

Accès à distance avec Telnet et SSH

Visualiser les services actifs

Lorsqu'un **service est actif** sur un serveur, un **port TCP ou UDP est en écoute** :

- le service Telnet utilise par défaut le port 23 ;
- le service SSH utilise par défaut le port 22.

Utilisez la commande suivante pour visualiser les ports UDP et TCP actif :

```
$ netstat -nltu  
$ ss -nltu
```

Si les ports 23 et 22 se sont pas en écoute → ces deux services ne sont pas installés. Lien : <https://artheodoc.wordpress.com/2016/06/19/voir-les-ports-ouverts-sous-linux/>

Installer le service Telnet sur la VM Linux

- Lancer votre VM Linux et depuis un terminal, utilisez la commande suivante :

```
$ sudo apt -y install telnetd
```

- Visualisez le **statut** du service et le **port 23 en écoute** :

```
$ sudo systemctl status inetd  
$ ss -nlt
```

Tapez Q pour quitter

- Prenez connaissance de l'adresse IP de votre VM

```
$ ip a
```

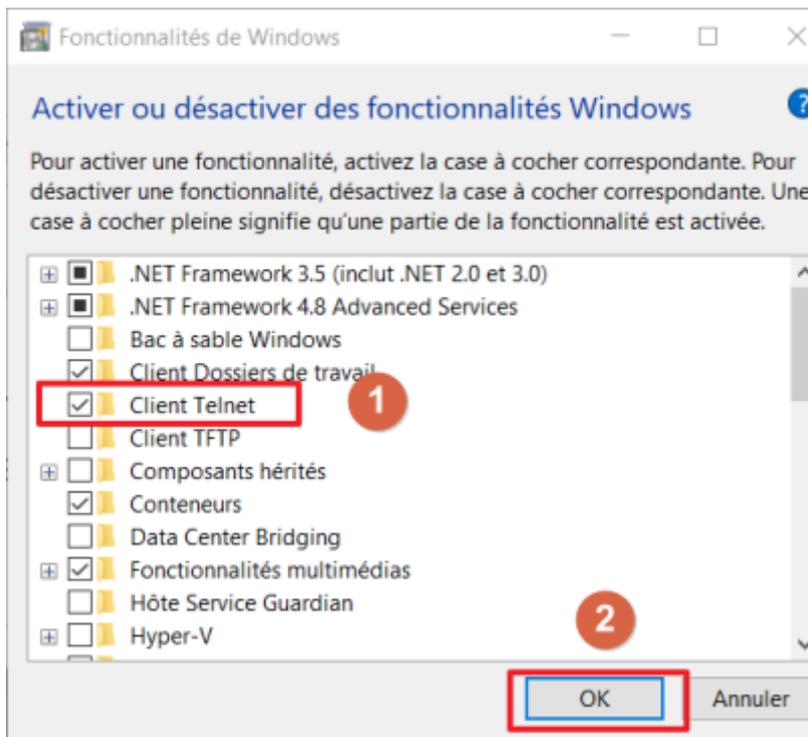
Configuration de l'accès Telnet en ligne de commande

Il est nécessaire de disposer d'un client Telnet.

Un client Telnet est disponible sous Windows.

Pour installer le client Telnet sous Windows :

- accédez à l'outil ou **Activer désactiver des fonctionnalités Windows**
- puis cocher le **Client Telnet** et validez en cliquant sur le bouton **OK**



* Lancez une **invite de commandes** et saisissez le nom du client telnet suivi de l'adresse du serveur distant :

```
C:>telnet 199.199.199.199
```

- saisissez vos **identifiants** (login et mot de passe)

