

# SSH (Secure Shell) Linux

Réalisé par :

EL Ouardi Mohamed  
EL Omari Zakaria  
Taabani Taha Yassine

Encadré par :

Mr.Moukhafi

# **Table de contenu :**

**01**

**Introduction au  
Réseau**

**02**

**Positionnement de  
SSH dans les Modèles  
de Référence**

**03**

**Introduction à SSH  
(Secure Shell)**

**04**

**Simulation de  
Contrôle avec SSH**

# **Table de contenu :**

**05**

**Introduction à  
Wireshark**

**06**

**Analyse des  
Résultats**

**07**

**Conclusion**

# 01

# Introduction

Au Réseau informatique



## A .C'est quoi un réseau informatique

- Un réseau informatique est une structure qui permet l'interconnexion de systèmes et de dispositifs afin de faciliter le partage de ressources et d'informations entre eux.
- L'objectif principal : est de permettre la communication efficace et la collaboration entre les différents éléments du réseau, favorisant ainsi le partage de données, de fichiers et de périphériques.

## B. Modèles de référence

# TCP/IP vs OSI

Le modèle TCP/IP est un ensemble de protocoles largement utilisé pour les communications sur Internet. Il est divisé en quatre couches, souvent regroupées en deux catégories : la couche hôte et la couche réseau.

Le modèle OSI est une norme de référence internationale pour la conception et le fonctionnement des réseaux informatiques. Il est divisé en sept couches, chacune ayant des fonctions spécifiques.

### Modèle OSI

7. Application

6. Présentation

5. Session

4. Transport

3. Réseau

2. Liaison de données

1. Physique

**Gère la traduction, la compression et le chiffrement des données.**

**Gère la fiabilité de la communication.**

**Gère l'accès au support physique, la détection d'erreurs, et le contrôle de flux.**

**Fournit des interfaces pour les applications réseau.**

**Établit, maintient, et termine les sessions entre les applications.**

**Routage des données à travers le réseau.**

**Responsable de la transmission brute des bits sur un support physique.**

## Modèle TCP/IP

Application

Transport

Internet

Accès  
réseau

- Correspondant aux couches session, présentation et application du modèle OSI.
- Fournit des services réseau directement aux applications utilisateur.

- Correspond à la couche transport du modèle OSI.
- Gère le transport des données de bout en bout.

- Correspond à la couche réseau du modèle OSI.
- Responsable du routage des paquets à travers le réseau.
- Correspond aux couches physiques et de liaison de données du modèle OSI.
- Gère l'accès au support physique et la transmission des données.



# 02

## **Positionnement de SSH dans Les Modèles de Référence**



# Modèle OSI

## Identifier la couche correspondante pour SSH dans le modèle OSI :

- SSH (Secure Shell) se positionne principalement au niveau de la couche application du modèle OSI.
- Il opère au-dessus des couches de transport (par exemple, TCP) et de réseau (par exemple, IP), assurant ainsi une sécurisation des données à un niveau élevé dans la pile de protocoles.

## Brève explication du rôle de cette couche :

- La couche application du modèle OSI est la couche la plus proche de l'utilisateur final et des applications logicielles. C'est à ce niveau que les protocoles fournissent des services de communication directement aux applications.
- SSH, en tant que protocole de la couche application, offre un moyen sécurisé d'accéder à distance à des systèmes et de transférer des données de manière cryptée. Il facilite l'authentification et la confidentialité des communications entre les utilisateurs et les systèmes distants.

---

# Modèle TCP/IP

## Localiser SSH dans les couches du modèle TCP/IP

- Dans le modèle TCP/IP, SSH s'insère également au niveau de la couche application, correspondant à la couche d'application du modèle OSI.
- Il utilise des protocoles de transport sous-jacents tels que TCP pour assurer la fiabilité de la communication.

