

BTS SIO sur

Sécurisation des équipements via SSH

Maîtriser l'utilisation sécurisée et efficace du protocole SSH

Table des matières

Introduction.....	2
A. Contextualisation de la sécurité informatique.....	2
B. SSH : Un outil essentiel dans la trousse de la sécurité.....	2
C. Objectifs du cours.....	2
Comprendre SSH	3
A. Qu'est-ce que SSH ?	3
B. L'importance de SSH	4
Mise en œuvre et sécurisation de SSH.....	5
A. Mise en place de SSH	5
B. Sécurisation de votre connexion SSH.....	6
C. Techniques avancées de sécurisation	6
Exemples pratiques et scénarios courants.....	7
A. Connexion à un serveur distant via SSH.....	7
B. Transfert de fichiers via SSH	8
C. Tunneling SSH pour naviguer en toute sécurité sur le Web.....	8
D. Dépannage d'une connexion SSH échouée	9
E. Installation et configuration SSH sur un équipement CISCO	9
Conclusion	11

Introduction

Dans notre ère numérique, où les activités malveillantes ne cessent de croître, comprendre les mécanismes de protection des équipements informatiques est devenu indispensable. Ce cours est conçu pour vous initier aux concepts fondamentaux de la sécurisation des communications entre équipements, en utilisant SSH, une pierre angulaire de la sécurité informatique.

A. Contextualisation de la sécurité informatique

La sécurisation des équipements informatiques ne concerne pas seulement la protection contre les virus ou les logiciels malveillants. Il s'agit d'un écosystème complexe qui inclut la sécurisation des connexions, la confidentialité des informations transmises, et l'intégrité des données. Par exemple, lorsqu'un administrateur système accède à distance à un serveur pour y effectuer des mises à jour, il doit s'assurer que sa connexion est sécurisée afin d'éviter toute interception de données, ce qui pourrait conduire à des violations de sécurité.

B. SSH : Un outil essentiel dans la trousse de la sécurité

SSH apparaît comme une solution à ce besoin de sécurité. Il s'agit d'un protocole qui assure une communication sécurisée entre deux systèmes utilisant une architecture de clé publique pour authentifier la session. Plutôt que d'envoyer des informations, comme des mots de passe en clair, SSH les transmet sous une forme cryptée, rendant la tâche extrêmement difficile pour les cybercriminels qui tenteraient d'intercepter les données.

C. Objectifs du cours

Ce cours vise à vous fournir une compréhension approfondie de la manière dont SSH fonctionne et de son importance dans la sécurisation des équipements. Nous explorerons comment SSH crypte les informations, comment il authentifie une session, et comment utiliser efficacement ses fonctionnalités pour garantir une connexion sécurisée. À travers des exemples concrets et des scénarios du monde réel, vous apprendrez comment implémenter SSH dans différents contextes opérationnels.

En somme, ce module vous prépare à naviguer avec compétence dans les aspects techniques de la sécurité des équipements informatiques, en vous fournissant les connaissances et compétences nécessaires pour exploiter la puissance de SSH. Que vous cherchiez à renforcer la sécurité de votre réseau domestique ou à gérer de manière sécurisée des systèmes à grande échelle dans un environnement professionnel, les leçons tirées de ce cours seront inestimables.

Comprendre SSH

Dans cette première section, nous allons décomposer les éléments fondamentaux de SSH. Comprendre ces bases est crucial pour toute personne souhaitant maîtriser la sécurisation des équipements informatiques. Nous explorerons également pourquoi SSH est si important dans les environnements modernes, en particulier pour les systèmes et réseaux informatiques.

