

BTS SIO

Situation professionnelle numéro 1

Déploiement d'un service SSH : Sécurité et maintenance"

Description :

SSH (Secure Shell) est un protocole essentiel pour sécuriser les accès distants aux systèmes d'information. Il permet des communications cryptées, une gestion centralisée des serveurs, et protège les données échangées tout en simplifiant l'administration.

Mots-clés :



Nom	Date	Tampon
Diallo Kadiatou Laetitia-Marie	03/03/2025	

Plan de la situation

Le cahier des charges.....	3
L'expression des besoins.....	3
La description de l'existant.....	3
Les offres du marché'.....	3
Les risques pour un service SSH.....	4
Attaques par force brute.....	4
Cle's SSH compromises.....	4
Acce`s non autorise' via des comptes compromis.....	5
Manque de mise à jour du serveur SSH.....	5
Attaques de type "Man-in-the-Middle" (MITM).....	5
Erreurs de configuration.....	5
Utilisation d'un protocole obsole`te.....	5
Re'vocation de cle's SSH.....	5
Mise en œuvre.....	6
1. Création du réseau.....	6
2. Configuration du périphérique Cisco (Routeur) pour SSH.....	8
3. Configurer le client (PC) pour SSH.....	10

Le cahier des charges

L'expression des besoins

La société es2com recherche une solution SSH fiable et sécurisée pour protéger les accès distants à ses serveurs. Elle souhaite renforcer la sécurité des connexions administratives, tout en assurant une gestion simplifiée et un contrôle strict des utilisateurs autorisés. Un de ses clients a exprimé le besoin de sécuriser ses échanges via un tunnel SSH pour ses opérations sensibles.

La description de l'existant

Actuellement, la société es2com n'a pas de solution d'administration à distance pour ses serveurs. Toute intervention nécessite un déplacement physique, même pour des tâches simples, ce qui entraîne des pertes de temps et des coûts élevés. Les serveurs, parfois hébergés dans d'autres villes ou pays, sont accessibles uniquement sur site. Aucun système sécurisé, comme SSH, n'a été déployé pour permettre une gestion à distance fiable et efficace. Cette situation limite la réactivité et l'agilité de l'entreprise face aux besoins de maintenance ou de dépannage.

Les offres du marché :

Pour répondre aux besoins de notre client plusieurs solutions sont disponibles :

Voici un tableau comparatif des offres du marché pour la gestion SSH :

Solution	Type	Avantage	Inconvénients	Adapté pour
OpenSSH	Open source, Gratuit	<ul style="list-style-type: none">- Large compatibilité (Linux, Mac, Windows avec adaptations).- Authentification par clé publique.- Robuste et flexible.	<ul style="list-style-type: none">- Interface en ligne de commande uniquement.- Nécessite des compétences techniques.	Environnements professionnels variés.
PuTTY	Gratuit	<ul style="list-style-type: none">- Léger et portable.- Compatible Windows.- Gestion des	<ul style="list-style-type: none">- Pas adapté à des configurations avancées.	Utilisateurs occasionnels.

		connexions basiques (SSH, SCP).	- Interface minimale.	
MobaXterm	Gratuit (version Pro payante)	<ul style="list-style-type: none"> - Interface intuitive. - Support multi-protocole (SSH, RDP, FTP, SCP). - Fonctionnalités avancées en version Pro. 	<ul style="list-style-type: none"> - Fonctionnalités limitées dans la version gratuite. - Peut être gourmand en ressources. 	Administrateurs recherchant un outil complet.
SolarWinds / Bitvise	Commercial, Payant	<ul style="list-style-type: none"> - Interface utilisateur conviviale. - Monitoring avancé. - Support technique dédié. 	<ul style="list-style-type: none"> - Coût élevé. - Complexité pour des besoins simples. 	Entreprises avec des infrastructures complexes.

OpenSSH reste la solution la plus adaptée pour une entreprise comme es2com grâce à sa fiabilité, son intégration native sous Linux et sa flexibilité pour sécuriser les connexions distantes. MobaXterm peut être une bonne alternative si une interface graphique est privilégiée.

