

Rapport SAE 5.01 - Concevoir, réaliser et présenter une solution technique

Sommaire :

Phase 1: Conception et Planification	1
Choix de l'hyperviseur.....	1
Répartition des tâches.....	1
Réalisation du gantt.....	2
Phase 2: Installation et Configuration de l'Hyperviseur	2
Installation Proxmox.....	2
Création d'un utilisateur sur proxmox.....	4
Phase 3: Création des Machines Virtuelles Modèles	4
1. Création d'une machine "Réseau".....	4
2. Création d'une machine "Développeur".....	6
3. Connexion au réseau.....	7
Phase 4: Interface Web pour l'Accès à Distance (en cours)	8
Mise en place de guacamol.....	8
Création d'une base de donnée.....	10
Test et réalisation.....	12

Phase 1: Conception et Planification

Choix de l'hyperviseur

Hyperviseur	Avantages	Inconvénients
Proxmox	<ul style="list-style-type: none">● Interface conviviale● Combinaison de KVM et LXC● Coût nul	<ul style="list-style-type: none">● Support limité par rapport à des solutions commerciales
VMWare	<ul style="list-style-type: none">● Robustesse● Support complet● Écosystème mature	<ul style="list-style-type: none">● Coûteux● Complexité pour les petites installations
KVM	<ul style="list-style-type: none">● Open-source● Flexible● Sans coûts de licence	<ul style="list-style-type: none">● Gestion moins intuitive pour les débutants
Hyper-V	<ul style="list-style-type: none">● Intégration Windows● Rentabilité pour les utilisateurs de Microsoft	<ul style="list-style-type: none">● Moins de fonctionnalités avancées comparé à VMware

Nous avons donc choisis l'hyperviseur **PROXMOX**.

Répartition des tâches

Conception et Planification : Tout le monde

Interface Web pour l'Accès à Distance : Théo Six

Installation et Configuration de l'Hyperviseur : Jules Verstaevel

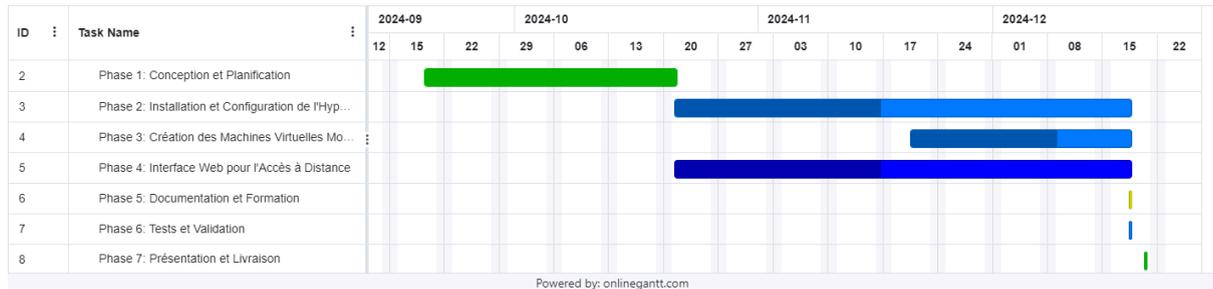
Création des Machines Virtuelles Modèles : Alexy Duquenoy

Documentation et Formation : Alexy Duquenoy

Tests et Validation : Tout le monde

Présentation et Livraison : Tout le monde

Réalisation du gantt



Phase 2: Installation et Configuration de l'Hyperviseur

Installation Proxmox

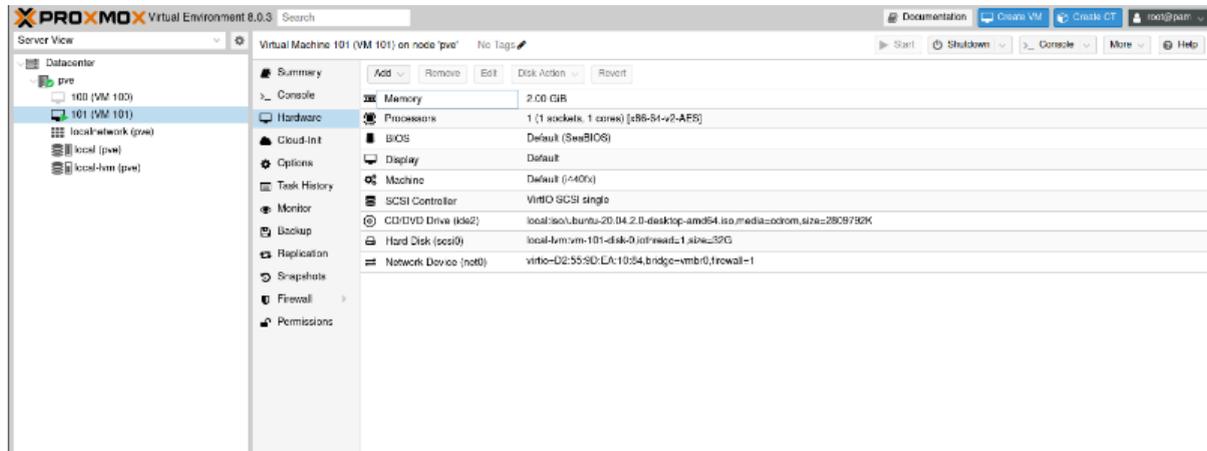
```

root@pve:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOW group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:00:08:09 brd ff:ff:ff:ff:ff:ff
    inet 172.31.19.9/29 brd 172.31.19.255 scope global dynamic enp0s3
        valid_lft 7775608sec preferred_lft 7775608sec
    inet6 fe80::a00:27ff:fe00:309/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:9f:c9:64 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.10/24 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe9f:c964/64 scope link
        valid_lft forever preferred_lft forever
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master vbr0 state UP group default qlen 1000
    link/ether 08:00:27:c0:43:f0 brd ff:ff:ff:ff:ff:ff
    inet 10.0.4.15/24 brd 10.0.4.255 scope global dynamic enp0s9
        valid_lft 86008sec preferred_lft 86008sec
5: vbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 08:00:27:c0:43:f0 brd ff:ff:ff:ff:ff:ff
    inet 10.0.4.15/24 brd 10.0.4.255 scope global dynamic vbr0
        valid_lft 86008sec preferred_lft 86008sec
    inet6 fe80::a00:27ff:fe00:43f0/64 scope link
        valid_lft forever preferred_lft forever
6: tap10110: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master fubr10110 state UNKNOW group default qlen 1000
    link/ether 9a:48:3c:8c:48:b9 brd ff:ff:ff:ff:ff:ff
7: fubr10110: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether a6:be:28:11:bb:b9 brd ff:ff:ff:ff:ff:ff
8: fupr101p00fupr10110: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master vbr0 state UP group default qlen 1000
    link/ether 6e:a9:e4:ec:e8:3e brd ff:ff:ff:ff:ff:ff
9: fupr101100fupr101p0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master fubr10110 state UP group default qlen 1000
    link/ether 0a:c1:74:12:8b:c9 brd ff:ff:ff:ff:ff:ff
    
```

