

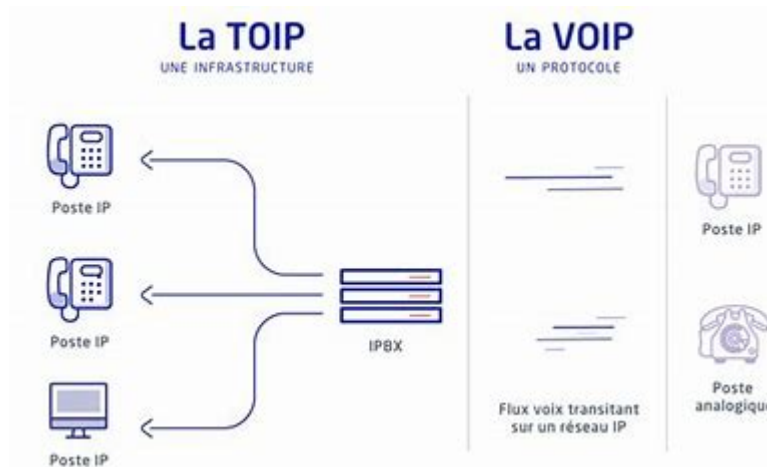
Projet VoIP

contexte :

Qu'est-ce qu'un téléphone VOIP (« Voice over Internet Protocol ») ? Le téléphone VoIP est une forme de communication via internet, cette technologie permet d'acheminer des appels téléphoniques grâce à une connexion internet. Un appareil mobile, un ordinateur de bureau, une tablette, un navigateur Web ou un téléphone de bureau permet de passer des appels téléphoniques via une connexion internet. Les prémices de la voix sur IP datent de 1928. Cette technologie permet d'enregistrer, synthétiser et encoder la voix pour pouvoir ensuite la transporter sur un réseau.

Cette invention, mise en application au sein du Voder, constitue les fondations de la VoIP et a été présentée en 1939 à l'exposition universelle de New York.

En 1999, Mark Spencer, développe la solution **d'IPBX** open source Asterisk. Toutes les cartes sont maintenant sur la table pour permettre sa démocratisation auprès des entreprises et du grand public.



Les avantages d'un téléphone VOIP :

- Le coût
- La flexibilité / l'évolutivité
- Mobilité
- Portabilité

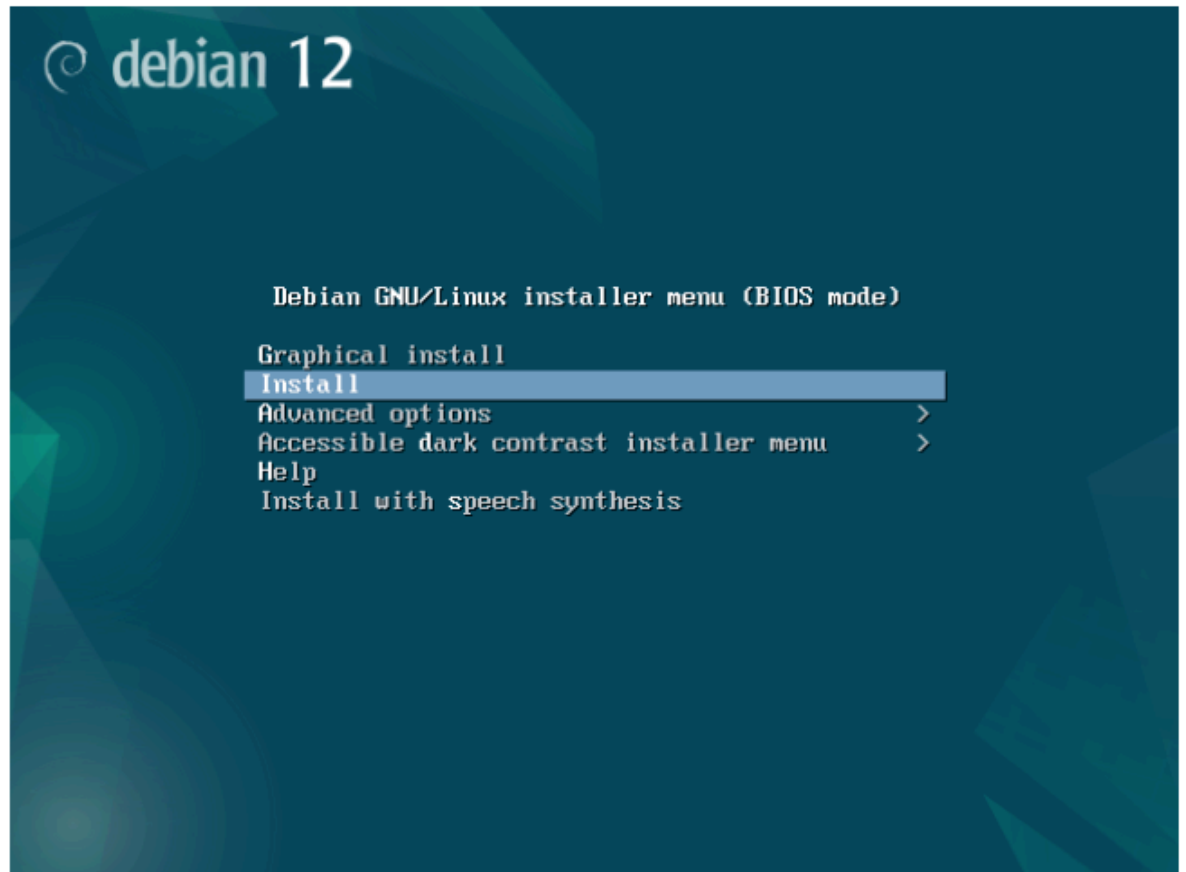
L'inconvénient d'un téléphone VOIP :

- Panne de courant

Exemples d'applications :

- Skype
- Zoom
- Whatsapp

1- Installation de debian avec interface graphique :



[!] Configure the network

Please enter the hostname for this system.

The hostname is a single word that identifies your system to the network. If you don't know what your hostname should be, consult your network administrator. If you are setting up your own home network, you can make something up here.

Hostname:

voipserver

<Go Back>

<Continue>

[!!] Set up users and passwords

You need to set a password for 'root', the system administrative account. A malicious or unqualified user with root access can have disastrous results, so you should take care to choose a root password that is not easy to guess. It should not be a word found in dictionaries, or a word that could be easily associated with you.

A good password will contain a mixture of letters, numbers and punctuation and should be changed at regular intervals.

The root user should not have an empty password. If you leave this empty, the root account will be disabled and the system's initial user account will be given the power to become root using the "sudo" command.

Note that you will not be able to see the password as you type it.

Root password:

☐ Show Password in Clear

<Go Back>

<Continue>

[!!] Set up users and passwords

A user account will be created for you to use instead of the root account for non-administrative activities.