Les risques pour un service SSH

Bien que **SSH** soit un protocole sécurisé, il comporte certains risques qu'il est important de prendre en compte pour assurer une gestion sécurisée des accès distants. Voici les principaux risques associés à l'utilisation de SSH :

Attaques par force brute

Les attaquants tentent de deviner des mots de passe pour accéder aux serveurs via SSH. Si des mots de passe faibles sont utilisés ou si les comptes ne sont pas protégés par des mécanismes de sécurité robustes, il devient facile pour un attaquant de compromettre le système.

Mesure préventive : Utiliser l'authentification par clé publique et désactiver l'accès par mot de passe.

Cle's SSH compromises

Si une clé privée est exposée ou volée, l'attaquant peut se connecter au serveur sans avoir besoin de mot de passe. Si les clés ne sont pas gérées correctement, elles peuvent devenir un vecteur d'attaque majeur.

Mesure préventive : Protéger les clés privées avec un mot de passe fort, et utiliser des mécanismes de gestion de clés.

Acces non autorise via des comptes compromis

Des comptes d'utilisateurs avec des privilèges élevés peuvent être exploités si un attaquant parvient à les compromettre. Cela est particulièrement risqué si l'accès root est autorisé via SSH.

Mesure préventive : Désactiver l'accès root et limiter l'accès aux utilisateurs autorisés uniquement.

Manque de mise à jour du serveur SSH

Si le logiciel SSH n'est pas régulièrement mis à jour, il peut contenir des vulnérabilités connues qui peuvent être exploitées par des attaquants.

Mesure préventive : Mettre à jour régulièrement le serveur et le logiciel SSH pour corriger les failles de sécurité.

Attaques de type "Man-in-the-Middle" (MITM)

Lorsqu'un attaquant intercepte les communications entre le client et le serveur SSH, il peut espionner ou manipuler les données échangées. Bien que SSH soit conçu pour être sécurisé, des configurations incorrectes (comme l'utilisation de clés publiques non vérifiées) peuvent rendre l'attaque possible.

Mesure préventive : Vérifier les empreintes des clés publiques et utiliser des certificats valides.

Erreurs de configuration

Une mauvaise configuration de SSH, comme l'utilisation de ports par défaut ou des permissions excessives sur les fichiers de clé, peut exposer le système à des attaques.

Mesure préventive : Revoir régulièrement les configurations de sécurité, modifier les ports par défaut et restreindre les permissions.

Utilisation d'un protocole obsolète

Les anciennes versions de SSH (comme SSH-1) présentent des vulnérabilités connues et doivent être évitées.

Mesure préventive : Forcer l'utilisation de SSH-2, qui est plus sécurisé, et désactiver les anciennes versions.

Re'vocation de clé's SSH

Si un utilisateur quitte l'entreprise ou que la clé privée d'un utilisateur est compromise, il est essentiel de révoquer l'accès immédiatement. Sinon, l'attaquant pourrait continuer à accéder aux ressources protégées par SSH.

Mesure préventive : Mettre en place une gestion stricte des clés SSH et leur révocation rapide en cas de besoin.

Pour minimiser ces risques, il est crucial de suivre les bonnes pratiques de sécurité : utiliser des clés SSH, désactiver l'accès root, maintenir le logiciel à jour, configurer des pare-feu pour limiter l'accès SSH, et surveiller les logs régulièrement. En adoptant ces mesures, SSH reste un moyen sûr et efficace pour gérer les accès distants.

Packet Tracer permet de configurer des équipements Cisco, mais il ne prend pas directement en charge tous les aspects d'un système Linux comme dans une vraie configuration SSH. Cependant il

simule l'administration à distance via SSH en utilisant un routeur ou un switch Cisco. Bien que limité par la plateforme, cela permet d'expérimenter et de comprendre les concepts de base d'une connexion SSH dans un réseau Cisco simulé.

1. Création du réseau

Pour simuler SSH dans Packet Tracer, vous n'avez besoin que d'un **PC** (pour le client SSH) et d'un **routeur** compatible ainsi que de câbles réseau pour connecter ces appareils. Si nécessaire, vous pouvez ajouter un **switch** pour étendre votre réseau ou simuler des scénarios plus complexes.