## Mettre en avant le rôle de SSH dans le contexte du modèle TCP/IP :

- La couche application du modèle TCP/IP englobe les protocoles qui fournissent des services directs aux applications utilisateur. SSH, en tant que protocole de cette couche, joue un rôle crucial dans la sécurisation des communications.
  - En utilisant le chiffrement et l'authentification forte, SSH assure un échange sécurisé d'informations entre un client et un serveur, que ce soit pour l'accès à distance, le transfert de fichiers, ou d'autres opérations réseau
-

---

# Modèle TCP/IP

- SSH utilise généralement le protocole TCP pour le transport fiable des données. Cela garantit que les informations échangées entre les parties sont intégrées, confidentielles et qu'elles parviennent à destination sans altération.
  - En résumé, SSH occupe une place stratégique au niveau de la couche application dans les modèles OSI et TCP/IP, offrant une sécurité robuste pour les communications réseau, en particulier lorsqu'il s'agit d'accès distant et de transfert de données sensibles.
-

# 03

## **Introduction à SSH (Secure Shell)**



## A. Définition de SSH

### Protocole de communication sécurisé:


- SSH, qui signifie "Secure Shell", est un protocole de communication sécurisé conçu pour permettre l'accès sécurisé à des systèmes distants sur un réseau non sécurisé.

- Il fournit un canal sécurisé sur une connexion non sécurisée, typiquement l'Internet, en utilisant des techniques de chiffrement pour protéger les données transitant entre le client et le serveur.

### Histoire de SSH:

SSH, développé en 1995 par Tatu Ylönen, est un protocole de sécurité pour les communications à distance. Évoluant de la version SSH-1 à la norme SSH-2, il a introduit le cryptage des données pour remédier aux failles de protocoles antérieurs comme Telnet. Malgré des vulnérabilités initiales, SSH-2 a renforcé la sécurité et la flexibilité. Devenu indispensable pour l'administration système et les transferts de fichiers sécurisés, il reste à jour grâce à des implémentations open source comme OpenSSH, demeurant ainsi un outil essentiel pour des connexions réseau sûres.

## Utilisé pour l'accès distant et la gestion de systèmes :

- SSH est largement utilisé pour l'accès distant à des systèmes, permettant aux utilisateurs de se connecter à des serveurs distants de manière sécurisée.
  - En plus de l'accès distant, SSH est également utilisé pour la gestion sécurisée des systèmes, le transfert de fichiers sécurisé (SFTP), et l'exécution de commandes à distance de manière sécurisée.
- 

## **B. Fonctionnalités de sécurité de SSH**

### **Chiffrement des données :**

- L'une des principales fonctionnalités de sécurité de SSH est le chiffrement des données. Les informations échangées entre le client SSH et le serveur SSH sont cryptées, rendant extrêmement difficile pour des tiers non autorisés d'intercepter et de comprendre le contenu des communications.
- Le chiffrement est appliqué à toutes les données, y compris les commandes, les informations d'identification, et tout autre trafic entre le client et le serveur.

### **Authentification forte :**

- SSH utilise un mécanisme d'authentification forte, ce qui signifie que l'identité des parties en communication est vérifiée de manière robuste.
- Il prend en charge plusieurs méthodes d'authentification, telles que les clés publiques/privées, les mots de passe, et même l'authentification à deux facteurs. Ces méthodes renforcent la sécurité en s'assurant que seules les parties autorisées ont accès au système.
- L'utilisation de clés publiques/privées est courante dans SSH. Dans ce cas, une clé publique est partagée avec le serveur, et la clé privée est conservée par l'utilisateur. L'authentification se fait en prouvant la possession de la clé privée.