A. Qu'est-ce que SSH ?

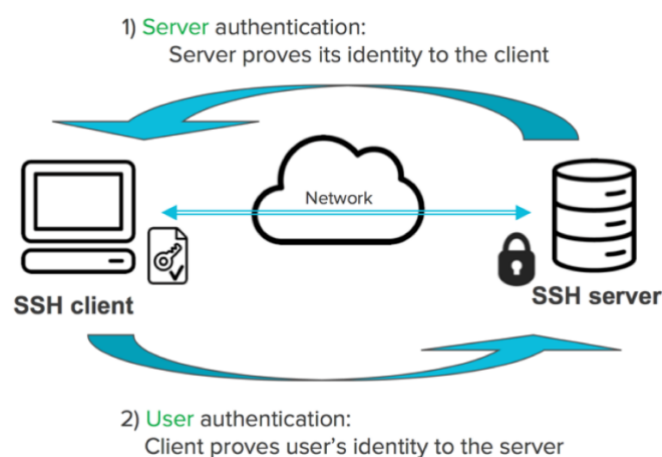
1. Définition et fonction de base

SSH, ou Secure Shell, est un protocole cryptographique utilisé pour opérer des services réseau de manière sécurisée à travers un réseau non sécurisé. Imaginons un instant que vous deviez envoyer une lettre très personnelle et confidentielle. Vous ne la glisseriez pas simplement dans une enveloppe standard et ne la confieriez pas à n'importe quel coursier. SSH, dans ce scénario, serait comme une enveloppe blindée transportée par un coursier de confiance qui possède un mot de passe secret pour vérifier qu'il livre votre message au bon destinataire et que personne d'autre ne peut lire son contenu.

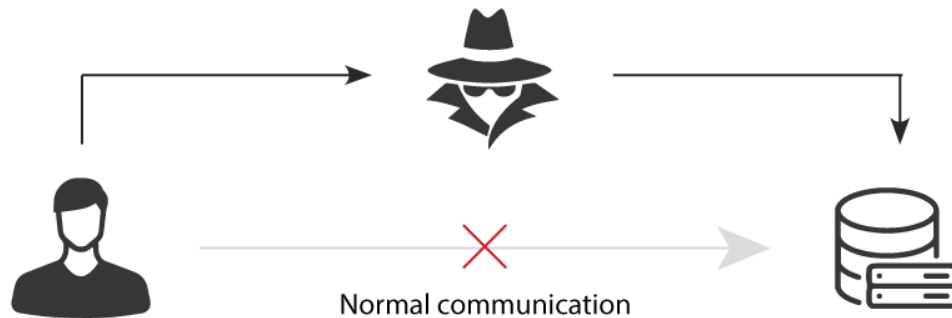
Dans la pratique, SSH permet un accès sécurisé à distance à un ordinateur ou un serveur. Les administrateurs système s'en servent pour prendre le contrôle des machines à distance, garantissant que, même si les données sont interceptées, elles ne seront pas lisibles sans la clé de chiffrement appropriée.

2. Chiffrement et authentification

SSH utilise des techniques de chiffrement pour que les données échangées entre l'utilisateur local et le serveur distant soient incompréhensibles si elles sont capturées. Il fonctionne avec une paire de clés, une "publique" et une "privée", qui sont utilisées pour crypter et décrypter les communications. C'est comme si votre maison avait une serrure unique et que vous donniez une copie de la clé à quelqu'un en qui vous avez confiance. Personne d'autre qu'avec la clé correspondante ne peut entrer.



En outre, SSH authentifie l'identité de l'hôte distant. En cas de première connexion à un système, SSH demande confirmation de l'identité du système hôte. Cela aide à prévenir les attaques de type "man-in-the-middle".



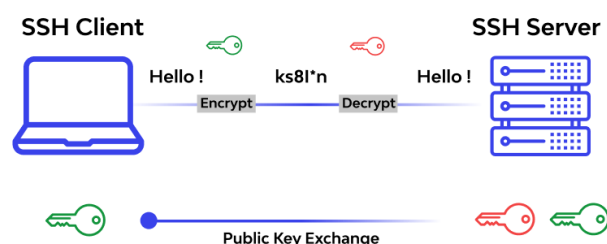
B. L'importance de SSH

1. Sécurité et confidentialité

L'une des préoccupations majeures dans la transmission de données, en particulier dans un environnement professionnel, est la sécurité des données. SSH offre une solution robuste à ce problème. Par exemple, lorsque vous vous connectez à distance à votre serveur d'entreprise pour télécharger des rapports financiers, SSH veille à ce que votre connexion soit cryptée et que les données confidentielles restent sécurisées. Sans SSH, vos mots de passe, ainsi que les informations sensibles, seraient transmis en clair, les rendant vulnérables à l'espionnage.