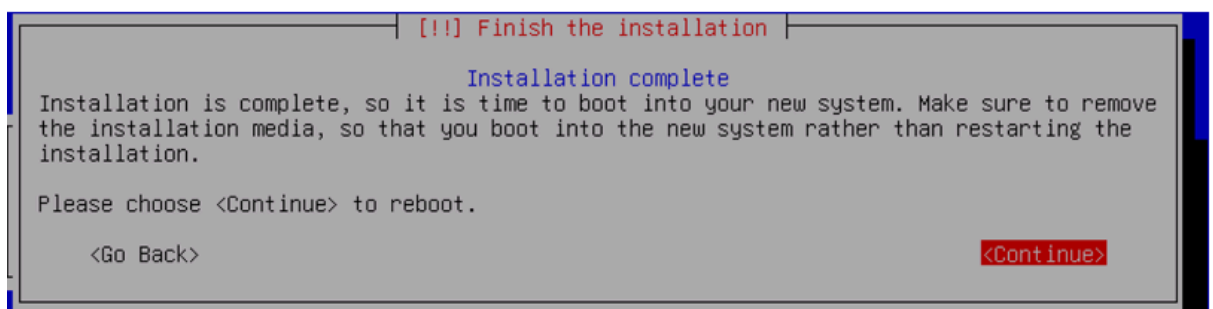
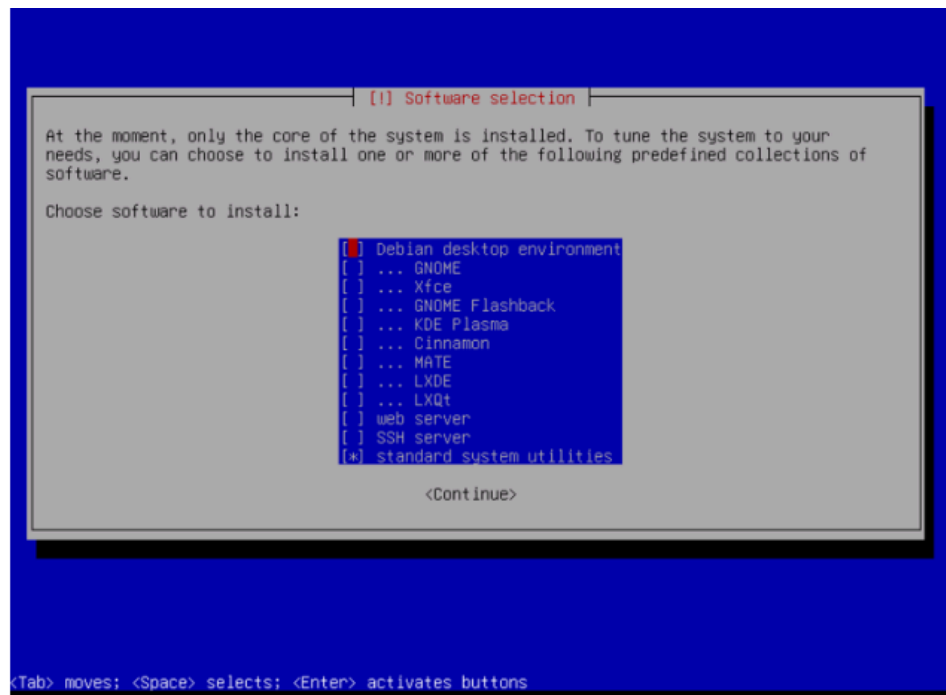
Please enter the real name of this user. This information will be used for instance as default origin for emails sent by this user as well as any program which displays or uses the user's real name. Your full name is a reasonable choice.

Full name for the new user:

VoIP Server

<Go Back>

<Continue>



1. Faire les mises à jour

"apt update"
"apt upgrade"

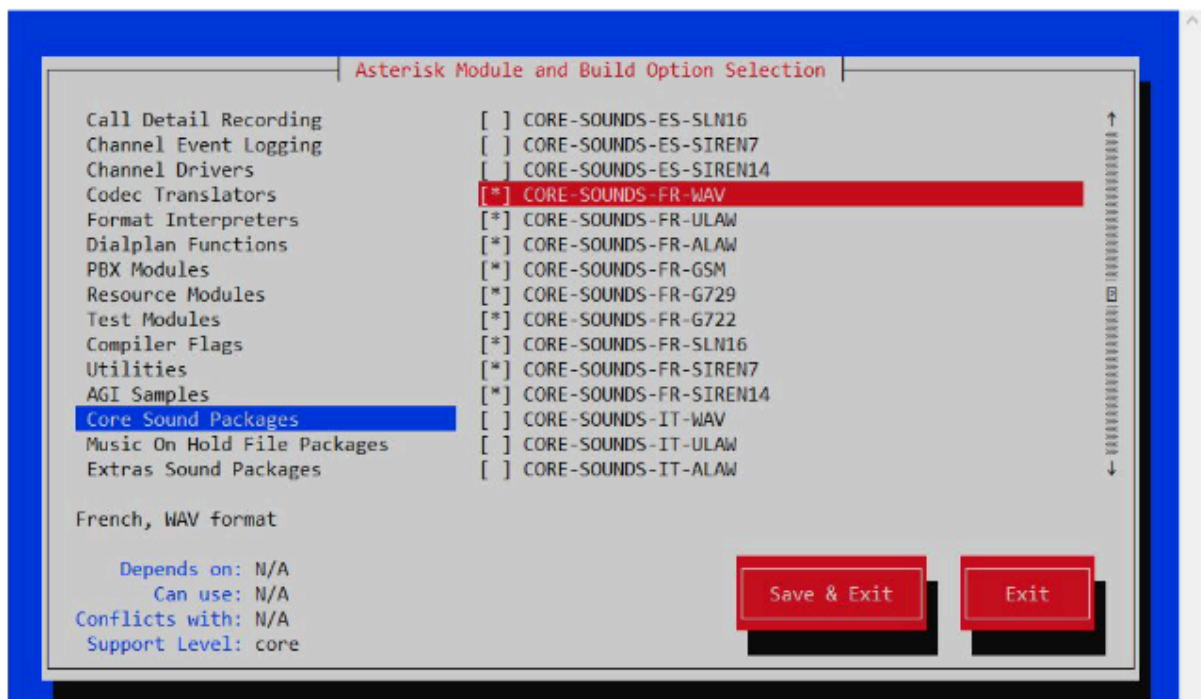
2. Installer Asterisk

***"sudo apt install build-essential libxml2-dev libncurses5-dev
 linux-headers-\$(uname -r) libsqlite3-dev libssl-dev libedit-dev uuid-dev
 libjansson-dev"***

3. Configuration de Asterisk

Après l'installation d'Asterisk on va le décompresser on utilisons la commande ***"tar -xvzf asterisk-20-current.tar.gz"***

On va exécuter la commande ***"./configure"*** qui effectuera plusieurs vérifications sur le système et préparera Asterisk à être installé sur le serveur.
"make menuselect" pour ajouter les packs français ou autre



- **make** (pour compiler le code source d'Asterisk)
- **make install**(installe les fichiers binaires)
- **make samples**(copie les fichiers de configuration d'exemple d'Asterisk)
- **make config**
- **systemctl start asterisk**
- **sudo asterisk -rv**
- **exit**
- **cd/etc/asterisk**