Voici un tableau comparatif des **routeurs Cisco** les plus appropriés pour héberger le service **SSH** dans **Packet Tracer** :

Routeur Cisco	Support SSH	Avantages	Inconvénients	Adapté pour
Cisco 2911	Oui	<ul style="list-style-type: none">- Supporte SSH.- Plusieurs interfaces disponibles.- Bon rapport performance/simplicité.	<ul style="list-style-type: none">- Peut être plus complexe à configurer pour les débutants.	Idéal pour une simulation complète de réseau.
Cisco 1941		<ul style="list-style-type: none">- Supporte SSH.- Modèle plus simple et accessible.- Moins coûteux en termes de ressources.	<ul style="list-style-type: none">- Moins puissant que le 2911.	Parfait pour des scénarios de réseau simples à moyens.

Cisco 2811	Oui	<ul style="list-style-type: none"> - Compatible avec SSH. - Plus ancien, mais robuste pour des scénarios simples. 	<ul style="list-style-type: none"> - Moins performant que les modèles plus récents. 	Utilisation basique pour une petite simulation.
Cisco 4331	Oui	<ul style="list-style-type: none"> - Modèle plus récent. - Plus puissant et performant pour des réseaux complexes. - Supporte SSH. 	<ul style="list-style-type: none"> Peut être excessif pour des simulations simples. 	Pour des réseaux plus avancés ou des simulations à grande échelle.

Pour une simulation de **SSH** dans **Packet Tracer**, le routeur le plus approprié pour héberger le service SSH et permettre un accès distant sécurisé serait le **Cisco 2911** ou **Cisco 1941**. Ces modèles sont des routeurs de gamme intermédiaire qui supportent les fonctionnalités de base nécessaires à la configuration de SSH dans un réseau simulé.



Switch

Routeur 2911

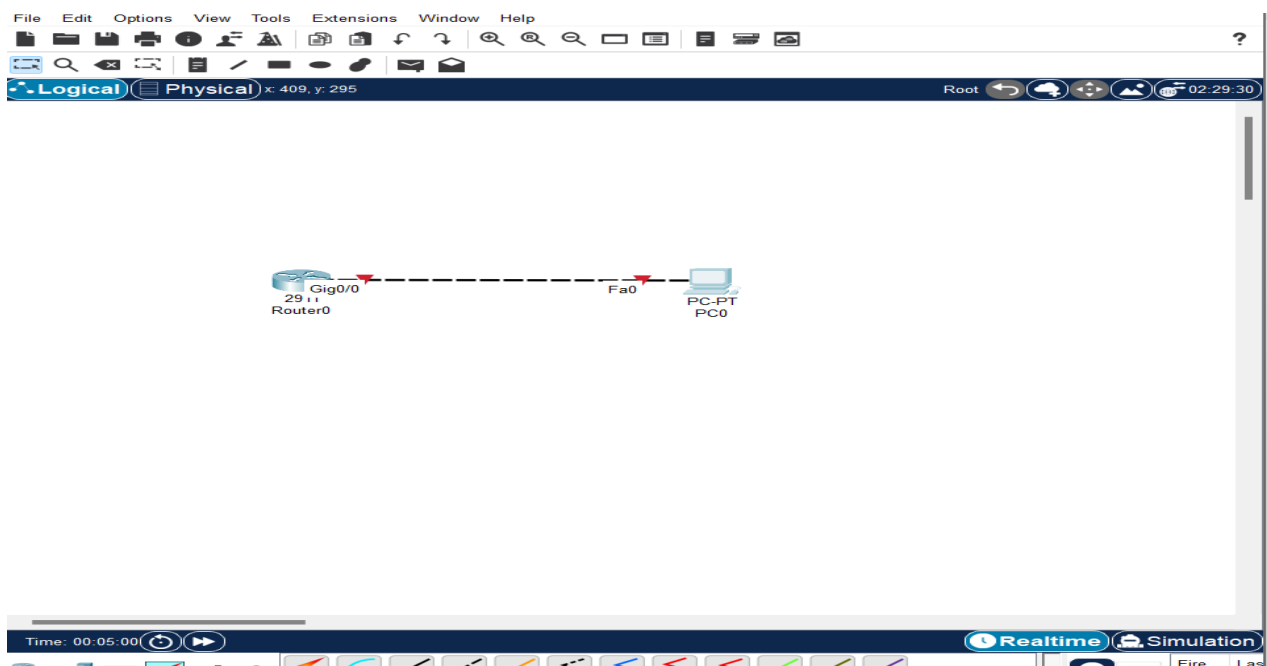


PC

2. Configuration du périphérique Cisco (Routeur) pour SSH :

La configuration se fait en deux temps :

Tout d'abord, lancez **Packet Tracer**, puis créez une topologie réseau comme indiqué dans l'image ci-dessous. Ajoutez le routeur à l'espace de travail.



