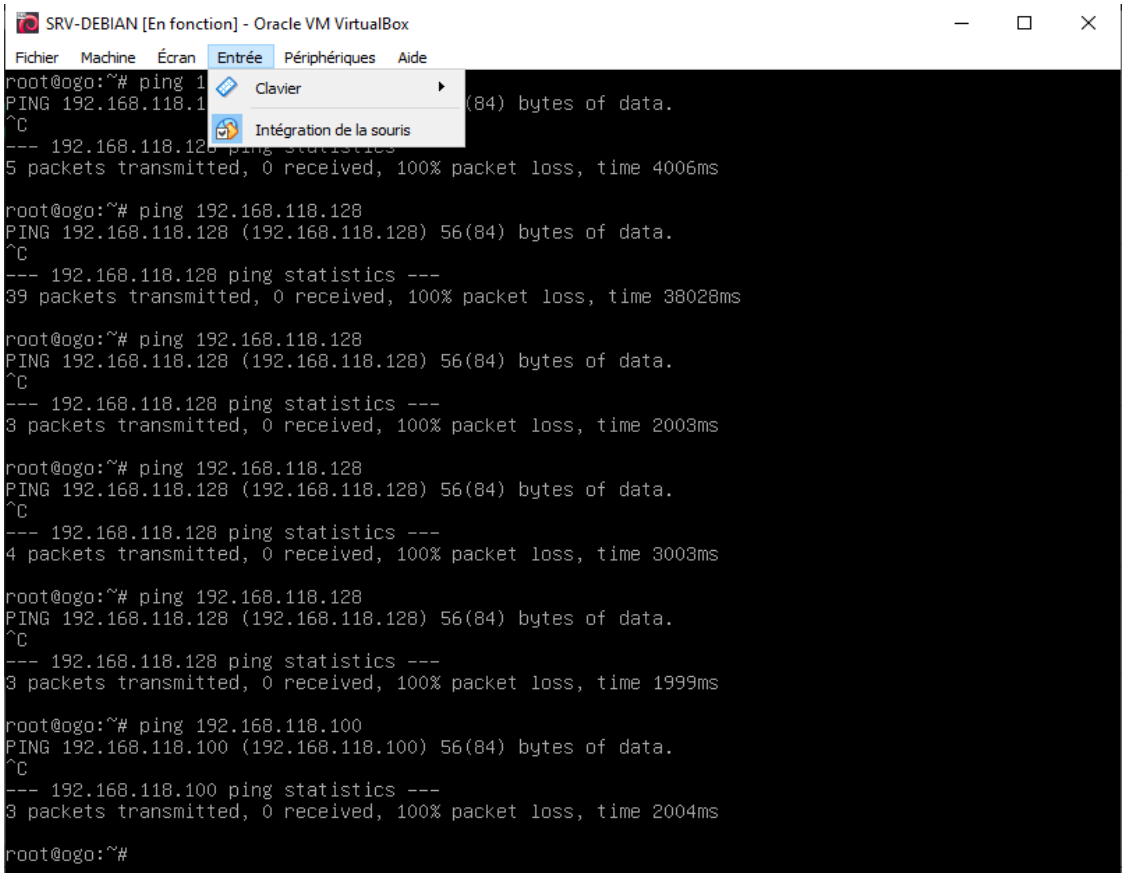
TEST PING VERS LE SERVEUR HTTP



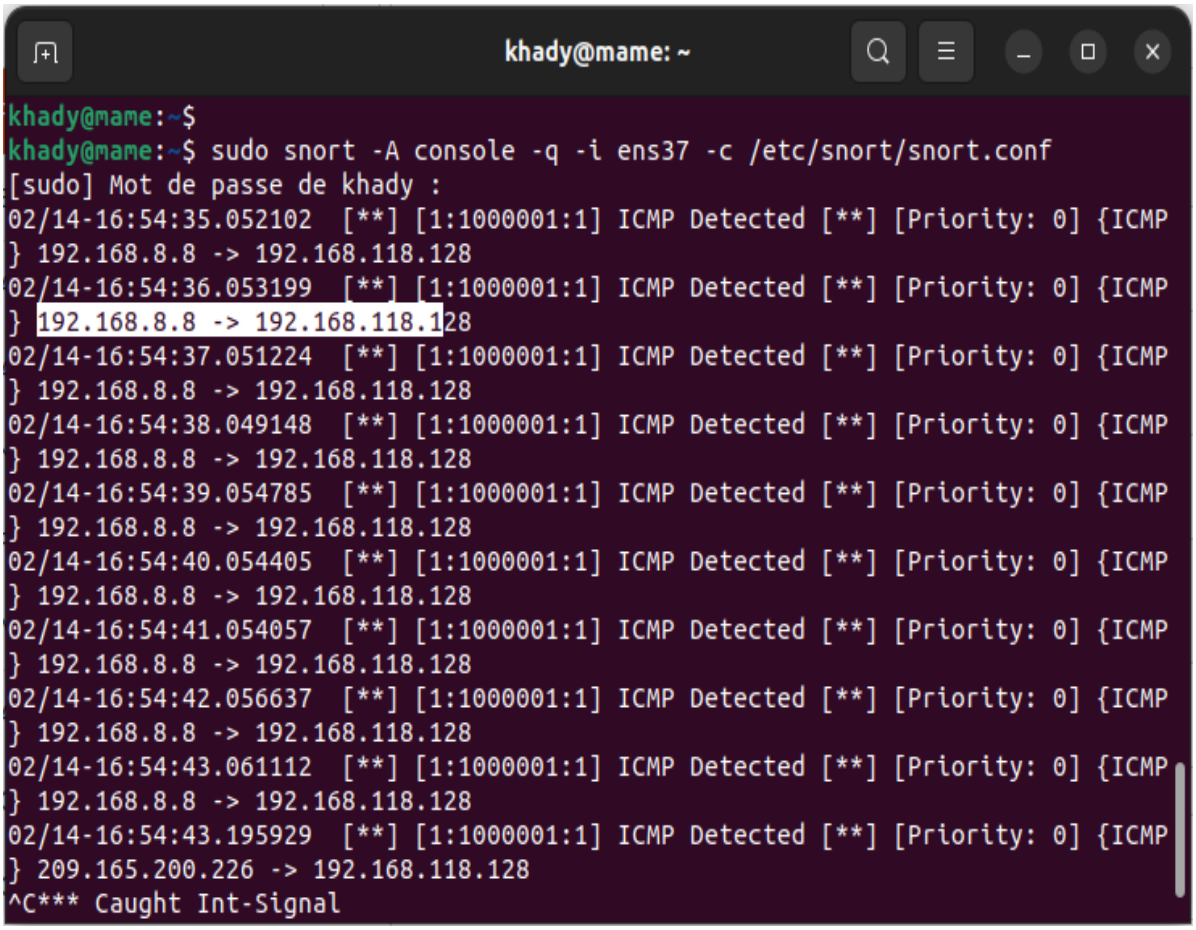
CONFIGURATION DU PAT (Port Address Translation) SUR LE ROUTER-UBUNTU-SERVER



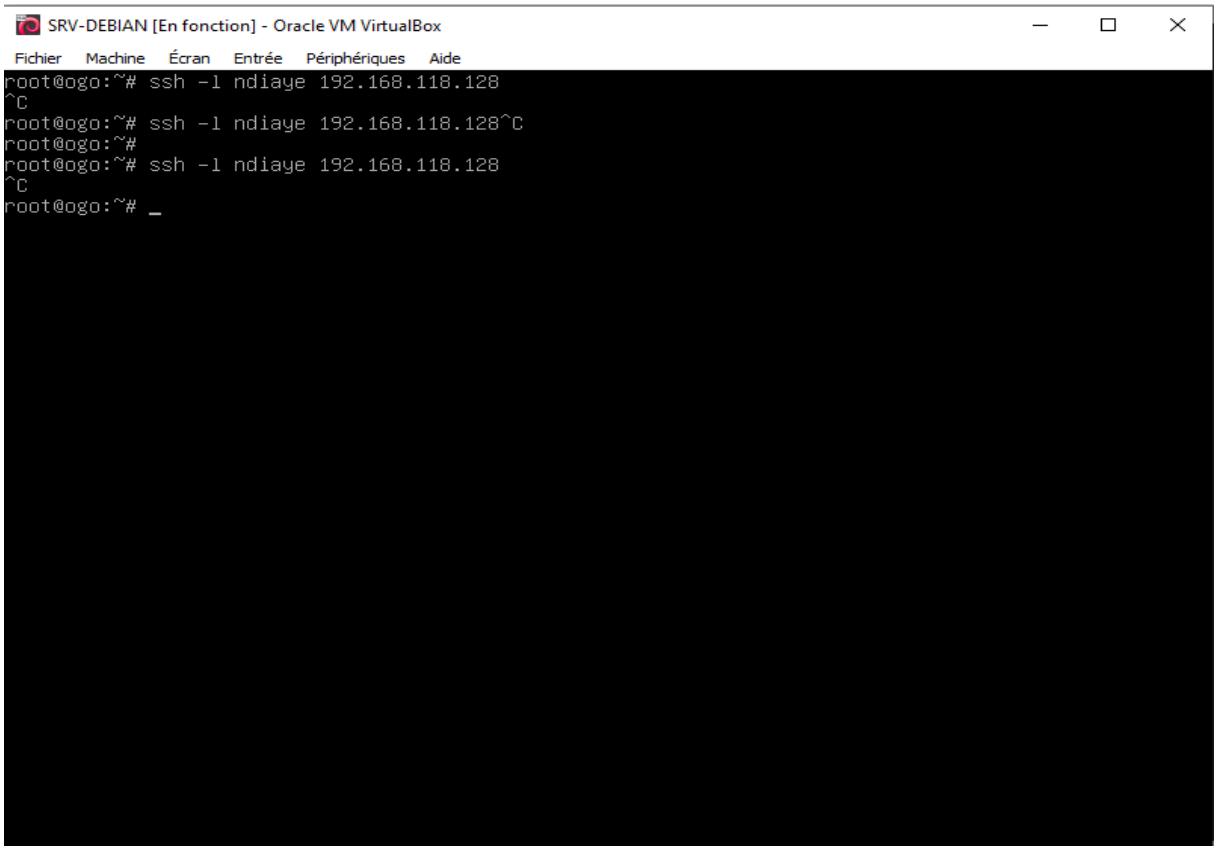
PING DE DEBIAN-SERVER VERS UBUNTU-SERVER