Une interface côté IUT pour pouvoir accéder à l'interface web :



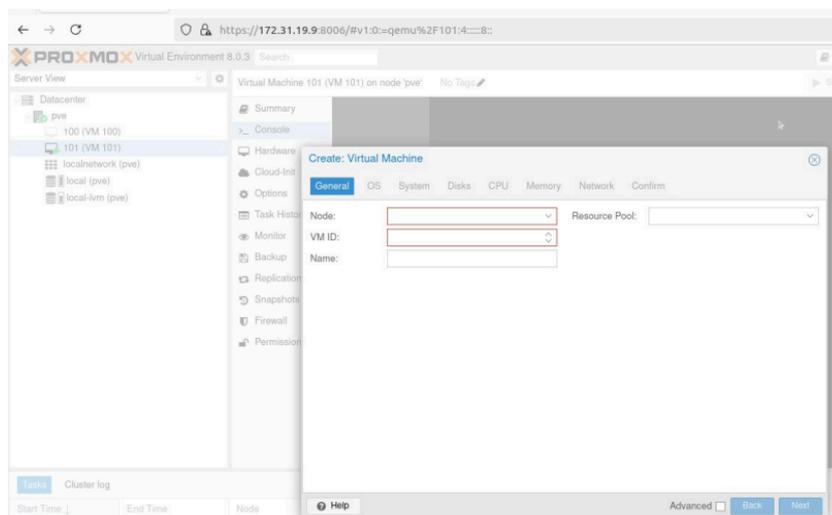
Une interface côté TP pour communiquer avec d'autres machines (192.168.56.10/24).
 Et une interface en NAT pour les machines virtuelles configurées sur PROXMOX.
 (10.0.4.15/24)



Suite au lancement, on peut procéder à l'installation d'une machine :

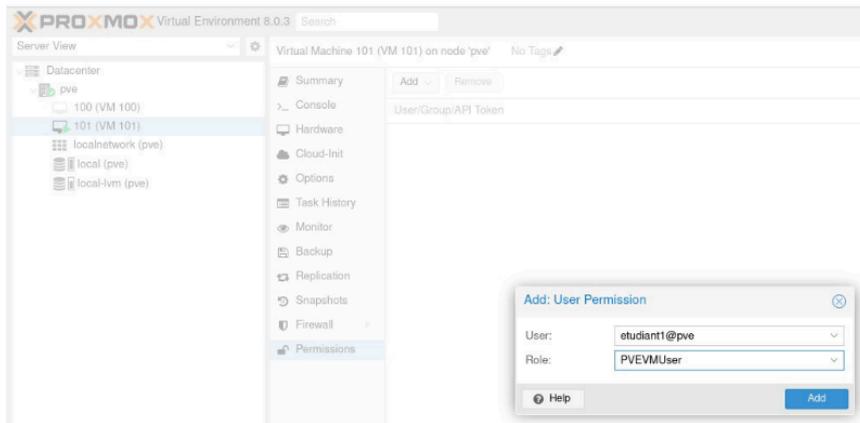


Suite au lancement, on peut procéder à l'installation d'une machine :
 Créer une machine virtuelle sur proxmox



Création d'un utilisateur sur proxmox

```
pveum user add etudiant1@pve --password progtr00
```



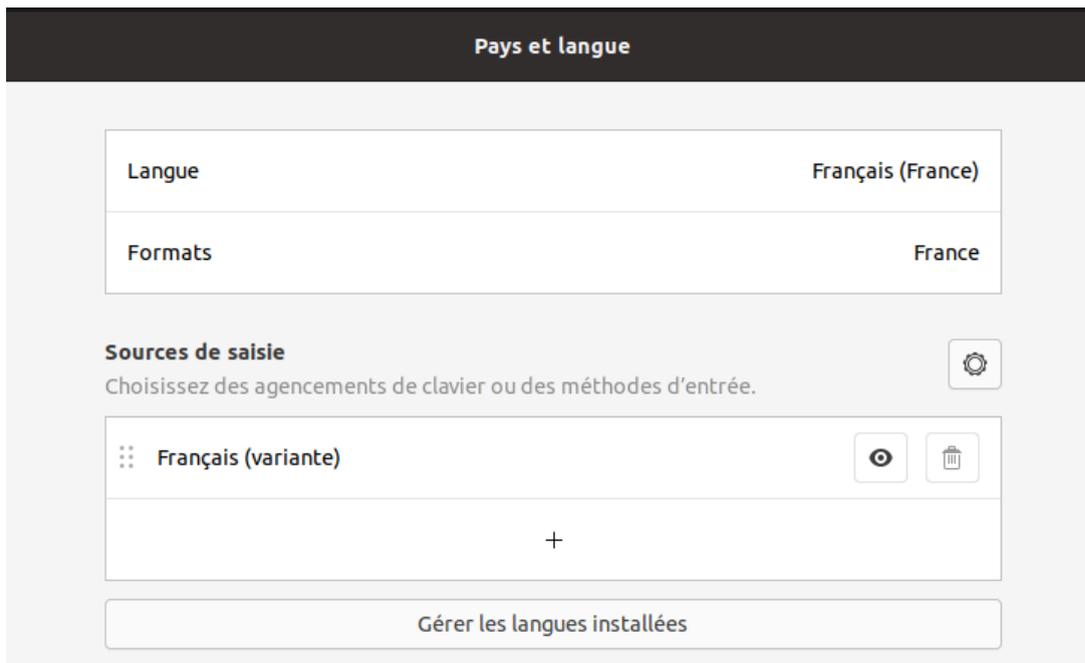
Phase 3: Création des Machines Virtuelles Modèles

1. Création d'une machine "Réseau"

On crée une machine Ubuntu (ubuntu-20.04.2-desktop-amd64)

Général	
Nom :	20res
Système d'exploitation :	Ubuntu (64-bit)
System	
Mémoire vive :	2048 Mo
Processeurs :	2
Ordre d'amorçage :	Disque dur, Optique, Disquette
Accélération :	Pagination imbriquée, Paravirtualisation KVM
Affichage	
Mémoire vidéo :	16 Mo
Contrôleur graphique :	VMSVGA
Serveur de bureau à distance :	Désactivé
Enregistrement :	Désactivé
Stockage	
Contrôleur :	IDE
Maître secondaire IDE :	[Lecteur optique] Vide
Contrôleur :	SATA
Port SATA 0 :	20res.vdi (Normal, 25,00 Gio)
Audio	
Pilote hôte :	Par défaut
Contrôleur :	ICH AC97
Réseau	
Interface 1 :	Intel PRO/1000 MT Desktop (NAT)
USB	
Contrôleur USB :	OHCI, EHCI
Filtres de périphérique :	0 (0 actif)
Dossiers partagés	
Aucun	
Description	
Aucune	

Ensuite on configure la langue et le clavier :



Pour finir on installe les applications nécessaires :

On télécharge le .deb de cisco packet tracer

Puis on exécute les commandes :

```
su - root #afin de se mettre en sudo
```

```
apt update
```

```
dpkg -i CiscoPacketTracer822_amd64_signed.deb
```