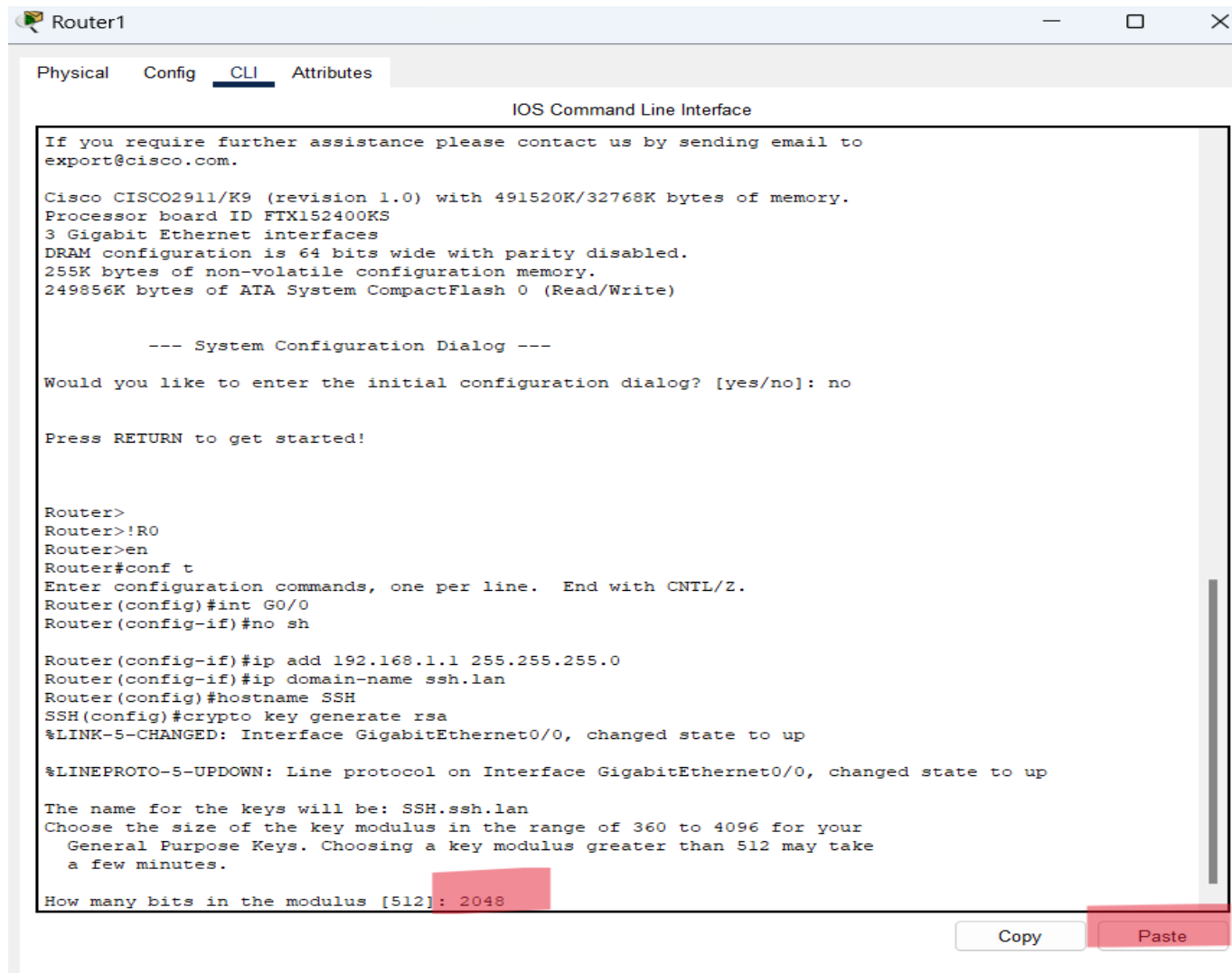
Sur le **routeur** on configure SSH pour l'administration à distance. Cette configuration se fera en deux temps :

On entre les lignes de commandes suivantes dans le CLI :