## C. Installation SSH (Linux)

```
> sudo pacman -S openssh
warning: openssh-9.5p1-1 is up to date -- reinstalling
resolving dependencies...
looking for conflicting packages...

Packages (1) openssh-9.5p1-1

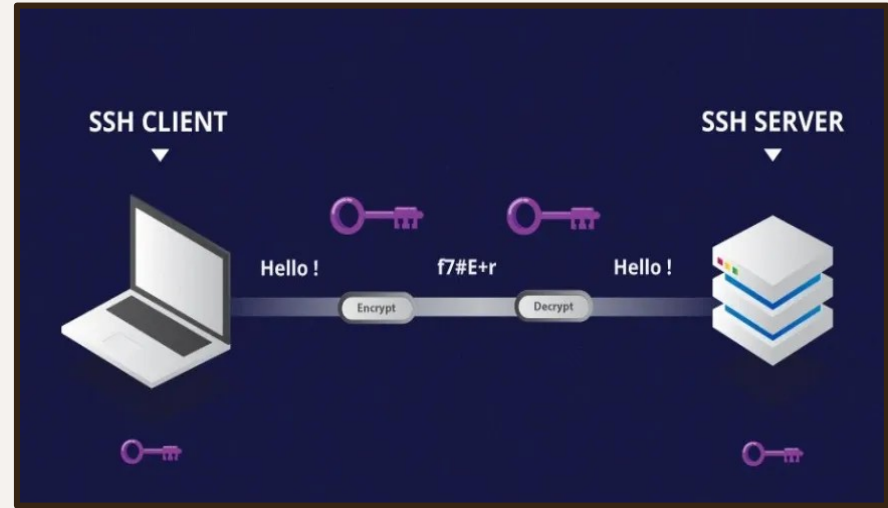
Total Installed Size: 4.90 MiB
Net Upgrade Size:      0.00 MiB

:: Proceed with installation? [Y/n]
(1/1) checking keys in keyring [#####] 100%
(1/1) checking package integrity [#####] 100%
(1/1) loading package files [#####] 100%
(1/1) checking for file conflicts [#####] 100%
(1/1) checking available disk space [#####] 100%
:: Processing package changes...
(1/1) reinstalling openssh [#####] 100%
:: Running post-transaction hooks...
(1/3) Reloading system manager configuration...
(2/3) Creating temporary files...
(3/3) Arming ConditionNeedsUpdate...
```

**Comment installer  
ssh dans Arch Linux**

\$ sudo apt install openssh-server	[On Debian, Ubuntu and Mint]
\$ sudo yum install openssh	[On RHEL/CentOS/Fedora and Rocky/AlmaLinux]
\$ sudo emerge -a openssh	[On Gentoo Linux]
\$ sudo apk add openssh	[On Alpine Linux]
\$ sudo pacman -S openssh	[On Arch Linux]
\$ sudo zypper install openssh	[On OpenSUSE]

**En résumé, SSH offre un environnement de communication sécurisé en chiffrant les données échangées et en utilisant des méthodes d'authentification forte pour s'assurer de l'identité des parties impliquées. Ces caractéristiques font de SSH un choix privilégié pour la gestion à distance sécurisée des systèmes et des communications sensibles.**



# 04

## **Simulation de Contrôle avec SSH**



## A. Mise en place d'une connexion SSH entre deux machines :


- L'administrateur (A) commence par établir une connexion SSH sécurisée avec le serveur distant (B) en utilisant un client SSH. Cela peut être réalisé en utilisant la commande SSH dans un terminal, en spécifiant l'adresse IP ou le nom de domaine du serveur, ainsi que les informations d'identification appropriées.