2. Intégrité des données

L'intégrité des données est tout aussi cruciale que la sécurité et la confidentialité, surtout lorsque des fichiers sont transférés. SSH assure non seulement que les données sont cryptées mais aussi qu'elles ne sont pas altérées en cours de route. C'est un peu comme envoyer un colis précieux : non seulement vous voulez vous assurer qu'il n'est pas ouvert, mais vous voulez aussi qu'il arrive en parfait état.



3. Gestion à distance sécurisée

SSH permet aux administrateurs de gérer des systèmes à distance avec un haut niveau de confiance et de sécurité, leur permettant d'effectuer des tâches critiques en dehors du site. Ceci est particulièrement important pour les grandes entreprises où les serveurs peuvent ne pas être physiquement accessibles.

En conclusion, SSH n'est pas simplement un outil ; c'est la réponse à un ensemble de risques de sécurité inhérents à la nature ouverte d'Internet. En comprenant son fonctionnement, vous pouvez considérablement améliorer la sécurité de vos interactions en ligne, protégeant ainsi vos systèmes, vos données, et finalement, votre entreprise ou votre intégrité personnelle des menaces potentielles.

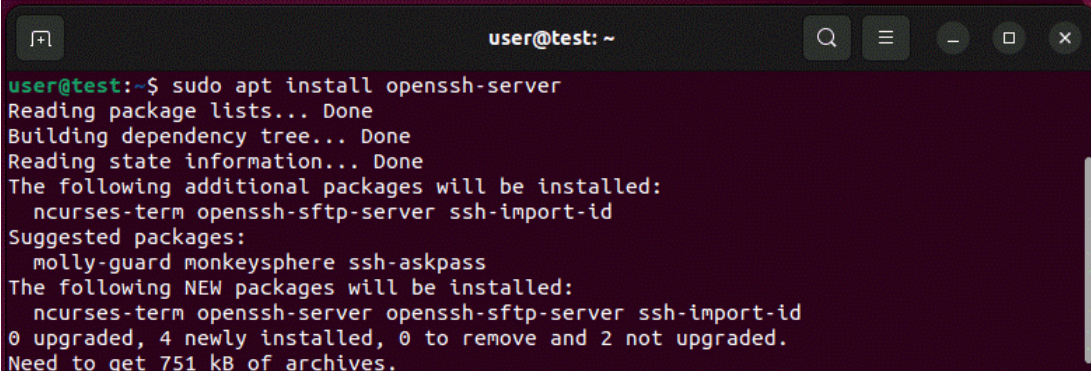
Mise en œuvre et sécurisation de SSH

Maintenant que nous avons établi une compréhension solide de SSH, nous devons examiner de plus près comment mettre en œuvre et sécuriser une connexion SSH. Utiliser SSH correctement est comme verrouiller efficacement votre porte d'entrée numérique ; cela nécessite plusieurs étapes essentielles pour garantir que seuls les utilisateurs autorisés puissent accéder à vos systèmes.

A. Mise en place de SSH

1. Installation du serveur SSH

Tout d'abord, le serveur sur lequel vous souhaitez vous connecter doit avoir un serveur SSH en cours d'exécution. Sur de nombreux systèmes Linux, le serveur SSH peut être installé via des gestionnaires de paquets comme `apt` ou `yum`. La commande est souvent aussi simple que `sudo apt-get install openssh-server`. C'est comme installer une nouvelle serrure de haute sécurité sur votre porte d'entrée.