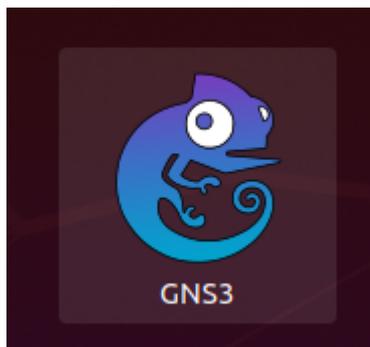


On télécharge ensuite GNS3 en exécutant les commandes suivantes :

```
add-apt repository ppa:gns3/ppa
```

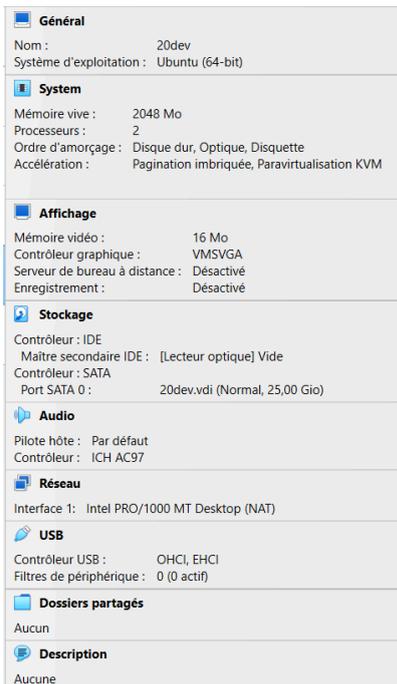
```
apt update
```

```
apt install gns3-gui gns3-server -y
```

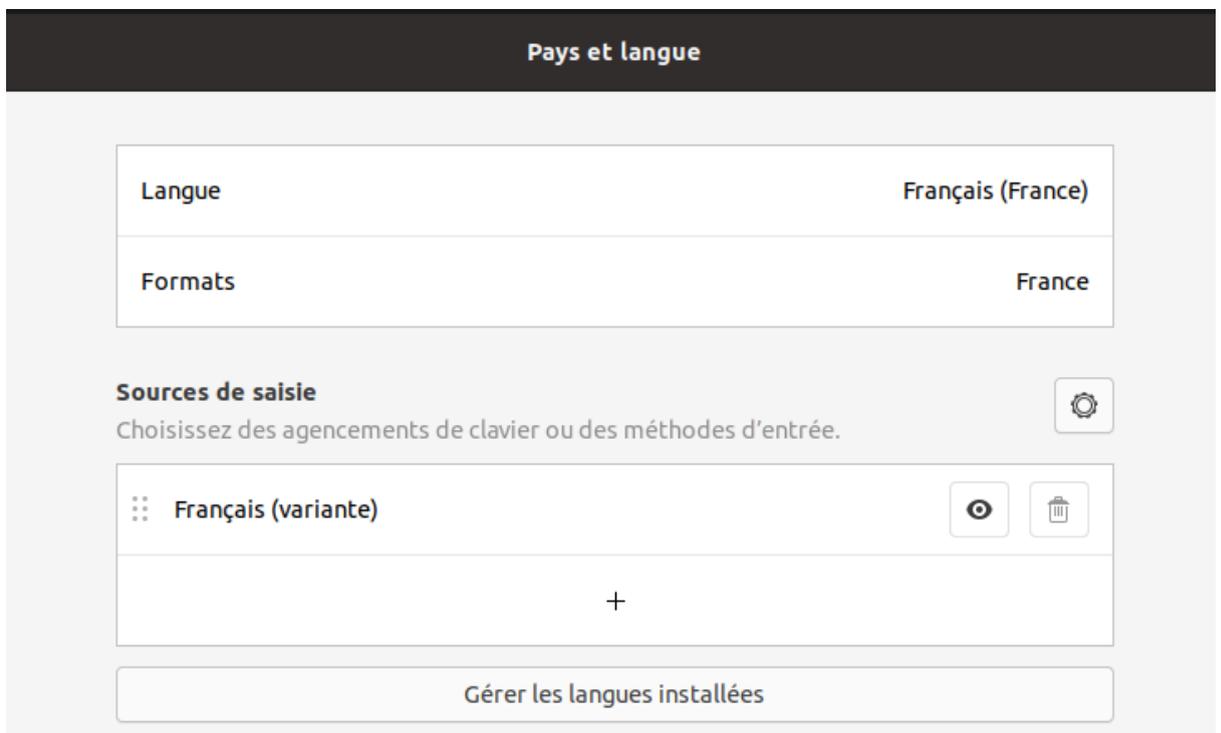


2. Création d'une machine "Développeur"

On crée une machine Ubuntu (ubuntu-20.04.2-desktop-amd64)

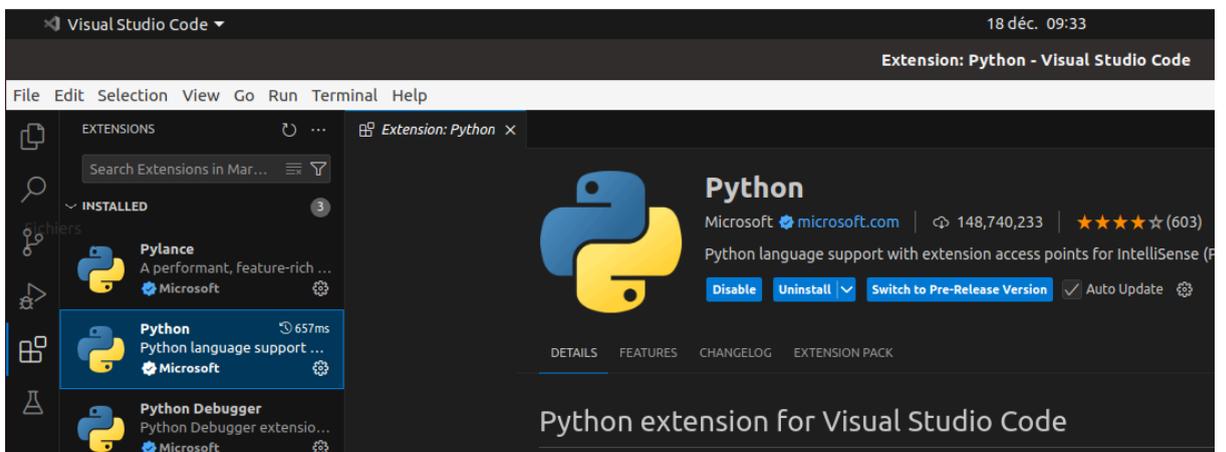
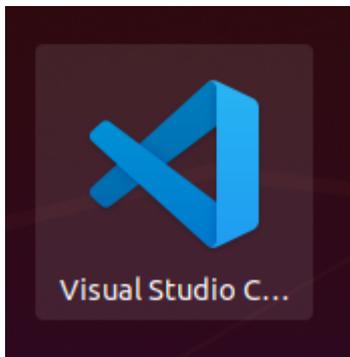


Ensuite on configure la langue et le clavier :



Pour finir on installe les applications nécessaires :

```
apt update
apt install python
snap install code --classic
```



3. Connexion au réseau

La machine obtient une adresse IP sur le réseau NAT (10.0.4.16/24).