• Création et configuration des compte SIP

- On commence par ouvrir le fichier **"pjsip.conf"** et insérer les informations de configuration spécifiques à chaque compte SIP dans le fichier

```

GNU nano 2.2.6 pjsip.conf
transport=udp
type=transport
protocol=udp
bind=0.0.0.0:5060

[1000]
type=endpoint
context=fon-internal
disallow=all
allow=ulaw
auth=1000
aors=1000

[1000]
type=aor
max_contacts=1
remove_existing=yes

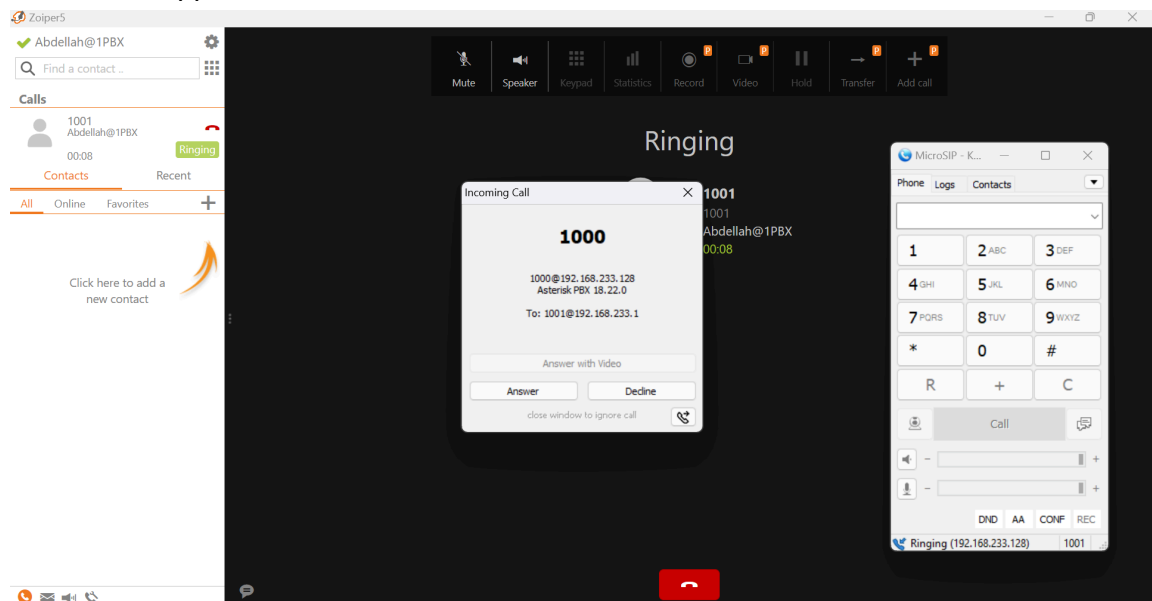
[1000]
type=auth
auth_type=userpass
username=1000
password=1000

[1001]
type=endpoint
context=fon-internal
disallow=all
allow=ulaw
auth=1001
aors=1001

```

Test et validation de la configuration

Ici j'ai exécuter un appel entre les deux utilisateurs



Sécurisation des appels

1. Création d'un dossier keys dans le répertoire **/etc/asterisk** où vont être logés nos clés : **mkdir /etc/asterisk/keys**
2. Ensuite, utilisation du script **"ast_tls_certe"** dans le répertoire source d'asterisk **"contrib/scripts"** qui va permettre de créer une autorité de certification auto-signée et un certificat Asterisk

"/ast_tls_cert -C 172.18.128.1 -O "Server VoIP" -d /etc/asterisk/keys -b 2048"

```

root@voipserver:/usr/src/asterisk-20.7.0/contrib/scripts# ./ast_tls_certificate -C 172.18.128.1 -O "Server VoIP" -d /etc/asterisk/keys -b 2048

No config file specified, creating '/etc/asterisk/keys/tmp.cfg'
You can use this config file to create additional certs without
re-entering the information for the fields in the certificate
Creating certificate /etc/asterisk/keys/asterisk.key
Creating signing request /etc/asterisk/keys/asterisk.csr
Creating certificate /etc/asterisk/keys/asterisk.crt
Certificate request self-signature ok
subject=CN = 172.18.128.1, O = Server VoIP
Enter pass phrase for /etc/asterisk/keys/ca.key:
Combining key and crt into /etc/asterisk/keys/asterisk.pem
root@voipserver:/usr/src/asterisk-20.7.0/contrib/scripts#

```

L'option **"-C"** est utilisée pour définir notre hôte - nom **DNS** ou **l'adresse IP**.

L'option **"-O"** définit notre nom d'entité.

L'option **"-d"** est le répertoire de sortie des clés.

L'option **"-b"** spécifie la taille du fichier de clé privée, par défaut 1024.

Ensuite on sera invité à saisir un mot de passe pour :

"/etc/asterisk/keys/ca.key", Cela créera le fichier **"ca.crt"**

De même pour **"/etc/asterisk/keys/asterisk.key"** + **"/etc/asterisk/keys/asterisk.crt"** seront automatiquement générés.

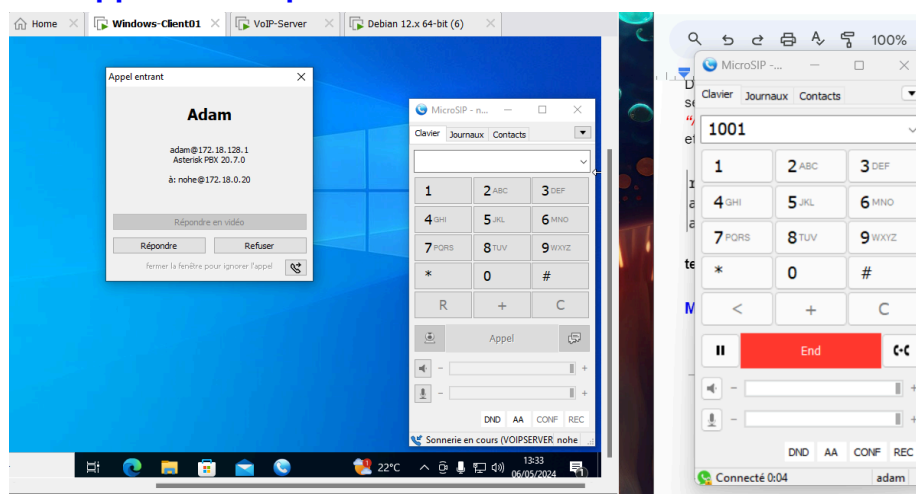
"/etc/asterisk/keys/asterisk.pem" sera créé, une combinaison des fichiers **"asterisk.key"** et **"asterisk.crt"**

```

root@voipserver:/etc/asterisk/keys# ls
asterisk.crt  asterisk.key  ca.cfg  ca.key
asterisk.csr  asterisk.pem  ca.crt  tmp.cfg

```

Test appel avec le protocole TLS



Test Validé

Mise en place du contexte des appels et de la messagerie vocal

- cd ***"/etc/asterisk"***
- nano ***"extensions.conf"***

[cowork]

```
exten => 1000,1,Answer()  
exten => 1000,2,Dial(PJSIP/adam,10) ; Après 10 secondes envoi vers la>  
exten => 1000,3,Playback(vm-nobodyavail)  
exten => 1000,4,VoiceMail(1000) ;répondeur compte  
exten => 1000,5,Hangup()  
exten => 1010,1,VoiceMailMain()      ; 1010 Numéro de téléphone du rép>  
  
exten => 1001,1,Answer()  
exten => 1001,2,Dial(PJSIP/nohe,10) ; Après 10 secondes envoi vers la>  
exten => 1001,3,Playback(vm-nobodyavail)  
exten => 1001,4,VoiceMail(1001) ; répondeur compte  
exten => 1001,5,Hangup()
```