A terminal window with a dark background and light text. The prompt is 'user@test: ~'. The command 'sudo apt install openssh-server' has been executed. The output shows the package lists being read, the dependency tree being built, and state information being read. It lists additional packages to be installed (ncurses-term, openssh-sftp-server, ssh-import-id) and suggested packages (molly-guard, monkeysphere, ssh-askpass). It then lists the new packages to be installed (ncurses-term, openssh-server, openssh-sftp-server, ssh-import-id) and shows the upgrade status (0 upgraded, 4 newly installed, 0 to remove, 2 not upgraded). Finally, it states the total size of the archives to be downloaded (751 kB).

```
user@test:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 2 not upgraded.
Need to get 751 kB of archives.
```

2. Démarrage et vérification du service SSH

Une fois installé, le serveur SSH doit être démarré. Vous pouvez généralement le faire avec une commande telle que `sudo systemctl start ssh`. Il est également crucial de s'assurer que le service démarre automatiquement au démarrage du système. C'est votre assurance que la serrure de la porte sera fonctionnelle même après que la maison ait été fermée puis rouverte.

B. Sécurisation de votre connexion SSH

1. Utilisation de clés pour l'authentification

L'utilisation de mots de passe peut être pratique, mais ils peuvent également être devinés ou interceptés. Les clés SSH offrent une solution bien plus sécurisée. En générant une paire de clés (une clé privée et une clé publique), vous pouvez configurer le serveur SSH pour n'autoriser l'accès qu'à quelqu'un possédant la clé privée correspondante. C'est comme avoir une serrure qui ne peut être ouverte qu'avec une empreinte digitale unique.

Vous créez votre paire de clés sur votre ordinateur local avec ``ssh-keygen`` et copiez la clé publique sur le serveur avec une commande comme ``ssh-copy-id``. Dès lors, vous pouvez vous connecter sans mot de passe, et personne sans la clé privée ne le peut.

2. Changer le port par défaut

Par défaut, SSH écoute sur le port 22. C'est bien connu des attaquants, qui vont souvent scanner ce port à la recherche de vulnérabilités. Changer le port d'écoute par défaut pour SSH est comme déplacer discrètement votre porte d'entrée. Les intrus pourraient même ne pas réaliser où elle est. Cela se fait en modifiant le fichier de configuration, généralement situé dans ``/etc/ssh/sshd_config``.

3. Configurer un pare-feu

Mettre en place un pare-feu ajoute une couche de sécurité supplémentaire. Il permet seulement le trafic entrant sur les ports que vous spécifiez, limitant ainsi l'accès non autorisé à votre serveur. Si vous changez le port SSH, vous devez également ajuster les règles de votre pare-feu pour refléter ce changement.

4. Mises à jour régulières

Enfin, il est vital de garder votre serveur SSH à jour. Les développeurs corrigent continuellement les failles de sécurité; ne pas mettre à jour votre système signifie que vous êtes vulnérable à des attaques connues. C'est comparable à savoir qu'une serrure a été forcée ailleurs et ne pas prendre la peine de vérifier la vôtre.

C. Techniques avancées de sécurisation

1. Jail utilisateur et environnements chroot

Mettre en place un jail pour les utilisateurs ou utiliser un environnement chroot signifie que les utilisateurs connectés via SSH sont restreints à un répertoire spécifique et ne peuvent pas parcourir

l'ensemble du système de fichiers. C'est comme donner à un invité l'accès à un certain espace dans votre maison, mais pas la capacité de se promener librement.

2. Désactivation des connexions root

Désactiver les connexions root empêche le superutilisateur de se connecter directement via SSH, obligeant les utilisateurs à se connecter d'abord en tant qu'utilisateur normal avant d'obtenir des droits d'administrateur. Cela réduit le risque d'accès non autorisé aux privilèges les plus élevés sur le système.

3. Utilisation de l'authentification à deux facteurs

Pour une sécurité accrue, l'authentification à deux facteurs (2FA) peut être mise en place pour SSH, nécessitant un second moyen d'identification en plus de la clé SSH. C'est comme avoir une alarme en plus de votre serrure.