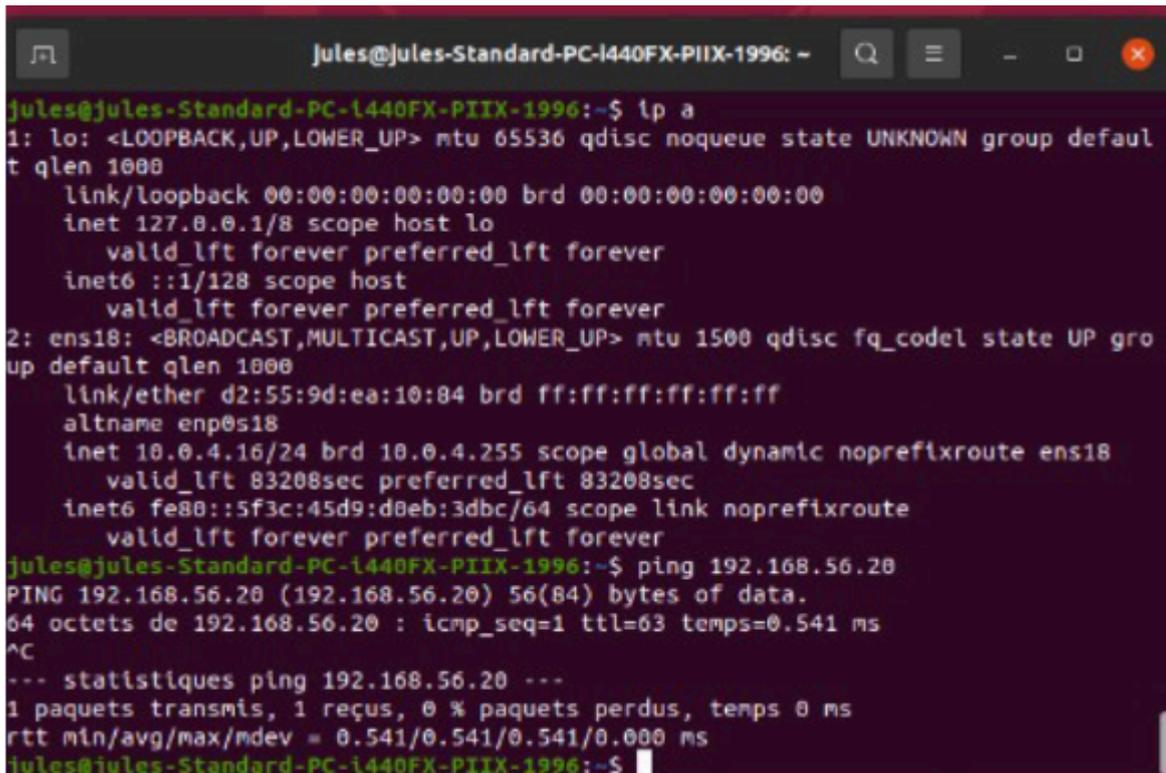
Pour pouvoir communiquer entre nos machines virtuelles proxmox et machines virtuelles de l'iut, il faut ajouter des routes :

```
jules@jules-Standard-PC-i440FX-PIIX-1996:~$ ip r
default via 10.0.4.15 dev ens18
10.0.4.0/24 dev ens18 proto kernel scope link src 10.0.4.16 metric 100
169.254.0.0/16 dev ens18 scope link metric 1000
192.168.56.10 dev ens18 scope link
```

sur la machine virtuelle guacamole IUT :

```
administrateur@rt-mv:~$ ip r
default via 192.168.56.10 dev enp0s8
10.0.4.15 dev enp0s8 scope link
192.168.56.0/24 dev enp0s8 proto kernel scope link src 192.168.56.20
```

ping entre machine virtuelle proxmox et machine iut :

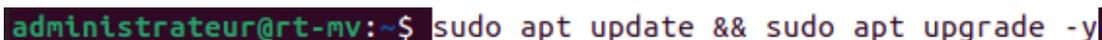


```
jules@jules-Standard-PC-i440FX-PIIX-1996:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether d2:55:9d:ea:10:84 brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 10.0.4.16/24 brd 10.0.4.255 scope global dynamic noprefixroute ens18
        valid_lft 83208sec preferred_lft 83208sec
    inet6 fe80::5f3c:45d9:d0eb:3dbc/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
jules@jules-Standard-PC-i440FX-PIIX-1996:~$ ping 192.168.56.20
PING 192.168.56.20 (192.168.56.20) 56(84) bytes of data:
64 octets de 192.168.56.20 : icmp_seq=1 ttl=63 temps=0.541 ms
^C
--- statistiques ping 192.168.56.20 ---
1 paquets transmis, 1 reçus, 0 % paquets perdus, temps 0 ms
rtt min/avg/max/mdev = 0.541/0.541/0.541/0.000 ms
jules@jules-Standard-PC-i440FX-PIIX-1996:~$
```

Pour créer une template de cette machine pour éviter de reconfigurer les paramètres réseaux à chaque création de machine ubuntu, il faut faire un clic droit sur la VM ciblé et « convert to template »

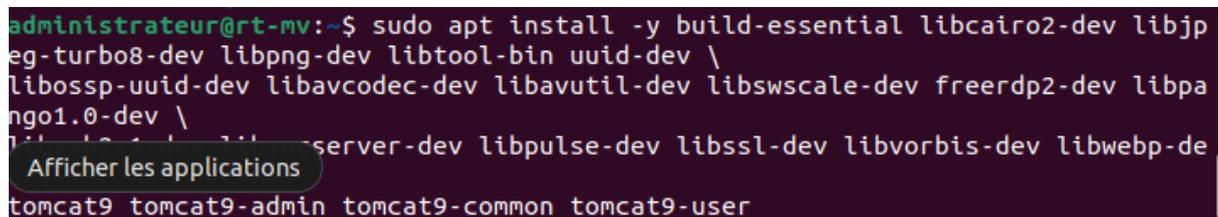
Phase 4: Interface Web pour l'Accès à Distance (en cours)

Mise en place de guacamol



```
administrateur@rt-mv:~$ sudo apt update && sudo apt upgrade -y
```

Installation des requis prévu pour le fonctionnement de guacamol



```
administrateur@rt-mv:~$ sudo apt install -y build-essential libcairo2-dev libjpeg-turbo8-dev libpng-dev libtool-bin uuid-dev \
libossp-uuid-dev libavcodec-dev libavutil-dev libswscale-dev freerdp2-dev libpng1.0-dev \
libltdl-dev libserver-dev libpulse-dev libssl-dev libvorbis-dev libwebp-dev
tomcat9 tomcat9-admin tomcat9-common tomcat9-user
```

Récupération avec un Wget de Guacamol ainsi que son extraction



```
administrateur@rt-mv:~$ wget https://downloads.apache.org/guacamole/1.5.3/binaries/guacamole-1.5.3.war
sudo mv guacamole-1.5.3.war /var/lib/tomcat9/webapps/guacamole.war
```

Première configuration de celui-ci

```
administrateur@rt-mv:~$ ./configure --with-init-dir=/etc/init.d
make
sudo make install
sudo ldconfig
```

Redémarrage des 2 services.

```
administrateur@rt-mv:~$ sudo systemctl enable guacd
sudo systemctl start guacd
```

```
administrateur@rt-mv:~$ sudo systemctl restart tomcat9
```

Modification de guacamol.properties

```
administrateur@rt-mv:~$ sudo mkdir /etc/guacamole
sudo nano /etc/guacamole/guacamole.properties
```

```
GNU nano 6.2 /etc/guacamole/guacamole.properties
## Hostname and port for guacd
guacd-hostname: localhost
guacd-port: 4822

# User-mapping for basic authentication
user-mapping: /etc/guacamole/user-mapping.xml
```

Ensuite nous allons modifier le fichier user-mapping. Ici nous déclarons les utilisateurs pour guacamole. Nous explorerons une nouvelle piste en charge des utilisateurs plus tard.