- Par exemple :

```
bash
```

```
ssh utilisateurA@adresse_IP_serveur
```

- Lors de la première connexion, l'utilisateur peut être invité à accepter la clé publique du serveur pour établir une relation de confiance. Une fois l'authentification réussie, l'utilisateur (A) a un accès distant sécurisé à la machine (B).



```
taha@taha:~$ systemctl start ssh
taha@taha:~$ systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ssh
update-rc.d: error: Permission denied
taha@taha:~$ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ssh
Created symlink /etc/systemd/system/ssh.service → /lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/systemd/system/ssh.service.
taha@taha:~$ systemctl status ssh
○ ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Drop-In: /etc/systemd/system/ssh.service.d
            └─00-socket.conf
   Active: inactive (dead)
 TriggeredBy: ● ssh.socket
    Docs: man:sshd(8)
          man:sshd_config(5)
   Main PID: 3321 (sshd)
     Tasks: 1 (limit: 9444)
    Memory: 1.4M
       CPU: 20ms
    CGroup: /system.slice/ssh.service
            └─3321 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Nov 13 22:13:55 taha systemd[1]: Starting ssh.service - OpenBSD Secure Shell se>
Nov 13 22:13:55 taha sshd[3321]: Server listening on :: port 22.
Nov 13 22:13:55 taha systemd[1]: Started ssh.service - OpenBSD Secure Shell ser>
```

```
taha@taha:~$ vim /etc/ssh/sshd_config
```

```
taha@taha:~$ ssh aizen@192.168.1.9
```

```
aizen@192.168.1.9's password:
```

```

      /\
     /\
    /\
   /\
  /\
 /\
/_-' ' _-\

aizen@Aizen
os      Archcraft
host    R530/R730
kernel  6.6.1-arch1-1
uptime  25m
pkgs     1142
memory  534M / 5862M
```

```
~ >
```

```
~ >
```

```
~ > mkdir pratique_ssh
```

```
~ >
```

```
~ >
```

```
~ > ls
```

```
Desktop  dotfiles  Installation  Music  Pictures  Public  RiceInstaller  sstest  Templates  usb  wayland-config
Documents Downloads learn  Myfiles  pratique_ssh  README.md  Scripts  tahaaa  tp1  Videos
```

```
~ > |
```

```
  /\
 /  \
/\    \
/      \
/        \
/          \
/            \
/_-''      ''-_ \


aizen@Aizen
os      Archcraft
host    R530/R730
kernel  6.6.1-arch1-1
uptime  34m
pkgs     1142
memory  539M / 5862M
```

~ > ls

Desktop	Installation	Pictures	RiceInstaller	Templates	wayland-config
Documents	learn	pratique_ssh	Scripts	tp1	
dotfiles	Music	Public	sshtest	usb	
Downloads	Myfiles	README.md	tahaaa	Videos	

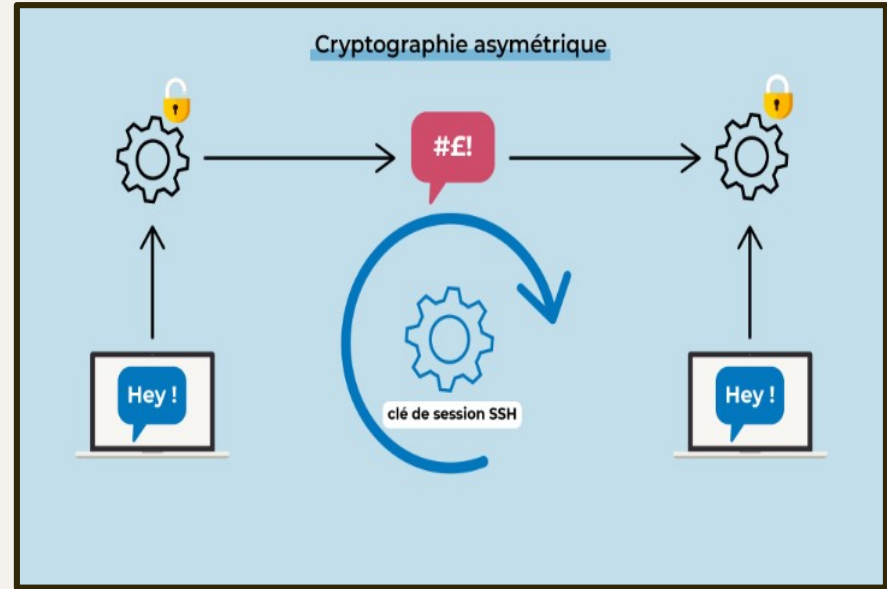
~ >

## **B. Illustration du chiffrement des données par SSH :**

- Pendant la démonstration, nous mettrons en évidence le chiffrement des données par SSH en utilisant des outils tels que Wireshark pour capturer le trafic réseau. La comparaison entre une connexion SSH sécurisée et une connexion non sécurisée soulignera l'efficacité du chiffrement.
  - Lors de l'analyse du trafic capturé, on peut observer que les données échangées entre le client et le serveur via SSH sont illisibles pour un observateur externe en raison du chiffrement. Cela confirmera visuellement comment SSH garantit la confidentialité des données pendant la communication.
- 



En résumé, la simulation de contrôle avec SSH permettra de démontrer concrètement comment SSH assure un accès distant sécurisé, en établissant une connexion chiffrée entre un utilisateur et une machine distante, renforçant ainsi la sécurité des opérations de gestion à distance.