!R0

en


```
conf t
int G0/0
no sh
ip add 192.168.1.1 255.255.255.0
ip domain-name ssh.lan
hostname SSH
crypto key generate rsa
```



```
Router1
Physical Config CLI Attributes
IOS Command Line Interface
If you require further assistance please contact us by sending email to
export@cisco.com.
Cisco CISCO2911/K9 (revision 1.0) with 491520K/32768K bytes of memory.
Processor board ID FTX152400KS
3 Gigabit Ethernet interfaces
DRAM configuration is 64 bits wide with parity disabled.
255K bytes of non-volatile configuration memory.
249856K bytes of ATA System CompactFlash 0 (Read/Write)

--- System Configuration Dialog ---
Would you like to enter the initial configuration dialog? [yes/no]: no

Press RETURN to get started!

Router>
Router>!R0
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int G0/0
Router(config-if)#no sh

Router(config-if)#ip add 192.168.1.1 255.255.255.0
Router(config-if)#ip domain-name ssh.lan
Router(config)#hostname SSH
SSH(config)#crypto key generate rsa
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

The name for the keys will be: SSH.ssh.lan
Choose the size of the key modulus in the range of 360 to 4096 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.
How many bits in the modulus [512]: 2048
```

Il affiche le message ci-dessus : **how many bits in the modulus [512] : 2048**

Le nombre **2048**, dans le contexte de la **taille du module (modulus)**, représente la **taille de la clé en bits** utilisée dans le chiffrement RSA .Elle est utilisé pour sécuriser les communications. Plus la clé est grande, plus la sécurité est renforcée, mais cela demande aussi plus de ressources informatiques pour le chiffrement et le déchiffrement. Une clé de **2048 bits** offre un bon compromis entre sécurité et performance, ce qui explique son usage répandu.

On peut entrer des lors le deuxieme fragment de ligne de commande dans le CLI :

```
line vty 0 4
```

transport input ssh

login local

username tux privilege 15 password linux

enable secret linux

do write memory

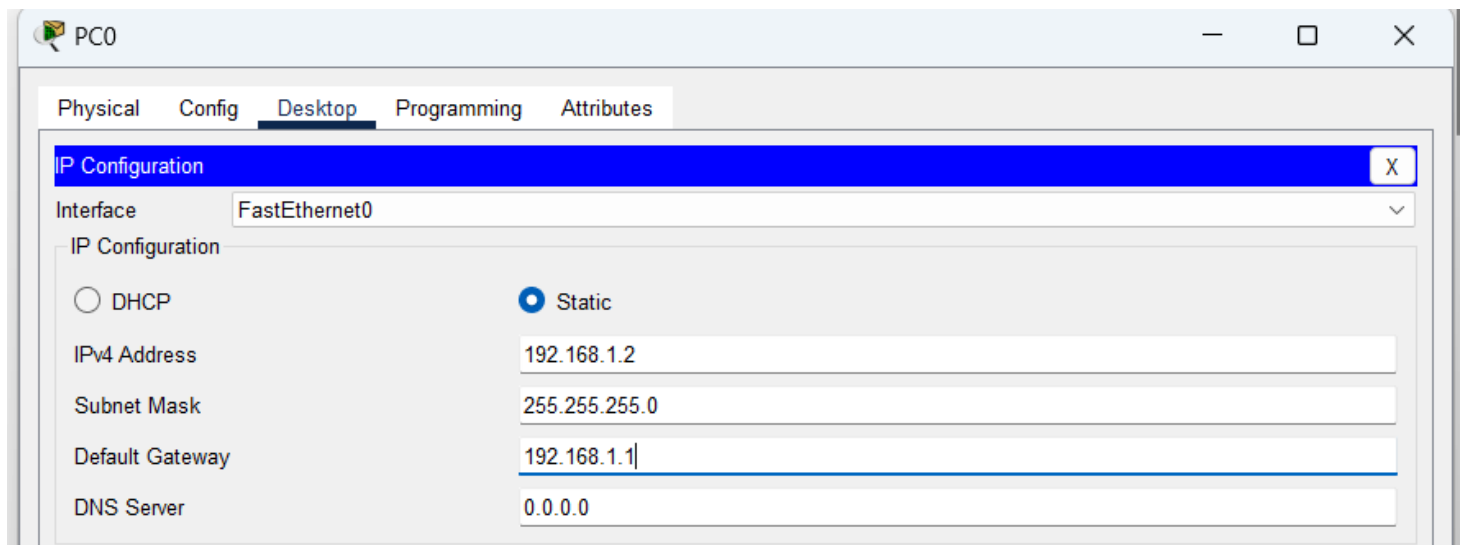
```
SSH(config)#
*Mar 1 0:3:26.500: %SSH-5-ENABLED: SSH 1.99 has been enabled
SSH(config)#line vty 0 4
SSH(config-line)#transport input ssh
SSH(config-line)#login local
SSH(config-line)#username tux privilege 15 password linux
SSH(config)#enable secret linux
SSH(config)#do write memory
Building configuration...
[OK]
SSH(config)#
```

