

<https://devopssec.fr/>

Documentation technique

AJDAINI Hatim

11/07/2019

TABLE DES MATIERES

Introduction.....	2
Installation du serveur Issabel.....	3
Configuration de base	6
Création des extensions	6
Ring Groups - Groupement d'appels.....	7
Follow Me - Redirection d'appels.....	8
Messagerie vocale.....	9
Passage du site en HTTPS	10
Mise en place du VPN.....	11
Côté serveur	11
Côté client	14
Configuration du TLS sur le serveur et mise en place du SRTP	16
Pourquoi est-il important de sécuriser les communications passant par la VoIP ?	16
Générer les certificats nécessaires au chiffrement des appels.....	18
Serveur	18
Client (cas de blink uniquement).....	20
Configuration des clients Softphones	21
PC Windows : MicroSip	21
PC : Blink.....	22
Configuration de GrandStream Wave sur iPhone (identique à Android)	25
Vérification du chiffrement	27
Contraintes, limites et recommandations :	28
Sources et sitographie	30

INTRODUCTION

Dans le cadre d'un workshop d'une durée de deux semaines à l'école l'IMIE mélangeant les filières et les niveaux d'étude, il nous a été demandé de répondre au besoin d'une entreprise nantaise spécialisée dans les domaines du réseau et de la téléphonie (c'est une entreprise réelle, mais par souci de confidentialité, nous ne pouvons pas divulguer sa vraie identité) qui souhaite mettre en place de la convergence entre les téléphones fixes et mobiles de l'entreprise.

Le projet réside en l'utilisation limitée de la téléphonie en situation de mobilité. La solution mise en place devra être sécurisée entre un téléphone mobile et un téléphone fixe (IPBX).

La solution conseillée par le client est un serveur IPBX basé sur Asterisk, Issabel. Elle doit être utilisable depuis les postes utilisateurs et les téléphones Android et iOS. Elle fonctionnera avec la 4G.

INSTALLATION DU SERVEUR ISSABEL

L'installation s'est faite sur une machine CentOS en net install. Nous avons utilisé le serveur mis à disposition par l'entreprise accessible en SSH.

Nous avons donc accès à un serveur CentOS :

```
user@issabel -> neofetch

      .:.
    .PLTJ.
      <><><>
KKSSV' 4KKK LJ KKKL.'VSSKK
KKV' 4KKKKK LJ KKKKAL 'VKK
V' ' 'VKKKK LJ KKKKV' ' 'V
.4MA.' 'VKK LJ KKV' '.4Mb.
. KKKKKA.' 'V LJ V' '.4KKKKK .
.4D KKKKKKKA.' LJ '''.4KKKKKKK FA.
<QDD ++++++++ ++++++++ GFD>
'VD KKKKKKKK'.. LJ ..'KKKKKKKK FV
' VKKKKK'..4 LJ K. 'KKKKKV '
'VK'..4KK LJ KKA. 'KV'
A. .4KKKK LJ KKKKA. .4
KKA. 'KKKKK LJ KKKKK' .4KK
KKSSA. VKKK LJ KKKV .4SSKK
      <><><>
    'MKKM'
      .:.

user@:
-----
OS: CentOS Linux 7 (Core) x86_64
Host: SYS-5038MD-H24TRF-0S012 0123456789
Kernel: 3.10.0-957.21.3.el7.x86_64
Uptime: 2 days, 1 hour, 45 mins
Packages: 882 (rpm)
Shell: fish /usr/bin/neofetch: line 1472: =bi
Terminal: /dev/pts/2
CPU: Intel Xeon D-1531 (12) @ 2.700GHz
GPU: 06:00.0 ASPEED Technology, Inc. ASPEED Gra
Memory: 895MiB / 31910MiB
```

Sur ce serveur, nous allons installer Issabel, qui est un fork de Elastix. C'est une version libre d'un IPBX configurable directement via interface web et utilisant actuellement Asterisk en version 13. Pour ce faire, nous avons récupéré et lancé un script d'installation en tant que root :

```
steven@pc-420:~/Documents/uxperiment$ scp ./issabel4-asterisk13-netinstall.sh user@ :/home/user/
user@ :s password:
issabel4-asterisk13-netinstall.sh 100% 17KB 360.8KB/s 00:00
```