---

**05**

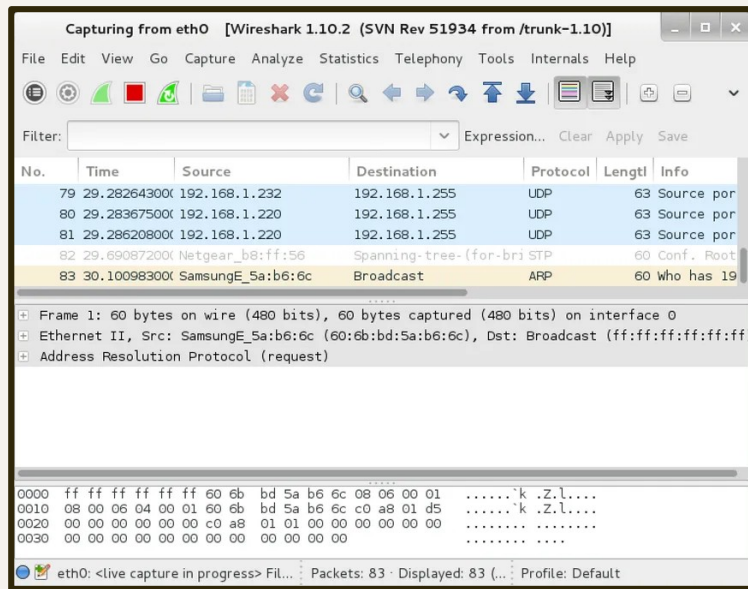
  
**WIRESHARK**



---


# A. Outil d'analyse de paquets réseau

**Wireshark** est un outil open-source d'analyse de paquets réseau qui permet de capturer, visualiser, et inspecter le trafic sur un réseau. Il offre une compréhension approfondie du fonctionnement des communications réseau en examinant les paquets de données qui transitent entre les différents points du réseau.



## **B . Outil d'analyse de paquets réseau**

### **Démonstration en temps réel de la capture du trafic non chiffré :**

- Pour la démonstration, Wireshark sera utilisé pour capturer le trafic entre le client et le serveur pendant une session non chiffrée. Cela peut être simulé en utilisant des configurations réseau où le chiffrement n'est pas activé.
  - L'attaquant, utilisant Wireshark, peut capturer les paquets de données échangés entre le client et le serveur. Ces paquets peuvent contenir des informations sensibles, et la démonstration permettra de visualiser comment ces données peuvent être facilement interceptées
- 

Ethernet

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ssh

No.	Time	Source	Destination	Protocol	Length	Info
46	39.981838	192.168.1.10	192.168.1.7	SSH	90	Client: Encrypted packet (len=36)
47	39.984721	192.168.1.7	192.168.1.10	SSH	90	Server: Encrypted packet (len=36)
49	40.101437	192.168.1.10	192.168.1.7	SSH	90	Client: Encrypted packet (len=36)
50	40.103173	192.168.1.7	192.168.1.10	SSH	90	Server: Encrypted packet (len=36)
52	40.437622	192.168.1.10	192.168.1.7	SSH	90	Client: Encrypted packet (len=36)
53	40.439166	192.168.1.7	192.168.1.10	SSH	106	Server: Encrypted packet (len=52)
54	40.453365	192.168.1.7	192.168.1.10	SSH	522	Server: Encrypted packet (len=468)
56	40.454718	192.168.1.7	192.168.1.10	SSH	106	Server: Encrypted packet (len=52)
57	40.464439	192.168.1.7	192.168.1.10	SSH	98	Server: Encrypted packet (len=44)
59	40.465064	192.168.1.7	192.168.1.10	SSH	122	Server: Encrypted packet (len=68)
269	191.004169	192.168.1.10	192.168.1.7	SSH	90	Client: Encrypted packet (len=36)
270	191.006720	192.168.1.7	192.168.1.10	SSH	106	Server: Encrypted packet (len=52)
271	191.009986	192.168.1.7	192.168.1.10	SSH	106	Server: Encrypted packet (len=52)
273	191.110662	192.168.1.7	192.168.1.10	SSH	130	Server: Encrypted packet (len=76)
277	191.252162	192.168.1.10	192.168.1.7	SSH	90	Client: Encrypted packet (len=36)
278	191.292099	192.168.1.7	192.168.1.10	SSH	106	Server: Encrypted packet (len=52)
279	191.292332	192.168.1.7	192.168.1.10	SSH	106	Server: Encrypted packet (len=52)
281	191.300025	192.168.1.7	192.168.1.10	SSH	130	Server: Encrypted packet (len=76)
283	191.396019	192.168.1.10	192.168.1.7	SSH	90	Client: Encrypted packet (len=36)
284	191.400104	192.168.1.7	192.168.1.10	SSH	122	Server: Encrypted packet (len=68)
285	191.414631	192.168.1.7	192.168.1.10	SSH	98	Server: Encrypted packet (len=44)
287	191.415935	192.168.1.7	192.168.1.10	SSH	122	Server: Encrypted packet (len=68)
291	191.547945	192.168.1.10	192.168.1.7	SSH	90	Client: Encrypted packet (len=36)
292	191.549437	192.168.1.7	192.168.1.10	SSH	106	Server: Encrypted packet (len=52)
293	191.549725	192.168.1.7	192.168.1.10	SSH	106	Server: Encrypted packet (len=52)