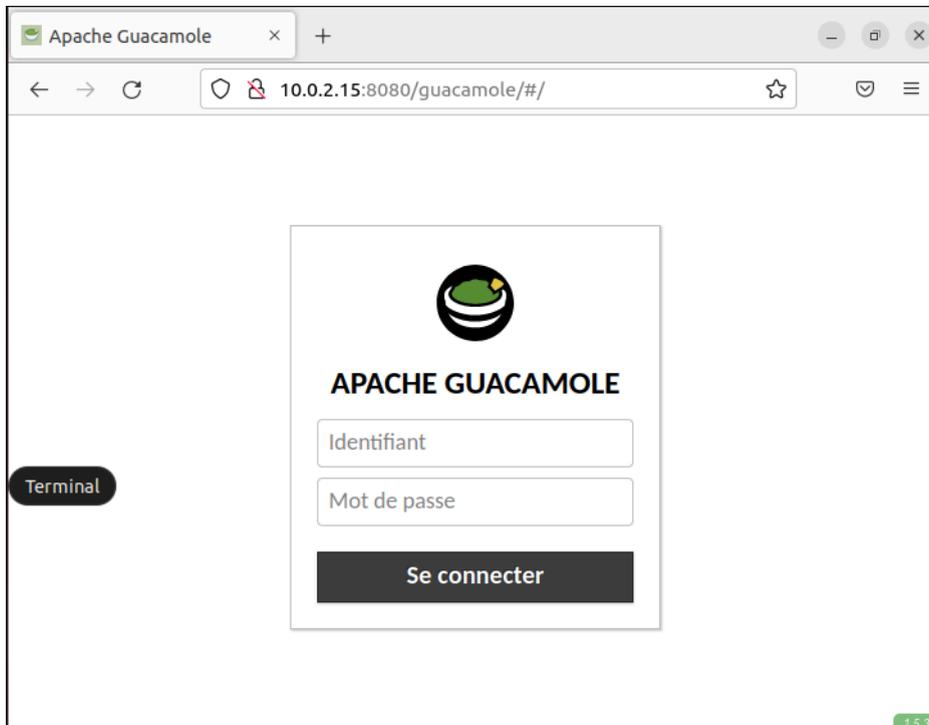
```
administrateur@rt-mv:~$ sudo nano /etc/guacamole/user-mapping.xml
```

Ici on déclare une connexion SSH vers 192.168.1.100

```
GNU nano 6.2 /etc/guacamole/user-mapping.xml
<user-mapping>
  <authorize username="etudiant1" password="progtr00">
    <connection name="VM SSH">
      <protocol>ssh</protocol>
      <param name="hostname">192.168.1.100</param>
      <param name="port">22</param>
      <param name="username">ubuntu</param>
      <param name="password">progtr00</param>
    </connection>
  </authorize>
</user-mapping>
```

```
administrateur@rt-mv:~$ sudo systemctl restart guacd tomcat9
```

Nous pouvons ensuite effectuer nos premiers tests. Ici on observe que l'interface guacamol est bien joignable.



Création d'une base de donnée

Nous allons ensuite créer une base de données pour nos informations.

```
administrateur@rt-mv:~$ sudo apt install mysql-server  
sudo mysql_secure_installation
```

```
administrateur@rt-mv:~$ sudo mysql -u root -p  
CREATE DATABASE guacamole_db;  
CREATE USER 'guacamole_user'@'localhost' IDENTIFIED BY 'password';  
GRANT SELECT,INSERT,UPDATE,DELETE ON guacamole_db.* TO 'guacamole_user'@'localh  
ost';  
FLUSH PRIVILEGES;  
EXIT;
```

Dans cette base de donnée nous utilisons un template pour guacamol afin de gérer les utilisateurs.

```
administrateur@rt-mv:~$ wget https://downloads.apache.org/guacamole/1.5.3/binar  
y/guacamole-auth-jdbc-1.5.3.tar.gz
```

```
administrateur@rt-mv:~$ cd guacamole-auth-jdbc-1.5.3/mysql/schema
```

```
administrateur@rt-mv:~$ tar -xvzf guacamole-auth-jdbc-1.5.3.tar.gz
```

```
administrateur@rt-mv:~/guacamole-server-1.5.3/guacamole-auth-jdbc-1.5.3/mysql/schema$ cat *.sql | mysql -u root -p guacamole_db
```

```
administrateur@rt-mv:~/guacamole-server-1.5.3/guacamole-auth-jdbc-1.5.3/mysql/schema$ sudo cp guacamole-auth-jdbc-mysql-1.5.3.jar /etc/guacamole/extensions/
```

Nous déclarons ensuite la base de donnée dans guacamol.properties

```
GNU nano 6.2 /etc/guacamole/guacamole.properties
# Hostname and port for guacd
guacd-hostname: localhost
guacd-port: 4822

# User-mapping for basic authentication
user-mapping: /etc/guacamole/user-mapping.xml

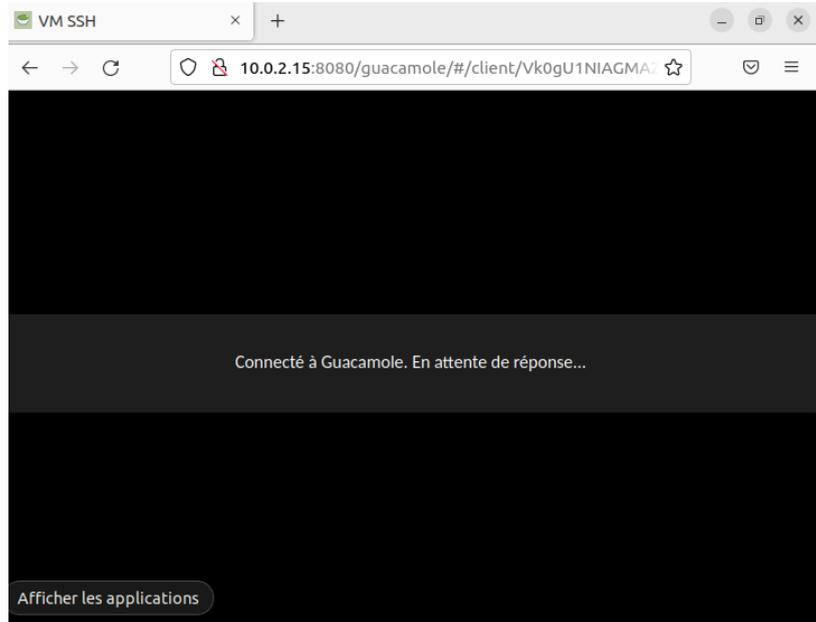
# MySQL properties
mysql-hostname: localhost
mysql-port: 3306
mysql-database: guacamole_db
mysql-username: guacamole_user
mysql-password: password
```