En résumé, sécuriser votre connexion SSH est un processus en plusieurs étapes qui implique la mise en place d'un serveur SSH, la sécurisation de l'accès à ce serveur, et la mise en œuvre de pratiques avancées de sécurisation pour assurer que seuls les utilisateurs autorisés ont accès. Chaque étape contribue à créer une défense plus robuste contre les accès non autorisés et les menaces potentielles.

Exemples pratiques et scénarios courants

Après avoir établi les bases théoriques et les méthodologies pour sécuriser SSH, il est temps d'ancrer cette connaissance à travers des exemples pratiques et des scénarios que vous pourriez rencontrer dans un environnement réel. Ces illustrations servent non seulement à renforcer la compréhension mais aussi à équiper les apprenants avec des compétences pratiques nécessaires pour naviguer dans des situations réelles.

A. Connexion à un serveur distant via SSH

Imaginons que vous travaillez pour une entreprise qui héberge son site web sur un serveur distant. Pour effectuer des mises à jour ou des modifications, vous devez vous connecter à ce serveur. Voici comment vous pourriez procéder :

1. Ouvrez votre terminal ou invite de commande.
2. Tapez la commande `ssh username@remote_host`, où "username" est votre nom d'utilisateur sur le serveur distant et "remote_host" est l'adresse IP ou le domaine du serveur.
3. Si c'est votre première connexion, vous serez invité à vérifier l'empreinte digitale du serveur. Après confirmation, et si vous utilisez un mot de passe, vous devrez le saisir. Si vous avez mis en place une authentification par clé, la connexion nécessite votre clé privée.

B. Transfert de fichiers via SSH

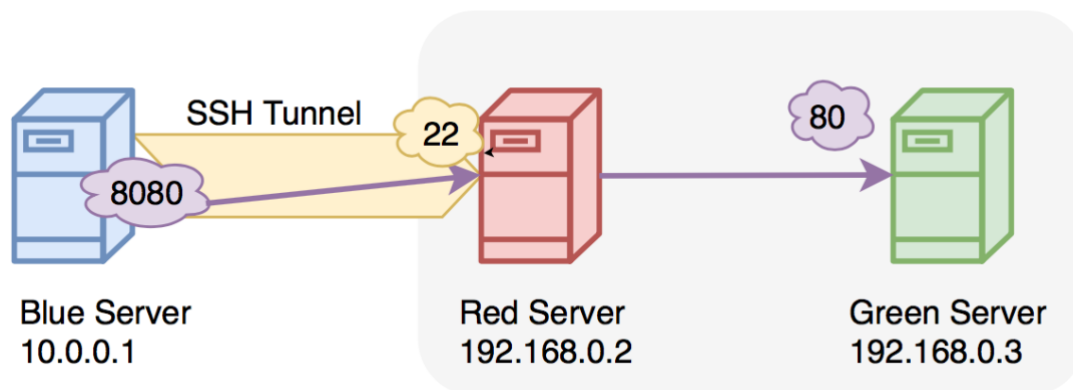
Supposons que vous deviez envoyer des fichiers de votre ordinateur vers un serveur distant. SSH offre une solution sécurisée appelée SCP (Secure Copy Protocol) ou SFTP (SSH File Transfer Protocol). Voici un scénario utilisant SCP :

1. Pour copier un fichier vers un serveur distant, ouvrez votre terminal.
2. Utilisez la commande ``scp /path/to/local/file username@remote_host:/path/to/remote/directory``. Remplacez les chemins, le nom d'utilisateur et l'hôte distant par les vôtres.
3. Entrez votre mot de passe ou utilisez l'authentification par clé si demandé, et le fichier sera transféré de manière sécurisée.