Copy

Paste

3. Configurer le client (PC) pour SSH

Dans Packet Tracer ,on configure le PC client en lui affectant une adresse ip qui se trouve dans le même réseau que le routeur.

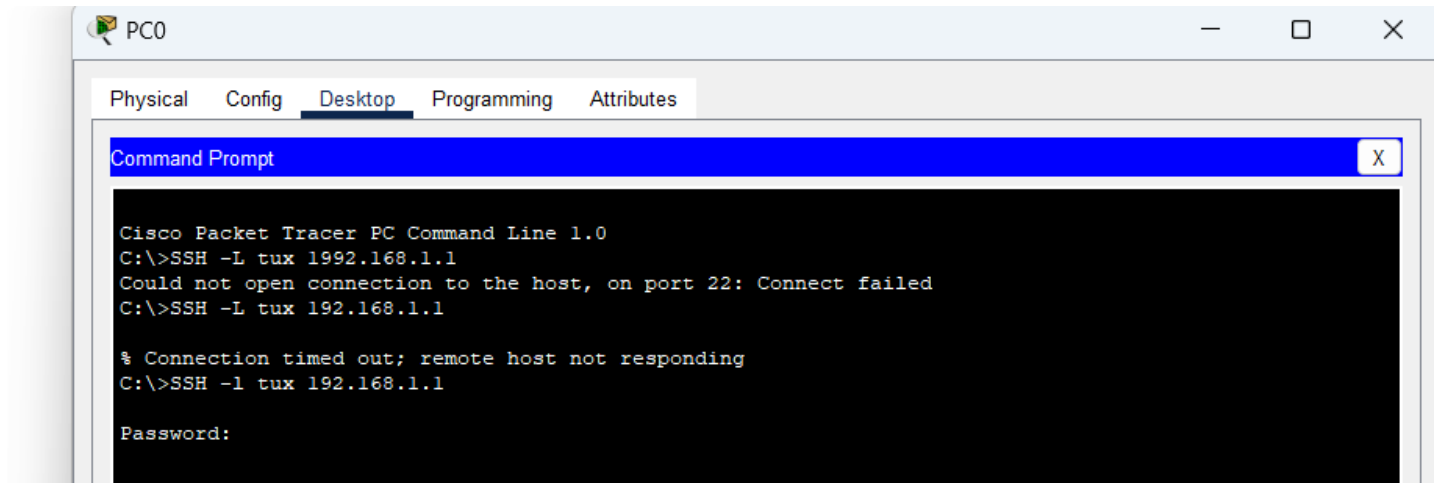


The screenshot shows the configuration window for PC0 in Packet Tracer. The 'Desktop' tab is selected, and the 'IP Configuration' section is expanded. The 'Interface' is set to 'FastEthernet0'. Under 'IP Configuration', the 'Static' radio button is selected. The fields are filled with the following values:

Field	Value
IPv4 Address	192.168.1.2
Subnet Mask	255.255.255.0
Default Gateway	192.168.1.1
DNS Server	0.0.0.0

Nous utilisons l'outil "**Command Prompt**" sur le PC pour tester la connexion SSH . on entre la commande :**SSH -l tux 192.168.1.1**. Cette commande est utilisée pour **établir une connexion SSH** vers un appareil distant (comme un routeur ou un serveur). Si un administrateur veut configurer un routeur situé dans un autre bâtiment ou une autre ville, il peut se connecter en SSH depuis son PC avec cette commande. Une fois connecté, il peut exécuter des commandes comme s'il était sur place. C'est une commande essentielle pour la gestion des infrastructures réseau à distance.