- nano ***"voicemail.conf"***

[VoiceMail]

```
1000 => 1000, adam  
1001 => 1001, nohe
```

1000,1001 = mot de passe d'accès à la messagerie, nom d'utilisateur

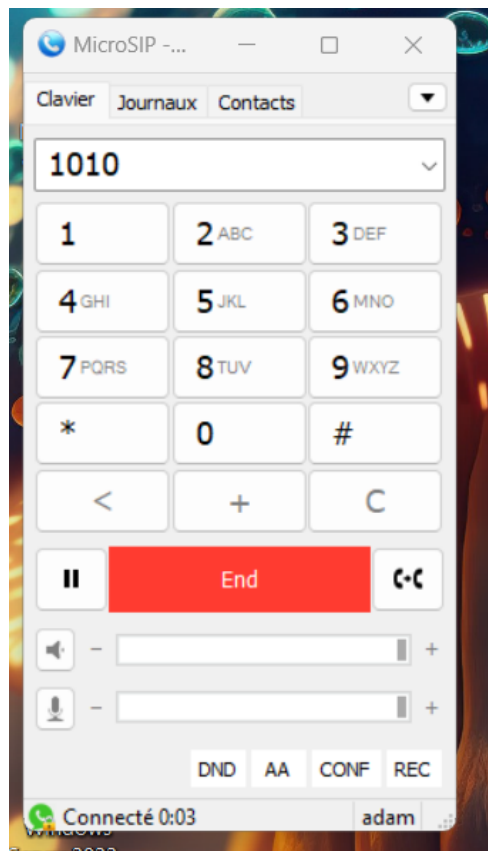
- ***"sudo asterisk -rx "core restart now""*** (Charger la configuration)
- sudo asterisk -rv
- ***"pjsip list aors"*** (lister les clients enregistrés)

```
===  
Connected to Asterisk 20.7.0 currently running on voipserver (pid = 15  
947)  
voipserver*CLI> pjsip list aors  
  
Aor: <Aor.....> <MaxC  
ontact>  
=====
```

Aor: adam	1
Aor: nohe	1

```
Objects found: 2
```


Test appeler la voicemail



Test

Test Validé

Fixer une adresse IP au serveur

Pour connaître l'interface réseau, utilisez la commande **"ip a"**, ici c'est **"ens33"**

```
root@voipserver:/etc/asterisk# 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 noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:6b:13:5b brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.18.128.1/16 brd 172.18.128.255 scope global ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe6b:135b/64 scope link
        valid_lft forever preferred_lft forever
```

Utilisez **"ip route show"** pour afficher la passerelle par défaut actuellement utilisée par notre machine.

Pour connaître notre nameserver, utilisez **"cat /etc/resolv.conf"**.

Ensuite, accédez à **"/etc/network/interfaces"** et appliquez la configuration suivante:

- Remplacez **"inet dhcp"** par **"static"**.
- Attribuez-lui l'adresse ip que vous avez choisie, ainsi que son masque réseau et enfin la passerelle

Pour ajout d'un second adaptateur :

1. sudo **"apt-get install net-tools"**

```
#Adresse ip static
allow-hotplug ens33
iface ens33 inet static
address 172.18.128.1
netmask 255.255.0.0
gateway 172.18.0.2
dns-nameservers 172.18.0.2
broadcast 172.18.128.255

#Bridge carte réseau
allow-hotplug ens37
iface ens37 inet dhcp

root@voipserver:/etc/asterisk# cat /etc/resolv.conf
# Generated by NetworkManager
search localdomain
nameserver 172.18.0.2
nameserver 172.18.0.15
```

5 - MISE EN PLACE D'UN AUTOMATE

Pour mettre en place notre automate, nous aurons besoin d'un système appelé **IVR**.

Qu'est-ce qu'un IVR ?

Un **IVR (Interactive Voice Response)** guide les utilisateurs vers le service qu'ils veulent contacter grâce à des options de menu prédéfinies.

En choisissant la bonne option, les appelants sont dirigés vers le département, l'individu ou la file d'attente correspondant.

À l'intérieur d'un **IVR**, vous pouvez configurer des règles spéciales pour déterminer qui prend chaque appel en fonction des compétences, du volume d'appels ou de la période de temps.

La mise en place d'un **IVR** se fait en deux étapes :

1er on fait parler Asterisk.

2ème création d'une **IVR**.

Comment faire parler Asterisk ?

Pour faire parler Asterisk, on installe un outil de synthèse vocale appelé **GoogleTTS**.

Installation de GoogleTTS

sudo apt-get install perl libwww-perl sox mpg123

```
root@voipserver:/home/voip# sudo apt-get install perl libwww-perl sox mpg123
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
perl est déjà la version la plus récente (5.36.0-7+deb12u1).
libwww-perl est déjà la version la plus récente (6.68-1).
sox est déjà la version la plus récente (14.4.2+git20190427-3.5).
mpg123 est déjà la version la plus récente (1.31.2-1).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 8 non mis à jour.
root@voipserver:/home/voip#
```

wget https://raw.githubusercontent.com/zaf/asterisk-googlelets/master/googlelets.agi

```
Résolution de raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.109.133, 185.199.110.133, 185.199.111.133, ...
Connexion à raw.githubusercontent.com (raw.githubusercontent.com)|185.199.109.133|:443... connecté.
requête HTTP transmise, en attente de la réponse... 301 Moved Permanently
Emplacement : https://raw.githubusercontent.com/zaf/asterisk-googlelets/master/googlelets.agi [suivant]
--2024-05-06 01:23:45-- https://raw.githubusercontent.com/zaf/asterisk-googlelets/master/googlelets.agi
Résolution de raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.109.133, 185.199.110.133, 185.199.111.133, ...
Connexion à raw.githubusercontent.com (raw.githubusercontent.com)|185.199.109.133|:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 10224 (10,0K) [text/plain]
Sauvegarde en : « googlelets.agi »

googlelets.agi 100%[=====] 9,98K --.-KB/s ds 0s

2024-05-06 01:23:45 (47,3 MB/s) - « googlelets.agi » sauvegardé [10224/10224]
root@voipserver:/var/lib/asterisk/agi-bin#
```

chmod +x googlelets.agi (pour accorder les droits)

Maintenant, GoogleTTS est installé. Voyons comment l'utiliser.

Configuration des IVR

La configuration des **IVR** se fait dans le fichier **extensions.conf**

1. Accéder au fichier : **cd /etc/asterisk** puis ouvrir **extensions.conf** avec l'éditeur de texte nano.
2. Ajouter le contexte souhaité (par exemple, **"menu"**)

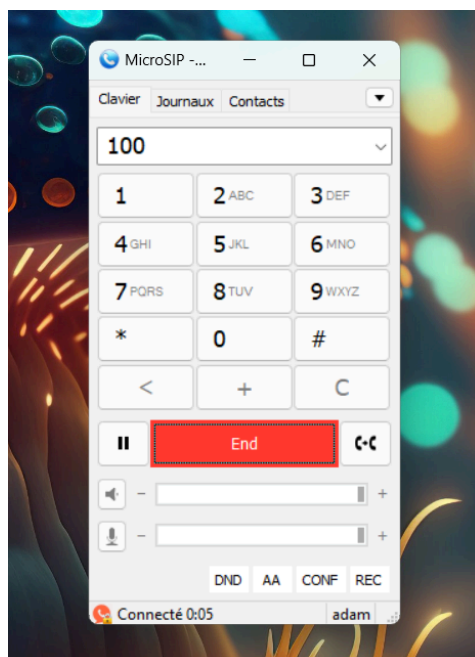
exten => **100**,1,Goto(menu,s,1) ; **100** = numéro pour accéder au menu (vous pouvez en choisir un autre)