C. Tunneling SSH pour naviguer en toute sécurité sur le Web

Le tunneling SSH vous permet de naviguer sur Internet en toute sécurité, même sur un réseau non sécurisé. Imaginez que vous êtes dans un café utilisant un Wi-Fi public. Vous pouvez configurer un tunnel SSH pour acheminer votre trafic Internet, empêchant ainsi les autres sur le réseau de surveiller votre activité. Voici comment :

1. Tapez la commande ``ssh -D 8080 -C -N username@remote_host``. Cette commande crée un tunnel SSH à travers lequel vous pouvez acheminer votre trafic Internet.
2. Configurez votre navigateur web pour utiliser un proxy SOCKS local, en général c'est l'adresse ``127.0.0.1`` sur le port ``8080``.
3. Votre navigation est maintenant sécurisée, avec toutes les données envoyées et reçues cryptées via votre connexion SSH.



D. Dépannage d'une connexion SSH échouée

Un jour, vous essayez de vous connecter à votre serveur via SSH, mais la connexion échoue. Voici quelques étapes pour diagnostiquer le problème :

1. Vérifiez votre connexion Internet en accédant à un site web ou en utilisant la commande `ping`.
2. Essayez de tracer la route vers le serveur distant avec `traceroute remote_host` pour voir s'il y a des problèmes de réseau.
3. Utilisez la commande `ssh -vvv username@remote_host` pour une connexion SSH avec un débogage détaillé. Cette commande vous donnera des informations détaillées sur ce qui pourrait causer le problème.
4. Vérifiez le pare-feu et les règles de sécurité du réseau pour vous assurer que le port utilisé par SSH (généralement 22) est ouvert et accessible.

E. Installation et configuration SSH sur un équipement CISCO

L'installation et la configuration de SSH sur un équipement réseau CISCO, notamment un commutateur ou un routeur, sont des tâches cruciales pour la sécurisation des accès administratifs. Les équipements réseau CISCO fonctionnent généralement avec leur propre système d'exploitation, appelé IOS (Internetwork Operating System), et nécessitent des étapes spécifiques pour l'installation de SSH. Voici comment vous pouvez procéder :

1. Préparation de l'équipement :

Avant tout, assurez-vous que votre version de l'IOS Cisco supporte le protocole SSH. Ceci est essentiel car toutes les versions ne sont pas compatibles avec cette fonctionnalité. Vous devrez peut-être effectuer une mise à niveau de l'IOS pour prendre en charge SSH.

2. Configuration de l'adresse IP :

Le routeur ou le commutateur doit être configuré avec une adresse IP valide et des informations de routage correctes, si nécessaire. Ceci permet à l'appareil d'être accessible sur le réseau.

```
interface vlan1
ip address 192.168.1.1 255.255.255.0
no shutdown
exit
```

3. Configuration des noms de domaine :

Le système doit avoir un nom de domaine configuré. Utilisez la commande suivante pour attribuer un nom de domaine :

```
ip domain-name mondomaine.com
```

4. Génération des clés de cryptographie :

SSH repose sur la cryptographie à clé publique pour authentifier les dispositifs distants. Par conséquent, vous devrez générer une paire de clés sur votre appareil Cisco. Entrez la commande suivante :

```
crypto key generate rsa
```

Lorsque vous y êtes invité, spécifiez la taille de la clé. Une clé de 1024 bits est généralement considérée comme suffisante, mais 2048 bits offrent une sécurité supérieure (mais à un coût de performance).

5. Activation de SSH :

Après avoir généré les clés, activez le serveur SSH sur l'appareil en utilisant la commande suivante :

```
ip ssh version 2
```

Il est recommandé d'utiliser SSH version 2, car elle offre des améliorations de sécurité par rapport à la version précédente.

6. Configuration des lignes vty :

Vous devez configurer les lignes vty pour accepter les connexions SSH. Par défaut, elles pourraient accepter des connexions Telnet non sécurisées. La configuration suivante limite l'accès aux connexions SSH :

```
line vty 0 4
transport input ssh
login local
password motdepasse
exit
```

7. Création d'un compte utilisateur :

Enfin, créez un compte utilisateur qui sera utilisé pour les connexions SSH. Configurez le nom d'utilisateur et le mot de passe en utilisant la commande suivante :

```
username admin secret motdepasse
```

8. Vérification de la configuration :

Après avoir appliqué les configurations, vérifiez votre configuration SSH en utilisant la commande :

```
show ip ssh
```

Cette commande devrait vous montrer la version de SSH activée ainsi que l'état de la clé RSA.