Nous créons aussi des certificats.

```
administrateur@rt-mv:~/guacamole-server-1.5.3/guacamole-auth-jdbc-1.5.3/mysql/schema$ sudo apt install certbot python3-certbot-apache
```

```
...
administrateur@rt-mv:~/guacamole-server-1.5.3/guacamole-auth-jdbc-1.5.3/mysql/schema$ sudo certbot --apache
```

Lors d'une connexion avec les identifiants trouvés précédemment la connexion se lance. Aucune réponse pour l'instant car en effet nous n'avons pas encore de client configuré.



Test et réalisation

Pour les tests nous n'utilisons pas la base de donnée mise en place afin de gérer plus simplement les modifications.

Dans notre User-mapping nous déclarons 2 instances, l'une en ssh et l'autre en vnc. La piste du vnc a été écartée pour le moment au profit d'une connexion rdp.

```
Ouvrir [ ] *user-mapping.xml /etc/guacamole Enregistrer [ ] [ ] [ ] [ ]
1 <user-mapping>
2   <!-- Utilisateur etudiant1 -->
3   <authorize username="etudiant1" password="progr00">
4     <connection name="VM SSH">
5       <protocol>ssh</protocol>
6       <param name="hostname">10.0.3.5</param>
7       <param name="port">22</param>
8       <param name="username">administrateur</param>
9       <param name="password">progr00</param>
10    </connection>
11  </authorize>
12
13  <!-- Nouvel utilisateur etp avec connexion RDP -->
14  <authorize username="etudiant2" password="progr00">
15    <connection name="VM RDP Etp">
16      <protocol>vnc</protocol>
17      <param name="hostname">10.0.3.5</param>
18      <param name="port">5901</param>
19      <param name="username">administrateur</param>
20      <param name="password">progr00</param>
21      <param name="resolution">1920x1080</param>
22    </connection>
23  </authorize>
24 </user-mapping>
```

La connexion RDP fonctionne cependant nous n'avons pas d'affichage sur guacamol, d'après nos recherches cela est liée à un problème de taille de la page affichée par rdp.

```

[20241218-11:41:43] [INFO ] started connecting
[20241218-11:41:43] [INFO ] lib_mod_connect: connecting via UNIX socket
[20241218-11:41:44] [INFO ] lib_mod_log_peer: xrdp_pid=5920 connected to X11rdp
pid=5923 X11rdp_uid=1000 X11rdp_gid=1000 client_ip=::ffff:10.0.3.4 client_port=3
5110
[20241218-11:41:44] [INFO ] connected ok
  
```

Cependant la connexion ssh est quant à elle fonctionnelle.

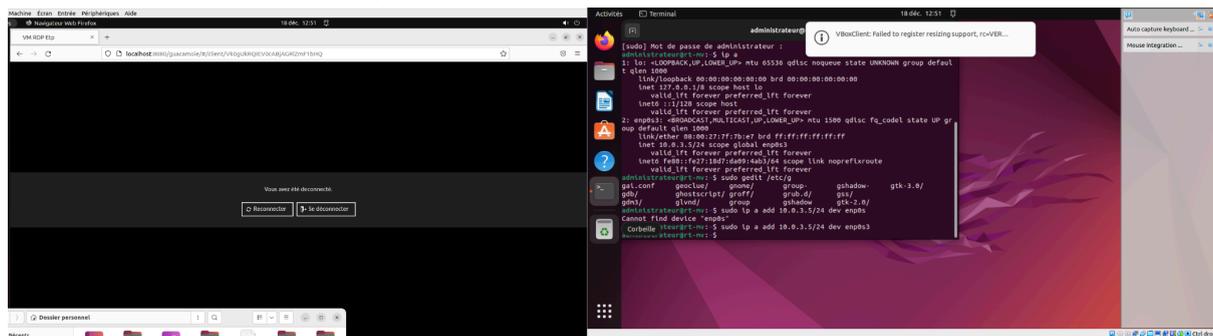
```

administrateur@rt-mv:~$ x2goclient
qt.qpa.xcb: could not connect to display
qt.qpa.plugin: Could not load the Qt platform plugin "xcb" in "" even though it was found.
This application failed to start because no Qt platform plugin could be initialized. Reinstalling the application may fix this problem.

Available platform plugins are: eglfs, linuxfb, minimal, minimalegl, offscreen, vnc, xcb.

Abandon (core dumped)
administrateur@rt-mv:~$ WARNING: failed to kill process group '7781': No such process
* Documentation: https://help.ubuntu.com
Bureau : commande introuvable
administrateur@rt-mv:~$ * Management: https://landscape.canonical.com
Bureau : commande introuvable
administrateur@rt-mv:~$ * Support: https://ubuntu.com/advantage
Bureau : commande introuvable
administrateur@rt-mv:~$
administrateur@rt-mv:~$
administrateur@rt-mv:~$ 359 mises à jour peuvent être appliquées immédiatement.
359 : commande introuvable
administrateur@rt-mv:~$ 359 de ces mises à jour sont des mises à jour de sécurité.
359 : commande introuvable
administrateur@rt-mv:~$ Pour afficher ces mises à jour supplémentaires, exécutez : apt list --upgradable
Pour : commande introuvable
administrateur@rt-mv:~$
administrateur@rt-mv:~$ Last login: Wed Dec 18 12:10:52 2024 from 10.0.3.4
La commande « Last » n'a pas été trouvée, voulez-vous dire :
  commande « last » du deb util-linux (2.37.2-4ubuntu3.4)
  commande « nast » du deb nast (0.2.0-9)
Essayez : sudo apt install -onem du deb>
administrateur@rt-mv:~$ administrateur@rt-mv:~$
administrateur@rt-mv:~$ : commande introuvable
administrateur@rt-mv:~$ administrateur@rt-mv:~$
administrateur@rt-mv:~$ : commande introuvable
administrateur@rt-mv:~$ administrateur@rt-mv:~$
administrateur@rt-mv:~$ administrateur@rt-mv:~$ x2goclient
administrateur@rt-mv:~$ : commande introuvable
administrateur@rt-mv:~$ qt.qpa.xcb: could not connect to display
  
```

Nous observons donc ici un problème lié à la connexion RDP.