Configuration de l'accès Telnet au serveur avec Putty

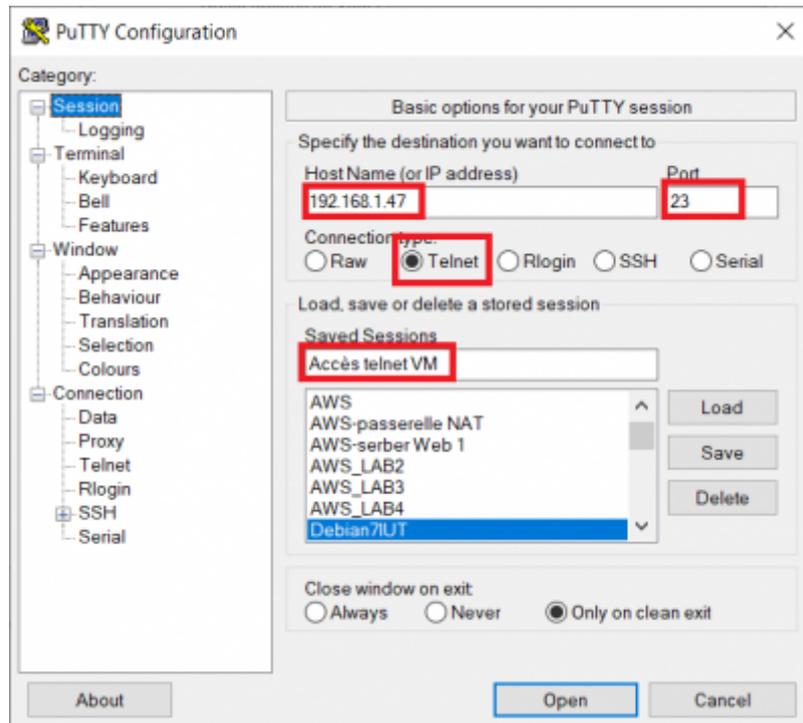
Putty est un utilitaire qui permet d'ouvrir une session **Telnet** ou **SSH** sur un serveur distant, c'est-à-dire un session de terminal. Ce logiciel correspond à un seul fichier **putty.exe**.

- Télécharger l'utilitaire **putty.exe** à partir du partage **Classe**.