9. Test de la connexion SSH :

Maintenant, depuis un PC ou un terminal SSH, essayez de vous connecter à l'appareil avec la commande :

```
ssh -l admin 192.168.1.1
```

Remplacez "admin" par le nom d'utilisateur que vous avez créé, et "192.168.1.1" par l'adresse IP de votre équipement.

Cela devrait vous amener à une invite de connexion sécurisée, démontrant que SSH est bien configuré et fonctionnel. Gardez à l'esprit que les étapes spécifiques peuvent varier légèrement en fonction de la version de l'IOS que vous utilisez, et il est toujours recommandé de consulter la documentation de Cisco pour les directives spécifiques à votre matériel et à votre version de logiciel.

Ces scénarios sont des illustrations de la manière dont SSH est utilisé dans la pratique, donnant vie aux principes et méthodes que nous avons discutés précédemment. Chaque exemple sert à montrer comment SSH peut être utilisé de manière sécurisée pour communiquer et transférer des données, soulignant l'importance de suivre les meilleures pratiques de sécurité dans toutes ces opérations.

Conclusion

Alors que nous concluons ce voyage à travers l'exploration de SSH, il est crucial de récapituler et de souligner les aspects essentiels et les compétences que ce module visait à inculquer. Le monde numérique dans lequel nous vivons aujourd'hui n'est pas sans menaces, et la sécurité des données est devenue une préoccupation centrale pour les individus et les entreprises. À cet égard, comprendre et être capable de mettre en œuvre des protocoles de sécurité tels que SSH est une compétence inestimable pour tout technicien d'assistance en informatique.

A. Réaffirmation de l'importance de SSH : Nous avons découvert comment SSH, en tant que protocole sécurisé, offre une avenue pour l'authentification sécurisée, la confidentialité, et l'intégrité des données. En utilisant le cryptage pour sécuriser les canaux de communication sur des réseaux non

sécurisés, SSH permet aux utilisateurs et aux administrateurs systèmes de gérer les serveurs à distance et d'effectuer d'autres tâches de communication entre différents systèmes.

B. Application pratique et compétences acquises : À travers des exemples concrets et des scénarios courants, nous avons vu comment les connaissances théoriques sur SSH sont appliquées dans la pratique. Ces compétences vont de la simple connexion à des serveurs distants, au transfert sécurisé de fichiers, et même à la configuration de tunnels pour sécuriser la navigation web. Chaque compétence acquise ici forme un élément vital dans l'arsenal d'un technicien informatique, vous préparant à faire face à diverses situations dans votre parcours professionnel.

C. Vue d'ensemble de la sécurité : Enfin, il est essentiel de reconnaître que la sécurisation de SSH est une partie d'un écosystème de sécurité plus vaste. Un technicien d'assistance doit être conscient des menaces en constante évolution et être prêt à continuer à apprendre et à s'adapter. Utiliser SSH avec les meilleures pratiques de sécurité n'est qu'un élément de la sécurisation des infrastructures informatiques. La vigilance, la mise à jour continue des systèmes et la connaissance des dernières cybermenaces sont tout aussi importantes.

En clôture, ce cours a été conçu pour vous fournir non seulement des connaissances théoriques mais aussi des compétences pratiques en matière de sécurisation des communications via SSH. Tout en marchant avec ces outils, souvenez-vous que la sécurité est un voyage, pas une destination. C'est une pratique qui nécessite une vigilance constante et un apprentissage continu. Alors que vous avancez dans votre carrière, que ce soit en tant que technicien d'assistance en informatique ou dans d'autres rôles informatiques, garder une longueur d'avance sur les pratiques de sécurité et les technologies émergentes vous servira bien.