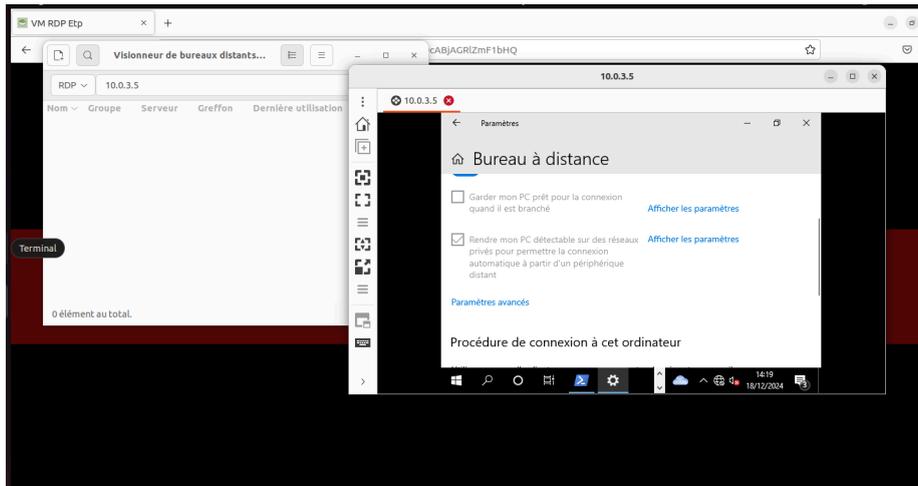


Nous sommes ensuite partie tester la connexion rdp sur une machine windows. Certains paramètres doivent être adaptés concernant le certificat.

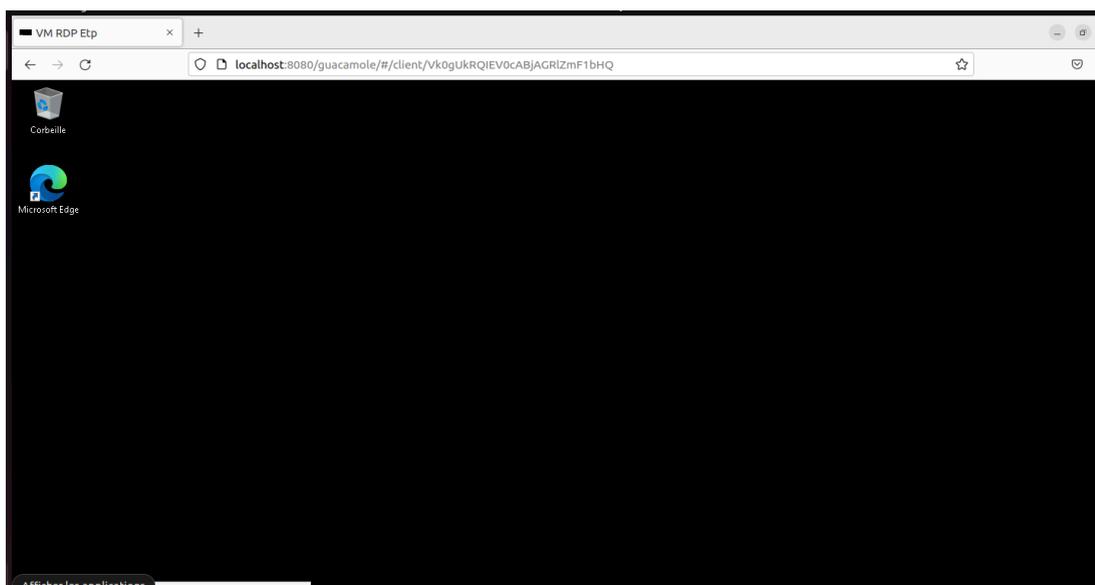
```

21 <param name="resolution">1530x814</param>
22 <param name="ignore-cert">>true</param>
23 <param name="security">any</param>
  
```

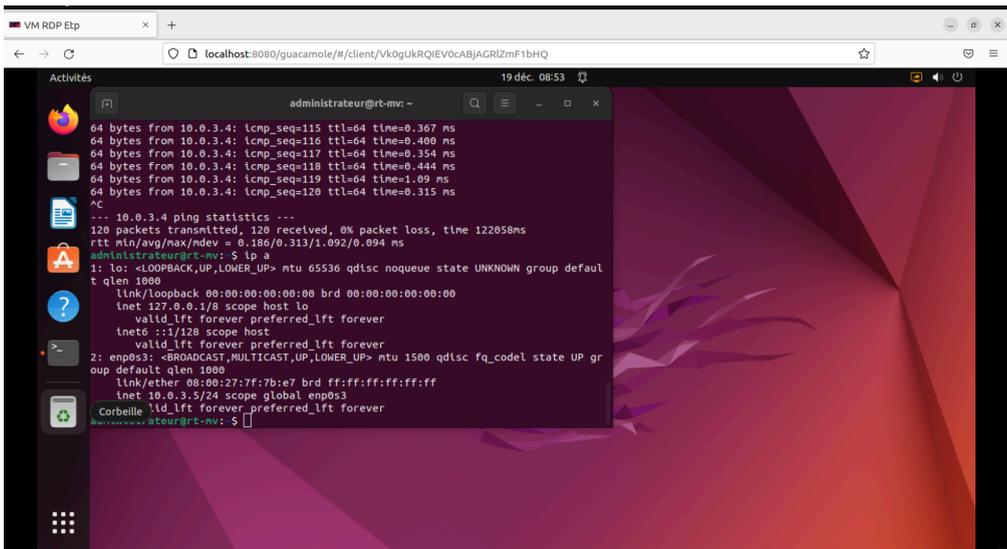
Dans un premier temps nous avons testé la connexion avec un outil nommé Remmina. Nous avons ici vue que la communication était possible et fonctionnelle



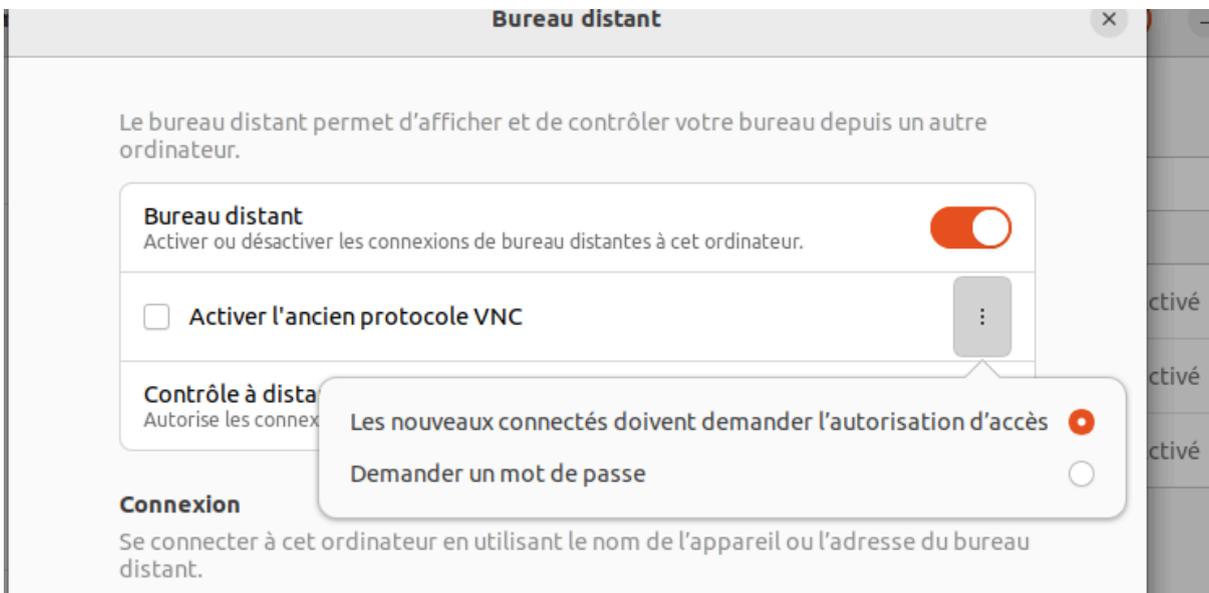
Nous avons donc ensuite testé avec guacamol qui nous a donné des résultats positifs. Nous pouvions donc a ce point utiliser guacamol pour joindre des Machine virtuel windows.