> Frame 59: 122 bytes on wire (976 bits), 122 bytes captured (976 bits) on interface \Device\NPF\_{C3779728-C391-409D-9000-000000000000} (Src: 192.168.1.7, Dst: 192.168.1.10)

> Ethernet II, Src: RealtekPciE\_000000000000, Dst: ASRockIN\_19:34:5f (9c:6b:00:19:34:5f)

> Internet Protocol Version 4, Src: 192.168.1.7, Dst: 192.168.1.10

> Transmission Control Protocol, Src Port: 22, Dst Port: 52987, Seq: 689, Ack: 109, Len: 68

> SSH Protocol

0000 9c 6b 00 19 34 5f 5c ac 4c 2e 66 01 08 00 45 48 ..k..4...\L.f...EH

0010 00 6c 4d ba 40 00 40 06 69 28 c0 a8 01 07 c0 a8 .IM.@.@:i{.....

0020 01 0a 00 16 ce ab 5c a6 b0 e5 c6 42 3a d3 50 18 .....\\...B:P-

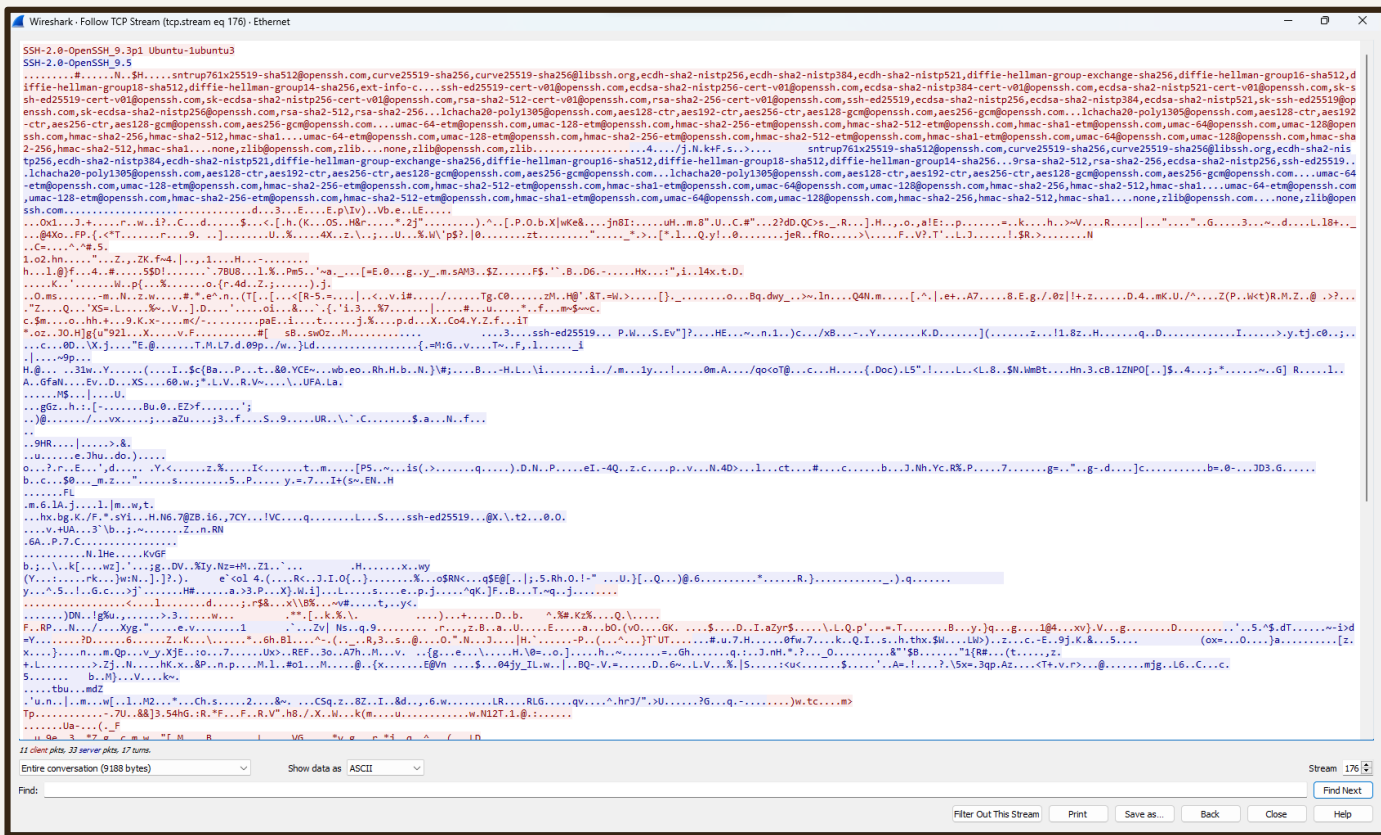
0030 00 f9 b5 0c 00 00 96 b7 46 27 66 f2 93 0a 9f c0 .....F'f.....

0040 06 39 27 c7 1c d0 b7 0b 8f 92 9d 19 bc 6e f7 57 :9'.....n-W

0050 33 d6 f3 96 59 61 61 e6 5d 07 17 a0 36 e1 58 58 3...Yaa- ]...6-XX

0060 78 f4 db 16 ff 90 37 2a 9f 98 57 93 17 4c 0a 54 x.....7\* --W--L-T

0070 23 50 0e 79 fd de 63 65 22 9c .....P.y-cc "



---