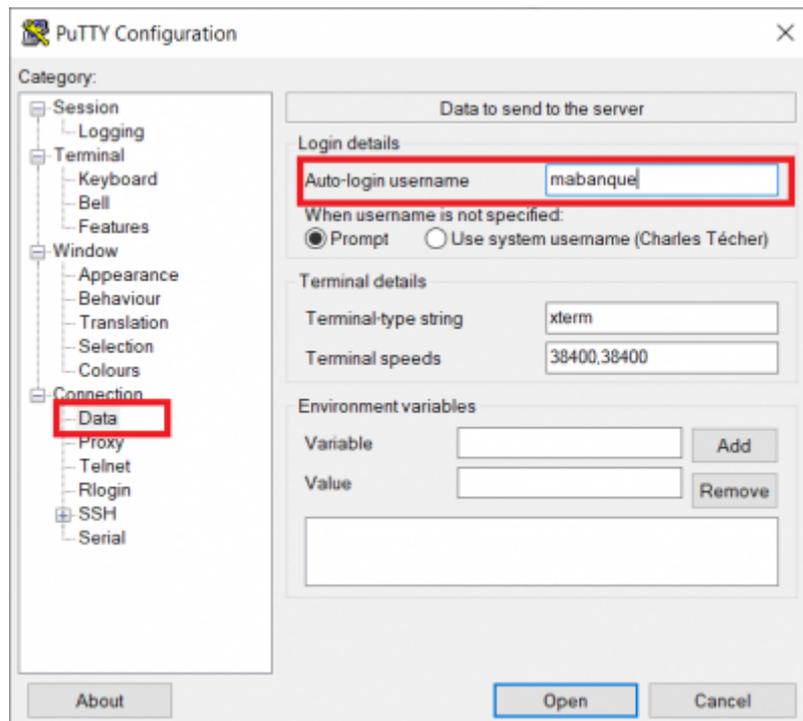


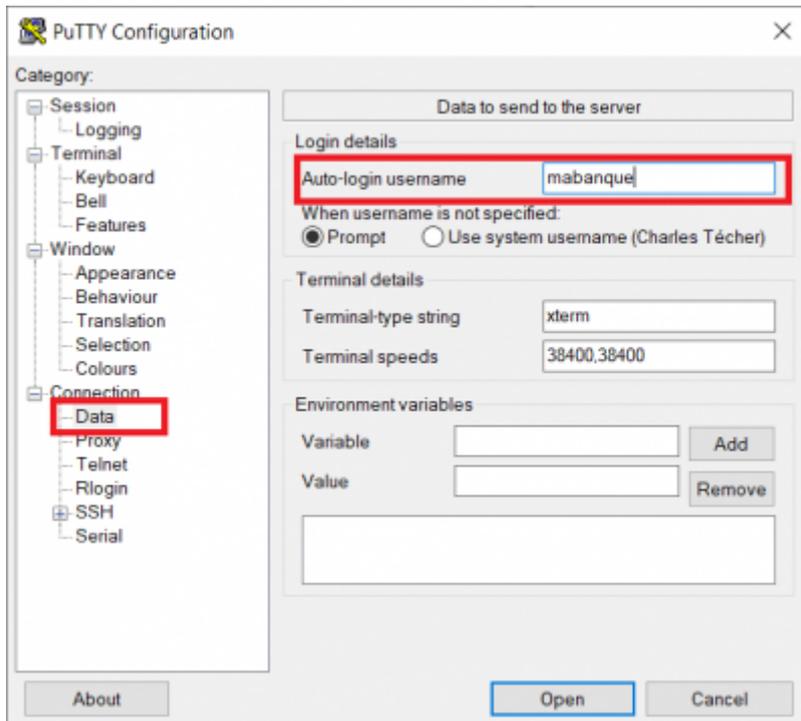
Putty est téléchargeable sur Internet à l'adresse du site <http://www.putty.org/>.

- Configurez Putty pour avoir un accès **console** à votre serveur.
- Indiquez l'**adresse IP** de votre serveur et le **port 23 (Telnet)**.

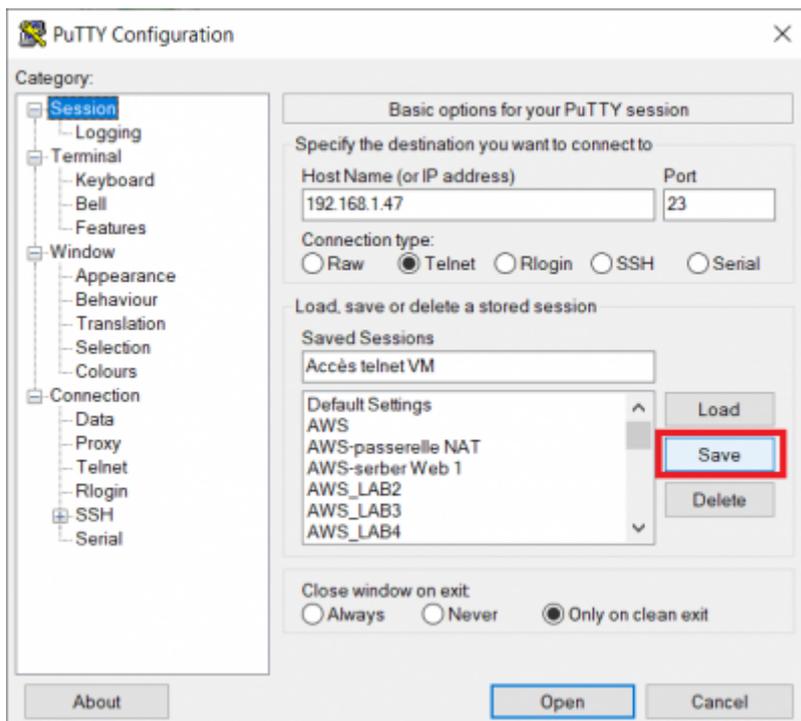


- précisez le compte de l'utilisateur existant dans votre système Linux **mabanque** pour vous connecter :