Exemple de menu

```
GNU nano 7.2 extensions.conf *
exten => 100,1,Goto(menu,s,1);100 = numéro pour accéder au menu
[menu]
exten => s,1,Answer()
same => n,Set(TIMEOUT(response)=10)
same => n,agi(googletts.agi,"Bienvenue sur le service AIS",fr,any)
same => n,Wait(2)
same => n,agi(googletts.agi,"Pour obtenir de l'aide en cas d'urgence, appuyer sur 1.")
same => n,Wait(2)
same => n,agi(googletts.agi,"Pour parler à un expert, appuyer sur 2.",fr,any)
same => n,Wait(2)
same => n,agi(googletts.agi,"Pour des informations sur nos services et produits, appu")
same => n,Wait(2)
same => n,agi(googletts.agi,"Pour toutes autres questions, appuyer sur 4.",fr,any)
same => n,WaitExten(10) ;
same => n,Goto(${EXTEN},1) ;
same => n,Goto(menu,s,1) ;
exten => 1,1,Goto(cowork,1000,1)
exten => 2,1,Goto(cowork,1001,1)
exten => _[5-9#],1,Goto(menu,s,1)
exten => t,1,Goto(menu,s,1)
```

Restart la configuration et tester votre IVR

"sudo asterisk -rx "core restart now""



Mise en place d'une solution d'appels automatisés pour la prospection

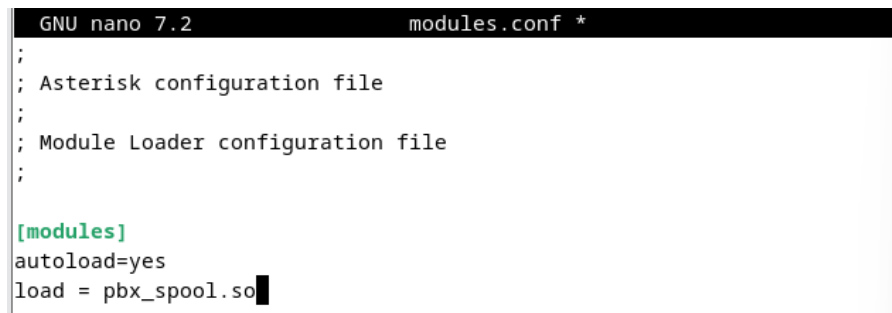
Premièrement avant de mettre en place notre solution on va d'abord vérifier une chose, dans le dossier **etc/asterisk**.

On vérifie dans le fichier **modules.conf** pour voir si un truc est bien activé **autoload=yes**.

Ensuite, on s'assure que le fichier **pbx_spool.so** est bien chargé

load = pbx_spool.so. Si ça n'est pas le cas, ça ne marchera pas.

```
GNU nano 7.2          modules.conf *
```

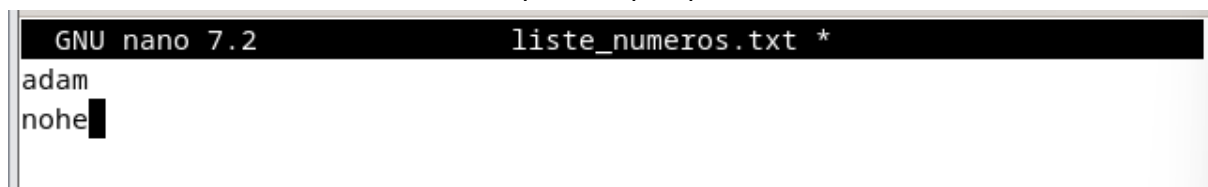


```
;
; Asterisk configuration file
;
; Module Loader configuration file
;
[modules]
autoload=yes
load = pbx_spool.so
.
```

Pour créer notre liste de contacts, on va utiliser un format simple, un fichier txt qu'on va appeler **liste_numeros.txt**

On va écrire les numéros des contacts qu'on va prospector à l'intérieur.

```
GNU nano 7.2          liste_numeros.txt *
```



```
adam
nohe
```

Dans Asterisk, le dossier spécial **/var/spool/asterisk/outgoing/** joue un rôle crucial dans la gestion des appels sortants. Lorsque vous déposez un fichier dans ce répertoire,

Asterisk le détecte automatiquement et l'utilise pour passer des appels. C'est dans ce répertoire que nous allons placer nos contacts pour automatiser les appels sortants.

Pour configurer cette fonctionnalité, nous allons utiliser un script Python. Ce script sera responsable de préparer les fichiers de contact appropriés et de les placer dans le répertoire **/var/spool/asterisk/outgoing/**

Une fois les fichiers déposés, Asterisk les traitera automatiquement pour passer les appels correspondants.

script.py

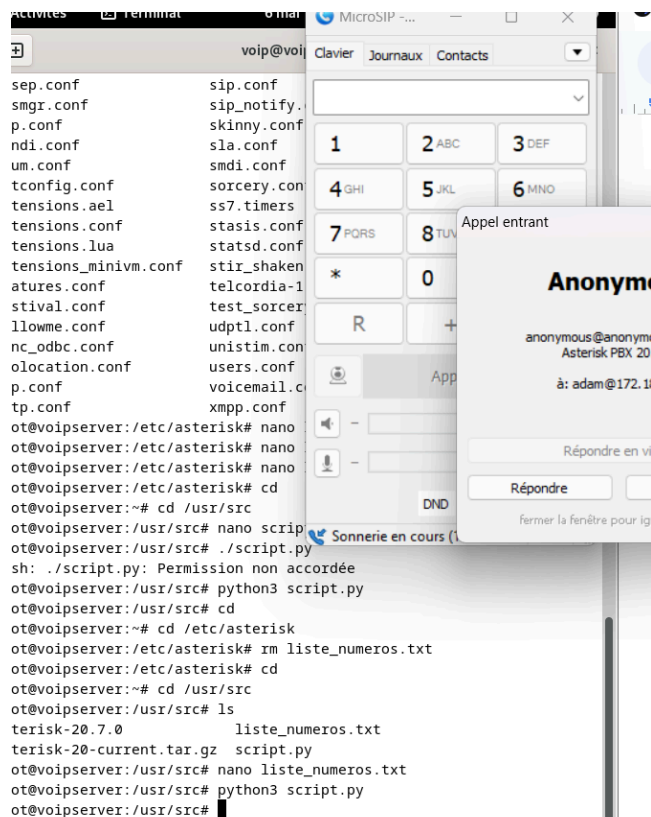
```
GNU nano 7.2 script.py *
import random
import os

# Lire la liste de numéros depuis le fichier texte
with open('liste_numeros.txt', 'r') as file:
    contacts = file.read().splitlines()

# Randomiser la liste des contacts
random.shuffle(contacts)

# Chemin du répertoire sortant d'Asterisk
outgoing_dir = '/var/spool/asterisk/outgoing/'

# Créer des fichiers d'appel Asterisk pour chaque contact
for i, contact in enumerate(contacts, start=1):
    filename = os.path.join(outgoing_dir, f'appel_{i}.call')
    with open(filename, 'w') as call_file:
        call_file.write(f'Channel: PJSIP/{contact}\n')
        call_file.write('MaxRetries: 3\n')
        call_file.write('RetryTime: 60\n')
        call_file.write('WaitTime: 30\n')
        call_file.write('Context: prospection\n')
        call_file.write('Extension: s\n')
        call_file.write('Priority: 1\n')
```



Test Validé

Fichier généré à la sortie par notre script

```
root@voipserver:~# cd /var/spool/asterisk/outgoing
root@voipserver:/var/spool/asterisk/outgoing# ls
appel_1.call  appel_2.call
```

Contenu d'un fichier

```
GNU nano 7.2
Channel: PJSIP/allaoui
MaxRetries: 3
RetryTime: 60
WaitTime: 30
Context: prospection
Extension: s
Priority: 1
```

extensions.conf

```
[prospection]
exten => s,1,NoOp(Début de la prospection)
same => n,AGI(googletts.agi,"Bienvenue chez Cybersecu Expert. Nous s>
same => n,AGI(googletts.agi,"Avec notre offre de base, nous offrons >
same => n,AGI(googletts.agi,"En plus de notre offre de base, nous pr>
same => n,Hangup()
```

Channel = Ce paramètre spécifie le canal sur lequel Asterisk doit placer l'appel sortant.