Nous devons donc encore faire fonctionner le process sur des machines ubuntu. Nous sommes alors partis sur un autre protocole que le RDP. Nous avons connecté les machines en VNC et plus précisément avec TigerVNC. Ceci nous a donc permis d'obtenir ce résultat sur guacamol et d'avoir la possibilité d'utiliser guacamol pour lancer des machine linux et windows.



Une fois dans proxmox il ne restera qu'à faire les mêmes modifications. Nous avons donc aussi créé une connexion SSH sur guacamol au serveur proxmox avec des identifiant administrateur. Voici la modification qui bloquait notre VNC. Nous avons activé l'ancien protocole de VNC.



Pour conclure voici notre fichier User-Mapping à la fin des différents tests. Avec un utilisateur SSH, un RDP, ou VNC et le dernier en ssh pour le serveur proxmox.

```

1 user-mapping
2 <!-- Utilisateur etudiant1 -->
3 <authorize username="etudiant1" password="progtr00">
4   <connection name="VM SSH">
5     <protocol>ssh</protocol>
6     <param name="hostname">10.0.3.5</param>
7     <param name="port">22</param>
8     <param name="username">administrateur</param>
9     <param name="password">progtr00</param>
10  </connection>
11 </authorize>
12
13 <!-- Nouvel utilisateur etp avec connexion RDP -->
14 <authorize username="etudiant2" password="progtr00">
15   <connection name="VM RDP Etp">
16     <protocol>rdp</protocol>
17     <param name="hostname">10.0.3.5</param>
18     <param name="port">3389</param>
19     <param name="username">admin</param>
20     <param name="password">progtr00</param>
21     <param name="resolution">1530x814</param>
22     <param name="ignore-cert">true</param>
23     <param name="security">any</param>
24   </connection>
25 </authorize>
26 <!-- Nouvel utilisateur etp avec connexion RDP -->
27 <authorize username="etudiant3" password="progtr00">
28   <connection name="VM RDP Etp">
29     <protocol>vnc</protocol>
30     <param name="hostname">10.0.3.6</param>
31     <param name="port">5900</param>
32     <param name="password">progtr00</param>
33     <param name="resolution">1530x814</param>
34   </connection>
35 </authorize>
36 <authorize username="admin" password="progtr00">
37   <connection name="VM RDP Etp">
38     <protocol>ssh</protocol>
39     <param name="hostname">10.0.3.1</param>

```

Phase 5: Documentation et Formation

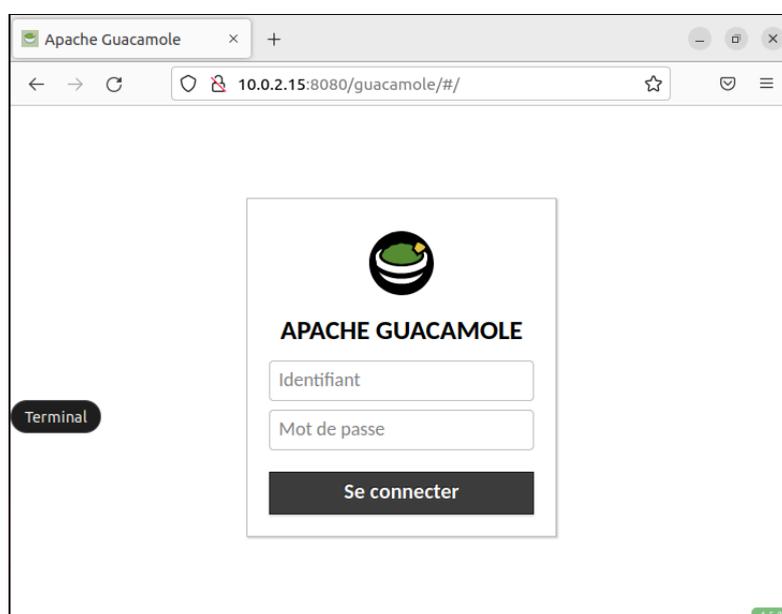
Voici un manuel pour les utilisateurs de notre solution.

Dans un premier temps joignez l'administrateur de la solution qui une fois dans la base de donnée vous donnera des identifiant de connexion.

Joignez la page Web :

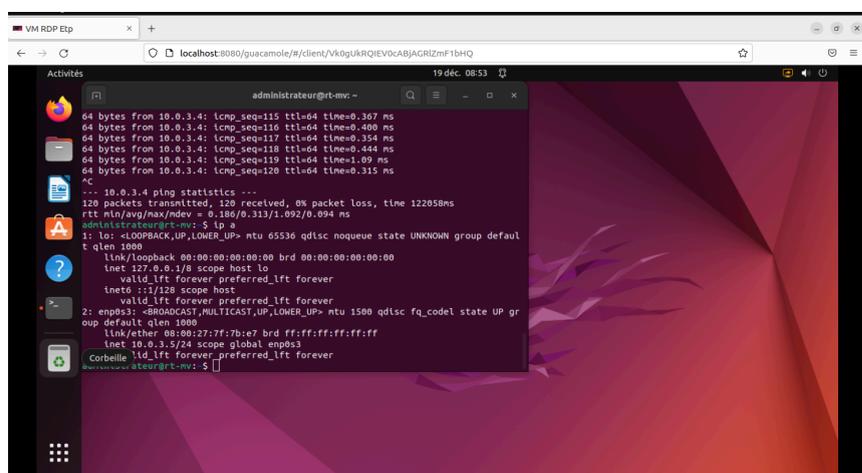
<https://10.0.3.4:8080/Guacamole>

Une fois le lien rentré dans le navigateur vous devrez vous retrouver face à cette page.



Entrer les identifiants fournis par votre administrateur réseau.

Cela vous redirigera vers la Machine virtuel qui vous a attirée par celui-ci.



La Machine virtuel a déjà été configurée pour répondre à vos besoins.
Un package d'application vous sera fourni avec votre compte utilisateur.
Par exemple Wireshark, Packet Tracer ou Visual Studio.
Il ne vous reste plus qu'à profiter de votre interface.
Pour tout problème ou modification n'hésitez pas à joindre votre administrateur.