Le script et les images d'Issabel sont disponibles ici :

<https://sourceforge.net/projects/issabelpbx/files/Issabel%204/>.

```
[root@sd-124518 user]# ./issabel4-asterisk13-netinstall.sh
setenforce: SELinux is disabled
Adding new user asterisk...
Modules complémentaires chargés : fastestmirror, langpacks
Determining fastest mirrors
 * base: centos.quelquesmots.fr
 * extras: centos.quelquesmots.fr
 * updates: centos.crazyfrogs.org
base | 3.6 kB 00:00:00
extras | 3.4 kB 00:00:00
updates | 3.4 kB 00:00:00
(1/4): base/7/x86_64/group_gz | 166 kB 00:00:00
(2/4): extras/7/x86_64/primary_db | 205 kB 00:00:00
(3/4): updates/7/x86_64/primary_db | 6.5 MB 00:00:00
(4/4): base/7/x86_64/primary_db | 6.0 MB 00:00:01
No packages marked for update
Modules complémentaires chargés : fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: centos.quelquesmots.fr
 * extras: centos.quelquesmots.fr
 * updates: centos.crazyfrogs.org
Résolution des dépendances
--> Lancement de la transaction de test
--> Le paquet epel-release.noarch 0:7-11 sera installé
--> Résolution des dépendances terminée

Dépendances résolues

=====
Package Architecture Version Dépôt Taille
=====
Installation :
epel-release noarch 7-11 extras 15 k
=====
Résumé de la transaction
=====
Installation 1 Paquet

Taille totale des téléchargements : 15 k
Taille d'installation : 24 k
Downloading packages:
epel-release-7-11.noarch.rpm | 15 kB 00:00:00
```

Une fois le script exécuté, nous avons accès à notre Issabel tout fraîchement installé via son adresse IP



Le login et le mot de passe sont définis lors de l'installation.




Une fois connecté, il va falloir ajouter des utilisateurs.

Dans *System > Users > Users*, cliquer sur *Create New User* :

[System](#) / [Users](#) / **Users**

+ Create New User

Delete User

	Login	Real Name	Group	Extension
	admin		Administrator	No extension associated
	maxime	maxime	Administrator	No extension associated
	steven	steven	Administrator	No extension associated

Issabel is licensed under [GPL](#). 2006 - 2019.

Ces utilisateurs pourront alors se connecter sur l'interface web.

CONFIGURATION DE BASE

Création des extensions

Pour commencer à passer des appels, nous allons ajouter des “extensions” (ID qui définit l’appareil utilisé par un nombre) à nos utilisateurs. Pour cela, il faut aller dans *PBX > PBX Configuration > Add a new extension* :

Add SIP Extension

- Add Extension

User Extension	<input type="text" value="1001"/>
Display Name	<input type="text" value="hatim"/>
CID Num Alias	<input type="text"/>
SIP Alias	<input type="text"/>

- Extension Options

Le mot de passe est défini dans le champ “secret”.

This device uses sip technology.

secret	<input type="text" value="71f4ca7414593bfb2e858a02a2d43a27"/>
dtmfmode	<input type="text" value="RFC 2833"/>
nat	<input type="text" value="No - RFC3581"/>

Pour émettre des appels il faut aussi activer le NAT dans la configuration des extensions.

nat	<input type="text" value="Yes"/>
-----	----------------------------------

La prochaine chose à faire est de lier les comptes utilisateurs aux extensions :

System / Users / Users

Main Fields

Login: *	<input type="text" value="hatim"/>	Name (Ex. John Doe):	<input type="text" value="hatim"/>
Password: *	<input type="password" value="....."/>	Retype password: *	<input type="password" value="....."/>
Group: *	<input type="text" value="Administrator"/>		

PBX Profile

Extension:	<input type="text" value="1001"/>
------------	-----------------------------------