MaxRetries = Ce paramètre spécifie le nombre maximum de tentatives d'appel en cas d'échec. Si l'appel initial échoue, Asterisk tentera de réessayer jusqu'à MaxRetries fois avant d'abandonner.

Par exemple, **MaxRetries: 3**, Asterisk tentera l'appel jusqu'à trois fois au total.

RetryTime = Ce paramètre spécifie le délai (en secondes) entre chaque tentative d'appel en cas d'échec. Après un échec d'appel, Asterisk attendra le nombre de secondes spécifié dans RetryTime avant de retenter l'appel.

Par exemple, **RetryTime: 60**, Asterisk attendra 60 secondes avant de retenter l'appel après un échec.

WaitTime = Ce paramètre spécifie le temps maximum (en secondes) à attendre pour que l'appel soit connecté avant de passer à l'étape suivante ou d'abandonner. Si l'appel ne se connecte pas dans le délai spécifié par WaitTime, Asterisk considérera l'appel comme un échec.

Par exemple, ici **WaitTime: 30**, Asterisk attendra jusqu'à 30 secondes pour que l'appel soit connecté.

Context = Ce paramètre spécifie le contexte dans lequel les appels sortants doivent être traités.

Le contexte détermine **les règles de routage** et **les actions à prendre** pour les appels sortants.

Extension = Ce paramètre spécifie l'extension dans le contexte spécifié à laquelle Asterisk doit envoyer l'appel sortant.

Priority = Ce paramètre spécifie la priorité dans le contexte et l'extension spécifiés à laquelle Asterisk doit envoyer l'appel sortant.

Si plusieurs actions sont définies pour une extension donnée, la priorité détermine l'ordre dans lequel elles doivent être exécutées.

6 - Déploiement automatisé

Installation d'une nouvelle machine virtuelle, comme vu précédemment.

Dans le terminal de la nouvelle machine :

1. Créez un fichier nommé selon votre préférence, par exemple, utilisez la commande **"nano script_install_asterisk.sh"**, puis ajouter le script


```

GNU nano 7.2      script_install_asterisk.sh
#!/bin/bash

# Mettre à jour les paquets système
apt update && sudo apt upgrade -y

#On install sudo
apt-get install sudo -y

# Télécharger et installer Asterisk
cd /usr/src
wget https://downloads.asterisk.org/pub/telephony/asterisk/asterisk-2>
tar xvf asterisk-20-current.tar.gz
cd asterisk-20.7.0/

# Configurer Asterisk
./contrib/scripts/install_prereq install
./configure
make
make install
make samples && make config

sudo systemctl start asterisk

# Configuration de fichier contact
cd /etc/asterisk

# Créer le contact PJSIP
mv pjsip.conf pjsip.conf.old
touch pjsip.conf

```

```

GNU nano 7.2      script_install_asterisk.sh

echo "[endpoint]" >> /etc/asterisk/pjsip.conf
echo "type=endpoint" >> /etc/asterisk/pjsip.conf
echo "context=cowork" >> /etc/asterisk/pjsip.conf
echo "disallow=all" >> /etc/asterisk/pjsip.conf
echo "allow=ulaw" >> /etc/asterisk/pjsip.conf
echo "allow=h263" >> /etc/asterisk/pjsip.conf
echo "language=fr" >> /etc/asterisk/pjsip.conf
echo "" >> /etc/asterisk/pjsip.conf

echo "[auth_userpass]" >> /etc/asterisk/pjsip.conf
echo "type=auth" >> /etc/asterisk/pjsip.conf
echo "auth_type=userpass" >> /etc/asterisk/pjsip.conf
echo "" >> /etc/asterisk/pjsip.conf

echo "[aor_dynamic]" >> /etc/asterisk/pjsip.conf
echo "type=aor" >> /etc/asterisk/pjsip.conf
echo "max_contacts=3" >> /etc/asterisk/pjsip.conf
echo "" >> /etc/asterisk/pjsip.conf

echo "[admin](endpoint)" >> /etc/asterisk/pjsip.conf
echo "auth=admin" >> /etc/asterisk/pjsip.conf
echo "aors=admin" >> /etc/asterisk/pjsip.conf
echo "" >> /etc/asterisk/pjsip.conf

echo "[admin](auth_userpass)" >> /etc/asterisk/pjsip.conf
echo "password=1234" >> /etc/asterisk/pjsip.conf
echo "username=admin" >> /etc/asterisk/pjsip.conf
echo "" >> /etc/asterisk/pjsip.conf
echo "[admin](aor_dynamic)" >> /etc/asterisk/pjsip.conf

echo "[client](endpoint)" >> /etc/asterisk/pjsip.conf
echo "auth=client" >> /etc/asterisk/pjsip.conf

```

2. Donner les autorisations du fichier afin qu'il puisse être exécuté avec la commande **"chmod +x script_install_asterisk.sh"**
3. Exécuter le script en exécutant la commande **"/script_install_asterisk.sh"**

Petit aperçu de la fin de la configuration

```
Installing file configs/samples/statsd.conf.sample
Installing file configs/samples/stir_shaken.conf.sample
Installing file configs/samples/test_sorcery.conf.sample
Installing file configs/samples/udptl.conf.sample
Installing file configs/samples/unistim.conf.sample
Installing file configs/samples/users.conf.sample
Installing file configs/samples/voicemail.conf.sample
Installing file configs/samples/xmpp.conf.sample
Updating asterisk.conf
/usr/bin/install -c -d "/var/spool/asterisk/voicemail/default/1234/INBOX"
build_tools/make_sample_voicemail "//var/lib/asterisk" "//var/spool/asterisk"
Installing file phoneprov/000000000000.cfg
Installing file phoneprov/000000000000-directory.xml
Installing file phoneprov/000000000000-phone.cfg
Installing file phoneprov/polycom_line.xml
Installing file phoneprov/polycom.xml
Installing file phoneprov/snom-mac.xml
/etc/asterisk/pjsip.conf
Unable to connect to remote asterisk (does /var/run/asterisk/asterisk.ctl exist?)
• asterisk.service - LSB: Asterisk PBX
  Loaded: loaded (/etc/init.d/asterisk; generated)
  Active: active (running) since Mon 2024-05-06 10:44:12 CEST; 119s
  Docs: man:systemd-sysv-generator(8)
  Process: 39021 ExecStart=/etc/init.d/asterisk start (code=exited, status=0/SUCCESS)
  Tasks: 1 (limit: 2252)
  Memory: 15.2M
  CPU: 194ms
  CGroup: /system.slice/asterisk.service
          └─39034 /usr/sbin/asterisk

mai 06 10:44:12 Asterisk-Script systemd[1]: Starting asterisk.service>
mai 06 10:44:12 Asterisk-Script asterisk[39021]: Starting Asterisk PB>
mai 06 10:44:12 Asterisk-Script systemd[1]: Started asterisk.service >
lines 1-14/14 (END)
```

Test Validé

8 – SÉCURISATION

Nous allons installer un système de supervision pour surveiller notre serveur, et pour cela, nous utiliserons **Zabbix**.