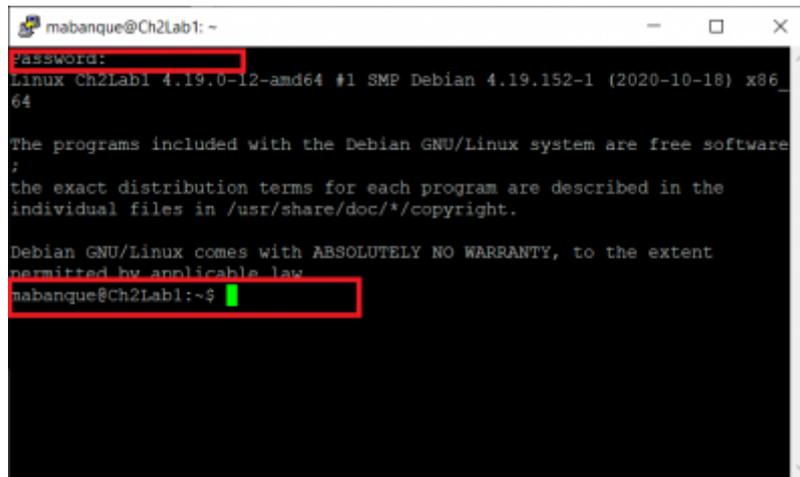




- Revenez sur la rubrique Session et sauvegardez les paramètres de votre connexion



- Puis cliquez sur **Open** pour lancer la connexion.

A terminal window titled 'mabanque@Ch2Lab1: ~' with a dark background. The prompt is 'password:' followed by a redacted password. Below that, the system information is displayed: 'Linux Ch2Lab1 4.19.0-12-amd64 #1 SMP Debian 4.19.152-1 (2020-10-18) x86_64'. This is followed by a paragraph of text about Debian GNU/Linux being free software and the warranty disclaimer. The prompt returns to 'mabanque@Ch2Lab1:~\$' with a green cursor.

Saisissez le mot de passe du compte pour accéder à l'invite de commande (shell).

Installer le service SSH

- ouvrez une invite de commandes sur le serveur Debian et lancez l'installation d'OpenSSH.

```
$ sudo apt install openssh-server
```

Accéder à distance à la VM

Vous accédez à distance en SSH en utilisant la **commande ssh** sous Linux, MacOS et Windows ou en utilisant un logiciel comme **PuTTY** sous Windows.

Vous pouvez ouvrir une session afin d'administrer votre serveur :

- en utilisant le compte **root**, solution pratique mais **déconseillée** pour des raisons de sécurité ;
- ou en utilisant un **compte qui n'est pas root**, disposant de moins de droits, mais qui est configuré afin d'avoir la possibilité d'obtenir une **élévation de privilèges** quand cela est nécessaire avec l'utilisation de **sudo**.

Accès à distance avec le compte root (déconseillé)

Pour des raisons de sécurité, l'**accès en SSH avec le compte root est interdit**. Pour le permettre, le fichier **/etc/ssh/sshd_config** doit être modifié :

- la ligne suivante doit être commentée

```
# PermitRootLogin prohibit-password
```

- ajoutez la ligne suivante :

```
PermitRootLogin yes
```

Relancez le service ssh

```
# systemctl restart ssh
```

Pour ouvrir une session à distance avec SSH utilisez maintenant :

- le compte **root** ;
- le mot de passe que vous avez défini (**btssio** suggéré).

Accès à distance avec le compte qui n'est pas root

Si nécessaire **créez un compte linux** sur votre VM en utilisant la commande **adduser**. Renseignez le mot de passe et les autres informations demandées :

```
# adduser btssio
```

Donnez maintenant la possibilité à ce compte d'avoir une **élévation de privilèges** :

- installer sudo

```
# apt install sudo
```

- ajouter l'utilisateur nouvellement créé au groupe sudo

```
# adduser btssio sudo
```

ou

```
# usermod -aG sudo btssio
```

Accès au serveur en ligne de commande avec SSH

Il est nécessaire de disposer d'un client SSH.

Un client SSH disponible sous Windows.

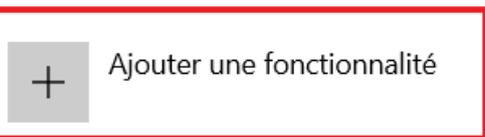
Pour installer le client SSH sous Windows :

- accédez à l'outil aux **fonctionnalités facultatives** de Windows ;
- puis ajouter le **Client OpenSSH**

← Paramètres

— □ ×

🏠 Fonctionnalités facultatives

[Voir l'historique des fonctionnalités facultatives](#)

Fonctionnalités installées

Trier par : Nom ▾

	Assistance rapide Microsoft	1,44 Mo
	Bloc-notes	316 Ko
	Client OpenSSH	5,05 Mo

* Lancez une **invite de commandes** et saisissez le nom du client OpenSSH suivi de l'adresse du serveur distant en précisant le compte de connexion :

```
C:>ssh compte@199.199.199.199
```

- saisissez ensuite le mot de passe du compte.