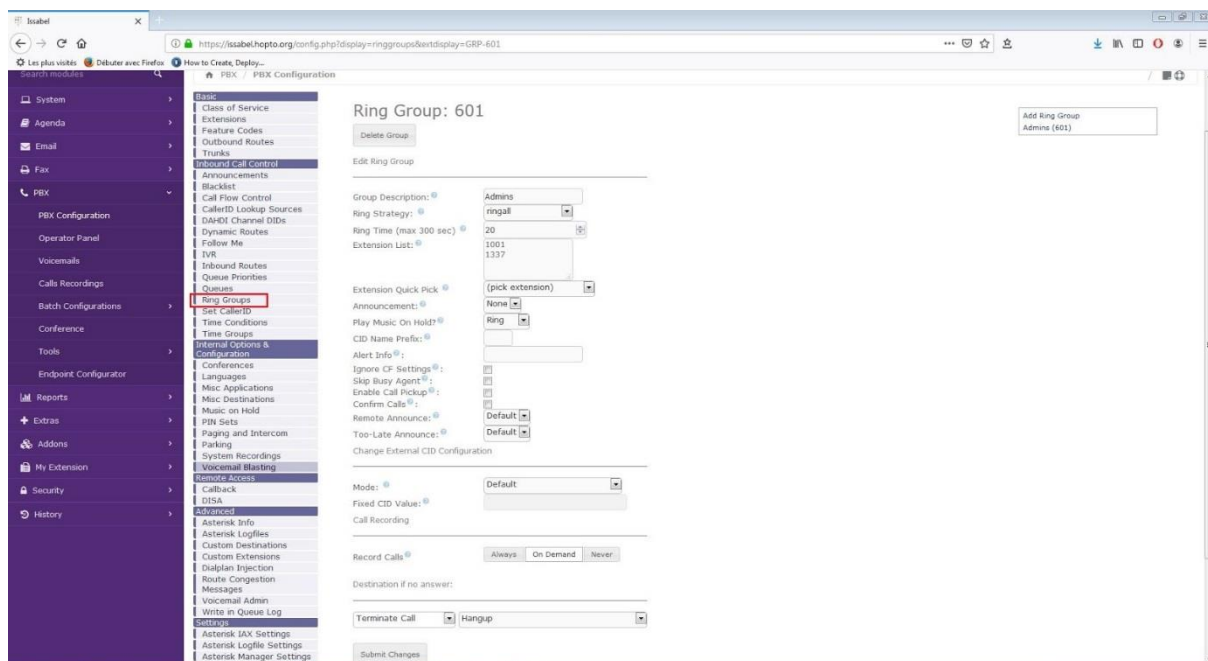
Dans le profil utilisateur (*System > Users > Users*) ajouter le numéro d’extension dans PBX Profile.

Ring Groups - Groupement d'appels

Pour configurer un groupement d'appels il faut se rendre dans *PBX > PBX Configuration > Ring Groups* (dans *Inbound Call Control*), puis cliquer sur *Add Ring Group*.

Pour atteindre ce groupement (601 dans notre cas), il faut :

- Indiquer le numéro à appeler.
- Entrer une description du groupement (601 correspond au groupement des Admins).
- Choisir une stratégie d'appel : « ringall » fera sonner tous les numéros disponibles du groupement en même temps jusqu'à ce que quelqu'un décroche l'appel.
- Ajouter les extensions concernées par ce groupement d'appels. Nos administrateurs sont 1001 et 1337.
- Préciser la marche à suivre en cas de non-réponse (*Destination if no answer*). Dans notre cas, l'appel se termine puis raccroche.