Zabbix est une solution logicielle qui permet de surveiller une multitude de paramètres réseau ainsi que la santé et l'intégrité des serveurs, des machines virtuelles, des applications, des services, des bases de données, des sites web, et même des environnements cloud.

Grâce à son mécanisme de notification flexible, les utilisateurs peuvent configurer des alertes par e-mail pour pratiquement tous les événements, assurant ainsi une réactivité rapide aux problèmes serveurs.

De plus, Zabbix propose des fonctionnalités avancées de reporting et de visualisation des données, basées sur les données collectées, ce qui en fait un outil idéal pour la gestion de la capacité.

Pour ce faire nous allons préparer une nouvelle machine virtuelle sur laquelle nous allons déployer notre solution.

On utilise la machine créer précédemment qui contenait le script qui automatisé l'installation et configuration asterisk, nous pouvons commencer.

```
Debian GNU/Linux 12 zabbix-server tty1
zabbix-server login: zabbix
Password:
Linux zabbix-server 6.1.0-20-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.85-1 (2024-04-11) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat May  4 20:40:55 EDT 2024 from 192.168.1.1 on pts/0
zabbix@zabbix-server:~$ _
```

Préparation de l'environnement

Avant de commencer l'installation de Zabbix, on va avoir besoin d'un système de base de données ici nous utiliserons la base de données MariaDB.

Mais de commencer l'installation nous allons mettre à jour votre système d'exploitation et ensuite installer la base de données :

1. ***sudo apt update***
2. ***sudo apt upgrade***
3. ***sudo apt install mariadb-server***

Sécuriser l'installation MariaDB

1. ***sudo mariadb-secure-installation***

Ensuite vous suivez les instructions et définissez un mot de passe (ce mot de passe nous l'utiliserons plus tard)

Installation de Zabbix

Pour l'installation suivez les instructions du site officiels [ici](#)

Si à l'étape de l'installation vous avez cette erreur :

```
root@Asterisk-Script:/home/adam# dpkg -i zabbix-release_6.4-1+debian12_all.deb
dpkg: avertissement: « ldconfig » introuvable dans la variable PATH ou non exécutable
dpkg: avertissement: « start-stop-daemon » introuvable dans la variable PATH ou non exécutable
dpkg: erreur: 2 programmes attendus non trouvés dans la variable PATH ou non exécutables
Note : la variable PATH du superutilisateur doit normalement contenir /usr/local/sbin, /usr/sbin et /sbin
```

Pour résoudre le problème :

1. créer un fichier avec touch et nomme le comme arbitrairement (**touch bashrc**)
2. ensuite edit le fichier avec nano (**nano bashrc**)
3. Ajouter cette ligne :

```
GNU nano 7.2 bashrc *
export PATH=$PATH:/usr/local/sbin:/usr/sbin:/sbin
```

4. Ensuite enregistrer et fermer le fichier
5. changer le nom du fichier en **"~/.bashrc"**
mv bashrc ~/.bashrc
6. **chmod +x ~/.bashrc**
7. source **~/.bashrc**

```
root@Asterisk-Script:/home/adam# mv bashrc ~/.bashrc
root@Asterisk-Script:/home/adam# chmod +x ~/.bashrc
root@Asterisk-Script:/home/adam# ls
Bureau      script_install_asterisk.sh
Documents   server.csr
Images      server.key
Modèles     Téléchargements
Musique     Vidéos
Public      zabbix-release_6.4-1+debian12_all.deb
root@Asterisk-Script:/home/adam# source ~/.bashrc
root@Asterisk-Script:/home/adam# █
```

8. Maintenant relance l'installation et cette fois-ci tout devrait bien se passer

```
root@Asterisk-Script:/home/adam# dpkg -i zabbix-release_6.4-1+debian12_all.deb
Sélection du paquet zabbix-release précédemment désélectionné.
(Lecture de la base de données... 188562 fichiers et répertoires déjà
installés.)
Préparation du dépaquetage de zabbix-release_6.4-1+debian12_all.deb ..
.
Dépaquetage de zabbix-release (1:6.4-1+debian12) ...
Paramétrage de zabbix-release (1:6.4-1+debian12) ...
root@Asterisk-Script:/home/adam#
```

Allons sur l'interface web

Une fois l'installation terminée, nous allons accéder à notre tableau de bord et le configurer pour notre première utilisation.

Pour y accéder : **<http://172.18.0.10/zabbix>** , remplacer (**172.18.0.10** par celui de votre serveur zabbix)

7 - SÉCURISATION *Prometheus/Grafana*

Mise en place du serveur Prometheus/Grafana

Prometheus est un logiciel libre de **surveillance** informatique et générateur d'alertes. Il enregistre des **métriques** en temps réel dans une base de données (avec une capacité d'acquisition élevée) en se basant sur le contenu de point d'entrée exposé à l'aide du protocole **Http**. Ces **métriques** peuvent ensuite être interrogées à l'aide d'un langage de requête simple : le **PromQL** et peuvent également servir à générer des **alertes**. Le projet est écrit en **Go** et est disponible sous licence **Apache 2**. Prometheus n'est pas conçu pour faire de la restitution d'informations sous la forme de tableau de bord bien qu'il dispose d'une solution pour le faire. Une bonne pratique est de faire appel à un outil comme **Grafana**. **Grafana** est un outil open source de monitoring informatique orienté data visualisation. Il est conçu pour générer des **dashboards** sur la base de **métriques** et données **temporelles**.

apt-get install prometheus

```
root@voipserver:~# apt-get install prometheus
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
prometheus est déjà la version la plus récente (2.42.0+ds-5+b5).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 8 non mis à jour.
1.
```

systemctl start prometheus
systemctl status prometheus

```

root@voipserver:~# systemctl status prometheus
● prometheus.service - Monitoring system and time series database
   Loaded: loaded (/lib/systemd/system/prometheus.service; enabled; >
   Active: active (running) since Mon 2024-05-06 03:29:42 CEST; 2mi
   Docs: https://prometheus.io/docs/introduction/overview/
        man:prometheus(1)
   Main PID: 5814 (prometheus)
     Tasks: 8 (limit: 4590)
    Memory: 28.5M
       CPU: 327ms
    CGroup: /system.slice/prometheus.service
           └─5814 /usr/bin/prometheus

mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
mai 06 03:29:42 voipserver prometheus[5814]: ts=2024-05-06T01:29:42.4>
lines 1-22/22 (END)

```

curl -s https://packages.grafana.com/gpg.key | apt-key add - >/dev/null

sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"

```

root@voipserver:~# sudo add-apt-repository "deb https://packages.grafa
na.com/oss/deb stable main"
Repository: 'deb https://packages.grafana.com/oss/deb stable main'
Description:
Archive for codename: stable components: main
More info: https://packages.grafana.com/oss/deb
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
Found existing deb entry in /etc/apt/sources.list.d/archive_uri-https_
packages_grafana_com_oss_deb-bookworm.list
Adding deb entry to /etc/apt/sources.list.d/archive_uri-https_packages
_grafana_com_oss_deb-bookworm.list
Found existing deb-src entry in /etc/apt/sources.list.d/archive_uri-ht
tps_packages_grafana_com_oss_deb-bookworm.list
Adding disabled deb-src entry to /etc/apt/sources.list.d/archive_uri-h
ttps_packages_grafana_com_oss_deb-bookworm.list
Atteint :1 http://security.debian.org/debian-security bookworm-securit
y InRelease
Atteint :2 http://deb.debian.org/debian bookworm InRelease
Atteint :3 http://deb.debian.org/debian bookworm-updates InRelease
Atteint :4 https://packages.grafana.com/oss/deb stable InRelease
Lecture des listes de paquets... Fait
W: https://packages.grafana.com/oss/deb/dists/stable/InRelease: Key is
stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the