# PRATIQUE





# 06

## **Analyse des Résultats**





## **A. Montrer que les données SSH sont sécurisées :**

- En examinant les résultats de la simulation avec SSH, on peut montrer que les données échangées entre le client et le serveur sont sécurisées. Le chiffrement des données par SSH garantit que même si un attaquant intercepte les paquets, il ne peut pas comprendre le contenu, car il est chiffré.
- La comparaison visuelle des données capturées avec Wireshark dans les deux scénarios soulignera la sécurité renforcée offerte par SSH. Les informations sensibles, telles que les identifiants de connexion, les commandes, ou les données confidentielles, restent confidentielles et ne peuvent pas être exploitées par un tiers non autorisé.
- On peut également mettre en avant le mécanisme d'authentification forte de SSH, qui ajoute une couche supplémentaire de sécurité en vérifiant l'identité des parties impliquées dans la communication.

---

07

**Conclusion**



---

Dans cette présentation, nous avons exploré les fondamentaux des réseaux, mettant en évidence les modèles OSI et TCP/IP, ainsi que le rôle crucial de SSH dans la sécurisation des communications. La démonstration en temps réel a illustré l'établissement sécurisé de connexions avec SSH et a souligné les risques associés à la communication non chiffrée, démontrés via Wireshark



**MERCI POUR  
VOTRE  
ATTENTION**