Au premier essai j'obtiens un message d'erreur: "**Could not open connection to the host, on port 22: Connect failed**"



The screenshot shows a Cisco Packet Tracer PC Command Line window for a PC named PC0. The window has tabs for Physical, Config, Desktop, Programming, and Attributes. The Desktop tab is active, displaying a Command Prompt window. The Command Prompt shows the following text:

```
Cisco Packet Tracer PC Command Line 1.0
C:\>SSH -L tux 192.168.1.1
Could not open connection to the host, on port 22: Connect failed
C:\>SSH -L tux 192.168.1.1

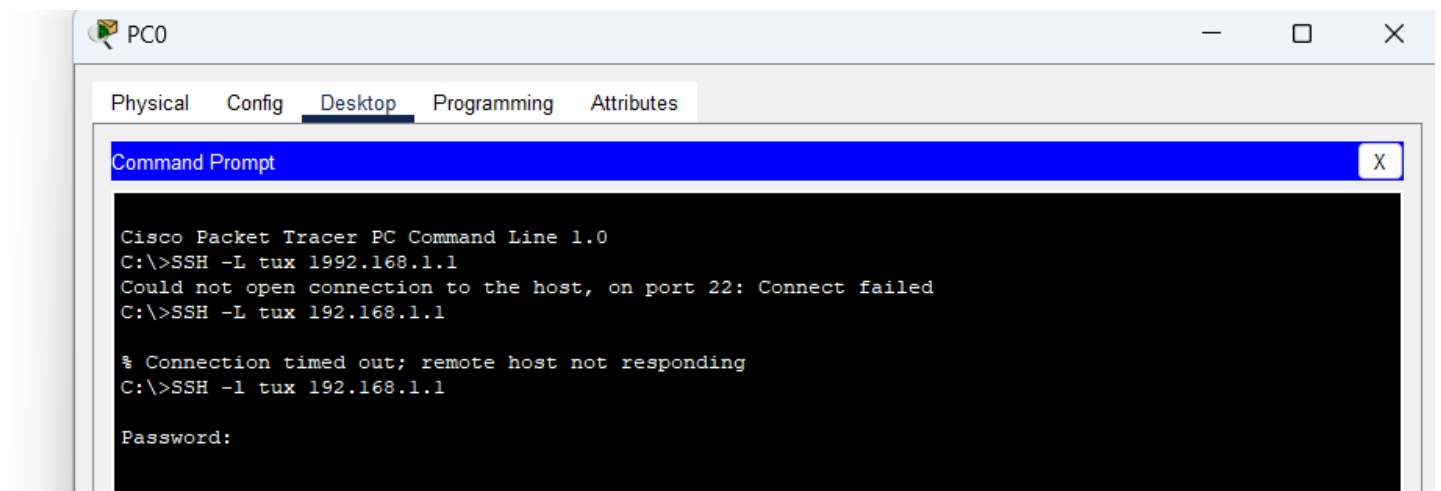
% Connection timed out; remote host not responding
C:\>SSH -l tux 192.168.1.1

Password:
```

Cela signifie que le **PC client** n'a pas pu établir de connexion SSH avec l'hôte (le routeur ou serveur cible) sur le port **22**. Que faire dans ce cas ?

- Tout d'abord on vérifie que le service SSH est correctement activé et configuré sur le routeur cible
- On vérifie que l'adresse IP du PC est bien dans le même sous-réseau que le routeur.
- On teste la connectivité réseau avec **ping**
- On vérifie que le PC et le routeur sont correctement connectés avec un **câble Ethernet** ou via un switch.
- On relance la commande SSH après correction.

Après réédification des toutes ces étapes , j'obtiens le message ci-dessous :



Cela signifie que le **service SSH** est actif et vous demande d'entrer le mot de passe pour l'utilisateur **tux**.

L'intégration du protocole **SSH** dans le réseau de l'entreprise représente une solution efficace pour répondre à son besoin d'administration à distance. Grâce à sa capacité à établir des connexions sécurisées et chiffrées, SSH permet aux administrateurs de gérer les serveurs sans avoir à se déplacer, même lorsqu'ils sont situés dans d'autres villes ou pays.

En adoptant SSH, l'entreprise pourra non seulement réduire les coûts et les délais liés aux interventions physiques, mais également améliorer la sécurité et la flexibilité de ses opérations. Cette mise en œuvre marque une étape clé dans la modernisation de ses infrastructures et le renforcement de ses capacités techniques.