```

apt-get -y install grafana

```
Les paquets supplémentaires suivants seront installés :
musl
Les NOUVEAUX paquets suivants seront installés :
  grafana musl
0 mis à jour, 2 nouvellement installés, 0 à enlever et 8 non mis à jour.
Il est nécessaire de prendre 115 Mo dans les archives.
Après cette opération, 424 Mo d'espace disque supplémentaires seront utilisés.
Réception de :1 http://deb.debian.org/debian bookworm/main amd64 musl
amd64 1.2.3-1 [406 kB]
Réception de :2 https://packages.grafana.com/oss/deb stable/main amd64
grafana amd64 10.4.2 [114 MB]
115 Mo réceptionnés en 16s (7 242 ko/s)
Sélection du paquet musl:amd64 précédemment désélectionné.
(Lecture de la base de données... 175701 fichiers et répertoires déjà
installés.)
Préparation du dépaquetage de .../musl_1.2.3-1_amd64.deb ...
Dépaquetage de musl:amd64 (1.2.3-1) ...
Sélection du paquet grafana précédemment désélectionné.
Préparation du dépaquetage de .../grafana_10.4.2_amd64.deb ...
Dépaquetage de grafana (10.4.2) ...
Progression : [ 33%] [#####.....]
```

cd /etc/init.d/grafana
systemctl start grafana-server.service


```

root@voipserver:/etc/init.d# systemctl start grafana-server.service
root@voipserver:/etc/init.d# systemctl status grafana-server.service
● grafana-server.service - Grafana instance
   Loaded: loaded (/lib/systemd/system/grafana-server.service; disabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-05-06 03:44:54 CEST; 10s ago
     Docs: http://docs.grafana.org
    Main PID: 11157 (grafana)
      Tasks: 15 (limit: 4590)
    Memory: 47.7M
       CPU: 2.369s
    CGroup: /system.slice/grafana-server.service
            └─11157 /usr/share/grafana/bin/grafana server --config=/etc/grafana/grafana.ini

mai 06 03:44:56 voipserver grafana[11157]: logger=ngalert.multiorg.alerts
mai 06 03:44:56 voipserver grafana[11157]: logger=ngalert.scheduler tasks
mai 06 03:44:56 voipserver grafana[11157]: logger=ticker t=2024-05-06 03:44:56
mai 06 03:44:56 voipserver grafana[11157]: logger=http.server t=2024-05-06 03:44:56
mai 06 03:44:56 voipserver grafana[11157]: logger=provisioning.dashboard
mai 06 03:44:56 voipserver grafana[11157]: logger=provisioning.dashboard
mai 06 03:44:56 voipserver grafana[11157]: logger=plugins.update.check
mai 06 03:44:56 voipserver grafana[11157]: logger=grafana.update.check
mai 06 03:44:56 voipserver grafana[11157]: logger=grafana-apiserver tasks
mai 06 03:44:56 voipserver grafana[11157]: logger=grafana-apiserver tasks
lines 1-21/21 (END)

```

apt install mariadb-server

```

Paramétrage de gawk (1:5.2.1-2) ...
Paramétrage de libcgi-pm-perl (4.55-1) ...
Paramétrage de libfcgi-bin (2.4.2-2) ...
Paramétrage de libhtml-template-perl (2.97-2) ...
Paramétrage de mariadb-client-core (1:10.11.6-0+deb12u1) ...
Paramétrage de socat (1.7.4.4-2) ...
Paramétrage de libdaxctl1:amd64 (76.1-1) ...
Paramétrage de pv (1.6.20-1) ...
Paramétrage de libndctl6:amd64 (76.1-1) ...
Paramétrage de libfcgi-perl (0.82+ds-2) ...
Paramétrage de libterm-readkey-perl (2.38-2+b1) ...
Paramétrage de libpmem1:amd64 (1.12.1-2) ...
Paramétrage de libdbi-perl:amd64 (1.643-4) ...
Paramétrage de rsync (3.2.7-1) ...
rsync.service is a disabled or a static unit, not starting it.
Paramétrage de mariadb-client (1:10.11.6-0+deb12u1) ...
Paramétrage de libcgi-fast-perl (1:2.15-1) ...
Paramétrage de libdbd-mariadb-perl (1.22-1+b1) ...
Paramétrage de mariadb-server-core (1:10.11.6-0+deb12u1) ...
Paramétrage de mariadb-server (1:10.11.6-0+deb12u1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service → /lib/systemd/system/mariadb.service.

Progression : [ 89%] [#####.....]

```

wget

https://github.com/prometheus/node_exporter/releases/download/v1.7.0/node_exporter-1.7.0.linux-amd64.tar.gz

```

cation%2Foctet-stream [suivant]
--2024-05-06 03:58:06-- https://objects.githubusercontent.com/github-
production-release-asset-2e65be/9524057/01323270-6ecb-47aa-813f-52a2a8
9cdc64?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAVCODYLSA5
3PQK4ZA%2F20240506%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20240506
T015635Z&X-Amz-Expires=300&X-Amz-Signature=043faa8480d1e350152c94d23c9
5b0dc48a798ad58f57bec37c6cca35d197562&X-Amz-SignedHeaders=host&actor_i
d=0&key_id=0&repo_id=9524057&response-content-disposition=attachment%3
B%20filename%3Dnode_exporter-1.7.0.linux-amd64.tar.gz&response-content
-type=application%2Foctet-stream
Résolution de objects.githubusercontent.com (objects.githubusercontent
.com)... 185.199.109.133, 185.199.110.133, 185.199.111.133, ...
Connexion à objects.githubusercontent.com (objects.githubusercontent.c
om)|185.199.109.133|:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 10419253 (9,9M) [application/octet-stream]
Sauvegarde en : « node_exporter-1.7.0.linux-amd64.tar.gz »

node_exporter-1.7 100%[=====>] 9,94M 13,1MB/s ds 0,8s

2024-05-06 03:58:07 (13,1 MB/s) - « node_exporter-1.7.0.linux-amd64.ta
r.gz » sauvegardé [10419253/10419253]

root@voipserver:/etc/prometheus#

```

tar -xzf node_exporter-1.7.0.linux-amd64.tar.gz

nano /etc/prometheus/prometheus.yml

```

GNU nano 7.2 /etc/prometheus/prometheus.yml
# Sample config for Prometheus.

global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds.
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The default
  # scrape_timeout is set to the global default (10s).

  # Attach these labels to any time series or alerts when communicating with
  # external systems (federation, remote storage, Alertmanager).
  external_labels:
    monitor: 'example'

# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
      - targets: ['localhost:9093']

# Load rules once and periodically evaluate them according to the global
rule_files:

```

systemctl restart prometheus

aller sur les serveur distant et installer le node exporter :

apt-get install prometheus-node-exporter puis basculer sur le serveur prometheus : tester la connexion telnet : **telnet 172.18.128.1 9100**

[illegible]