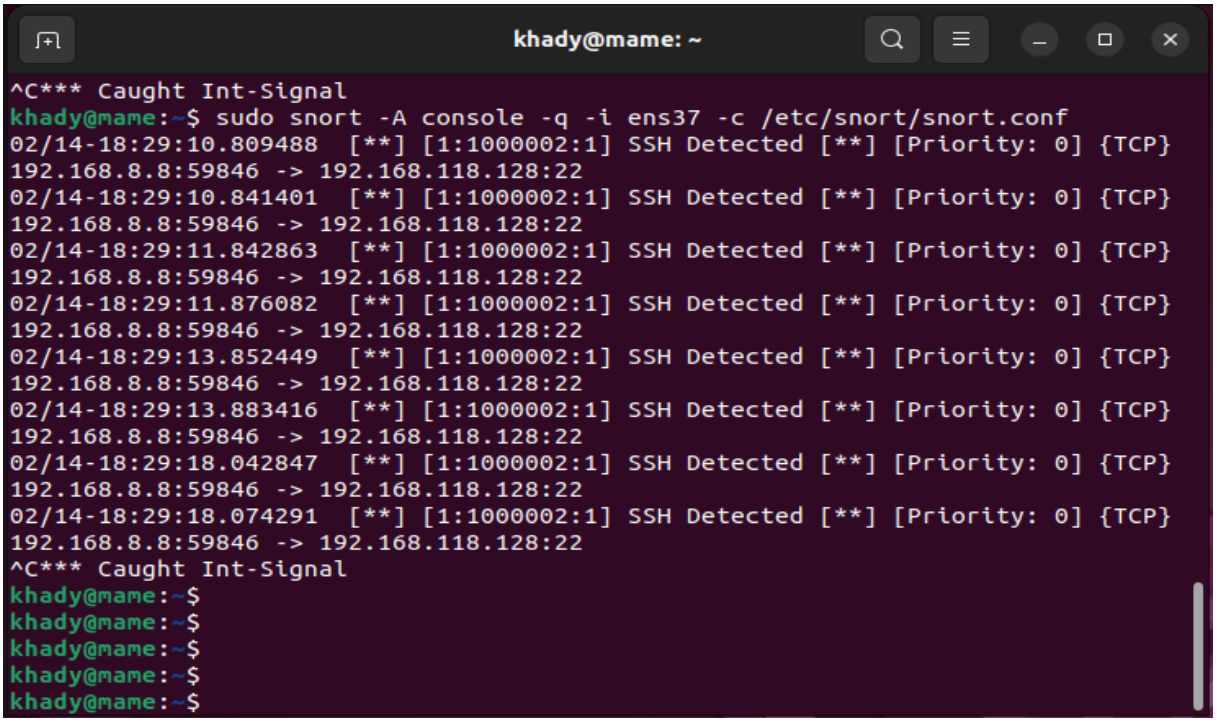
Si on voit bien les pings vers UBUNTU-SERVER ne passent pas, pourquoi parce que PAT Bloque déjà l'accès à notre adresse IP privée alors le PAT fait la translation des adresses IP provenant de la ZONE PUBLIQUE vers la ZONE PRIVEE avec l'IP 209.165.200.225.



Détection des paquets ICMP provenant du DEBIAN-SERVER vers notre UBUNTU-SERVER malgré que les pings ne passent mais cela ne montre pas que UBUNTU-SERVER n'a pas reçu les pings provenant du DEBIAN-SERVER, juste que les ICMP sont bloqués au niveau du PAT et n'autorise pas les pings vers la ZONE PRIVEE.



Ci-dessus est la tentative de connexion SSH vers notre ZONE PRIVEE qui ne se passe pas comme prévu, toujours le PAT bloque l'accès à DEBIAN-SERVER d'y avoir accès au service SSH via UBUNTU-SERVER, pour que DEBIAN-SERVER n'accède pas au shell de l'utilisateur ndiaye.



Détection de connexion SSH provenant du DEBIAN-SERVER malgré qu'il n'ait pas pu se connecter au shell de l'utilisateur ndiaye de UBUNTU-SERVER.

CONCLUSION

Ce projet nous a permis de configurer Snort et d'évaluer son efficacité à travers des tests en environnement simulé. Son bon fonctionnement repose sur une configuration rigoureuse et une mise à jour continue des règles de détection. Les simulations sous GNS3 ont souligné l'importance d'un paramétrage précis pour une détection optimale des menaces. Ainsi, Snort constitue un outil puissant pour la sécurisation des réseaux, à condition d'être intégré dans une stratégie de cybersécurité proactive.

FIN