Il est possible de préciser le mot de passe à la connexion de la manière suivante (déconseillé)

```
C:>ssh compte:motdepasse@199.199.199.199
```

Configuration le clienr SSH

Lien : http://octetmalin.net/linux/tutoriels/ssh-fichier-etc-ssh_config-configuration-machine-client.php

Sur le client Linux, le fichier **/etc/ssh/ssh_config** permet de configurer les **paramètres globaux** du client pour toutes les connexions vers des serveurs **ssh**.

Pour chaque compte utilisateur, une configuration personnalisée se fait créant/modifiant le fichier

config situé dans le répertoire utilisateur **.ssh** (/home/[nom_utilisateur]/.ssh/config sous Linux)

Les options utilisables sont les même que celles du fichier **/etc/ssh/ssh_config**.

Exemple :

```
Host serveurssh
  Hostname 192.168.10.20
  Port 4242
  User root
```

Explication :

- Host ⇒ défini un nom pour le serveur ssh
- Hostname ⇒ adresse IP ou nom DNS du serveur
- Port ⇒ port ssh si différent du port SSH par défaut 22
- User ⇒ précise le nom de connexion

Utilisation

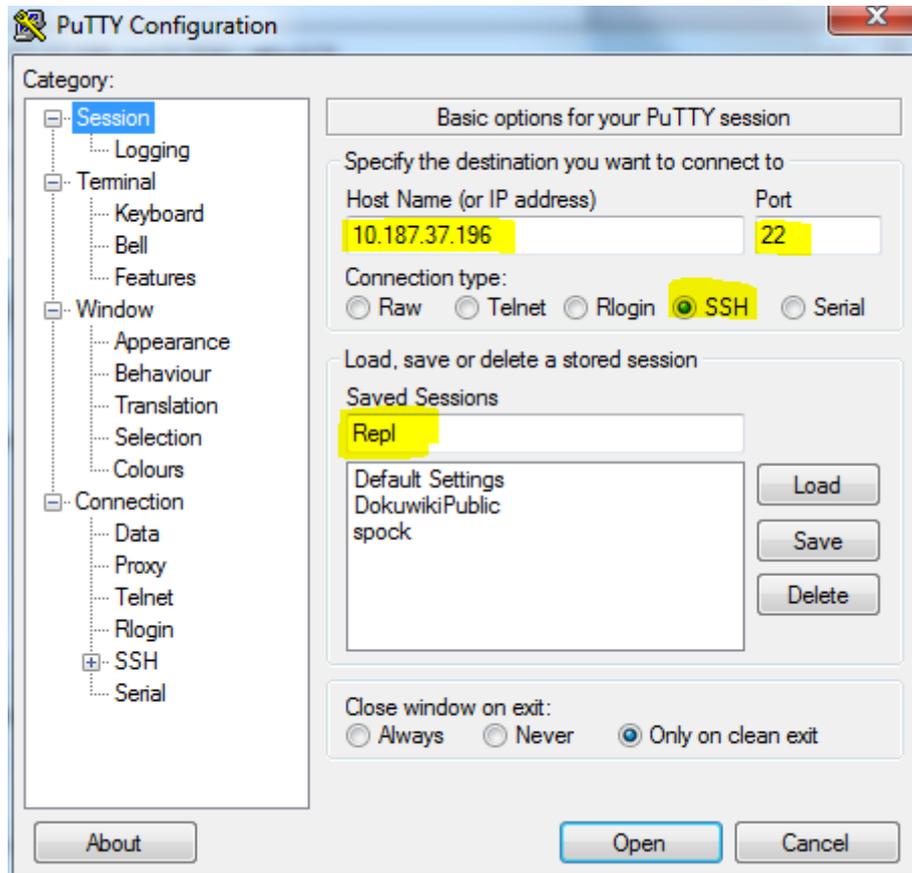
```
ssh serveurssh
```

Autres options

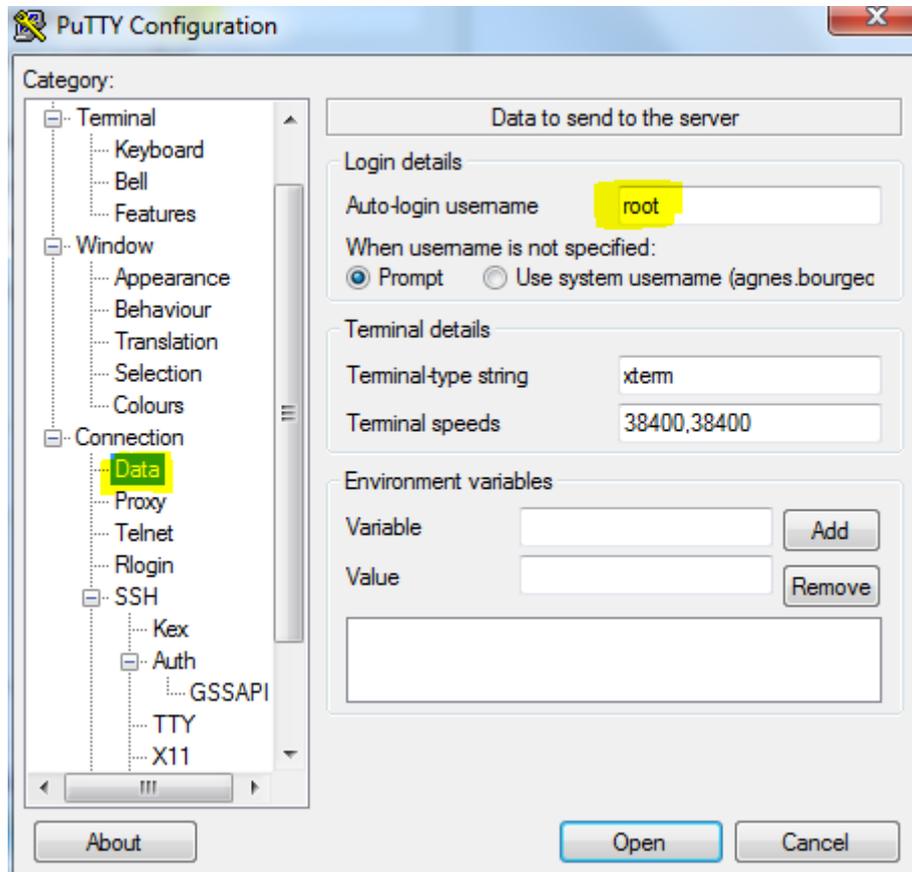
- Hote : Permet de définir vers quel machine les paramètres vont s'appliquer, l'étoile veut dire toutes.
- RSAAuthentication yes : indique de tenter une authentification RSA, clé publique/privé généré avec "ssh-keygen"
- PubKeyAuthentication yes : authentification avec une clé public.
- PasswordAuthentication yes : autorise l'authentification de base avec mot de passe.
- CheckHostIP yes : Spécifie si le ssh doit vérifier l'adresse IP de l'hôte qui se connectent au serveur pour détecter une usurpation DNS.
- IdentityFile ~/.ssh/id_dsa : définit la clé privé a utiliser pour s'authentifier lors de la connexion au serveur
- User nom_du_compte : définit le nom du compte utilisateur à distance à utiliser pour ce connecter.
- Port 22 : numéro de port du serveur SSH distant.

Configuration de l'accès SSH au serveur avec Putty

- Configurez Putty pour avoir un accès **console** à votre serveur.
- Indiquez l'**adresse IP** de votre serveur et le **port 22 (SSH)**.



- précisez le compte **root** pour vous connecter :



- Cliquez sur **Open** pour lancer la connexion.

Last update: 2023/04/29
11:47

reseau:debian:ssh <https://siocours.lycees.nouvelle-aquitaine.pro/doku.php/reseau/debian/ssh?rev=1682761620>

From:

<https://siocours.lycees.nouvelle-aquitaine.pro/> - **Les cours du BTS SIO**

Permanent link:

<https://siocours.lycees.nouvelle-aquitaine.pro/doku.php/reseau/debian/ssh?rev=1682761620>

Last update: **2023/04/29 11:47**