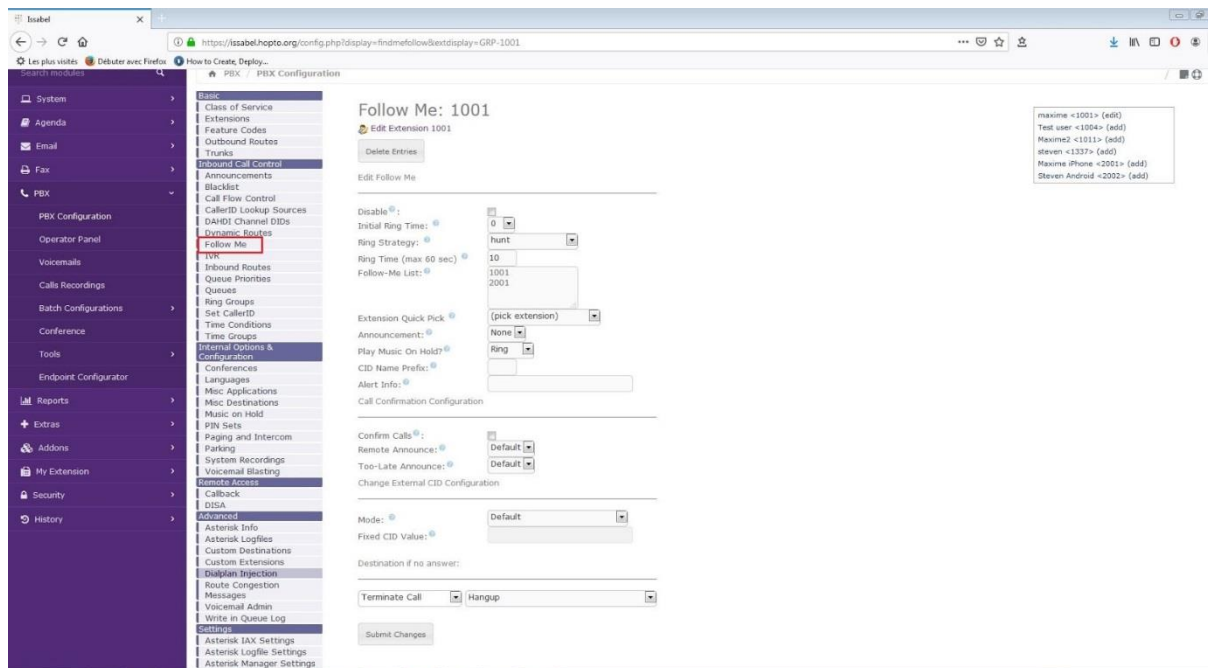
Exemple de notre configuration d'un groupement d'appels.

Follow Me - Redirection d'appels

Pour configurer une redirection d'appels, il faut se rendre dans *PBX > PBX Configuration > Follow Me* (dans *Inbound Call Control*).

Sélectionner une extension, par exemple 1001.

Pour rediriger un appel, il faut sélectionner « hunt » dans *Ring Strategy*, c'est-à-dire que l'appelant fera sonner la première extension avant d'être redirigé vers la seconde extension en cas de non-réponse (pour nous, si 1001 ne répond pas c'est 2001 qui reçoit l'appel). Il faut enfin choisir que faire en cas de non-réponse (fin d'appel puis raccrocher dans notre cas).



Messagerie vocale

Se rendre dans *PBX > PBX Configuration > Extensions* et sélectionner l'extension à modifier. Dans la rubrique *Voicemail* passer le statut à « Enabled » et ajouter un mot de passe.

Il est alors possible de laisser un message sur l'extension contactée s'il n'y a pas de réponse. Le client pourra alors composer le *97 depuis son Softphone pour accéder à sa messagerie. Il lui sera demandé son mot de passe. Il est par ailleurs possible d'accéder à sa messagerie depuis une autre extension (un autre Softphone) en composant le *98. Il faudra entrer son numéro d'extension puis son mot de passe.

- Voicemail

Status	Enabled ▾
Voicemail Password ⓘ	1234
Email Address ⓘ	
Pager Email Address ⓘ	
Email Attachment ⓘ	<input type="checkbox"/> yes <input type="checkbox"/> no
Play CID ⓘ	<input type="checkbox"/> yes <input type="checkbox"/> no
Play Envelope ⓘ	<input type="checkbox"/> yes <input type="checkbox"/> no
Delete Voicemail ⓘ	<input type="checkbox"/> yes <input type="checkbox"/> no
VM Options ⓘ	
VM Context ⓘ	default

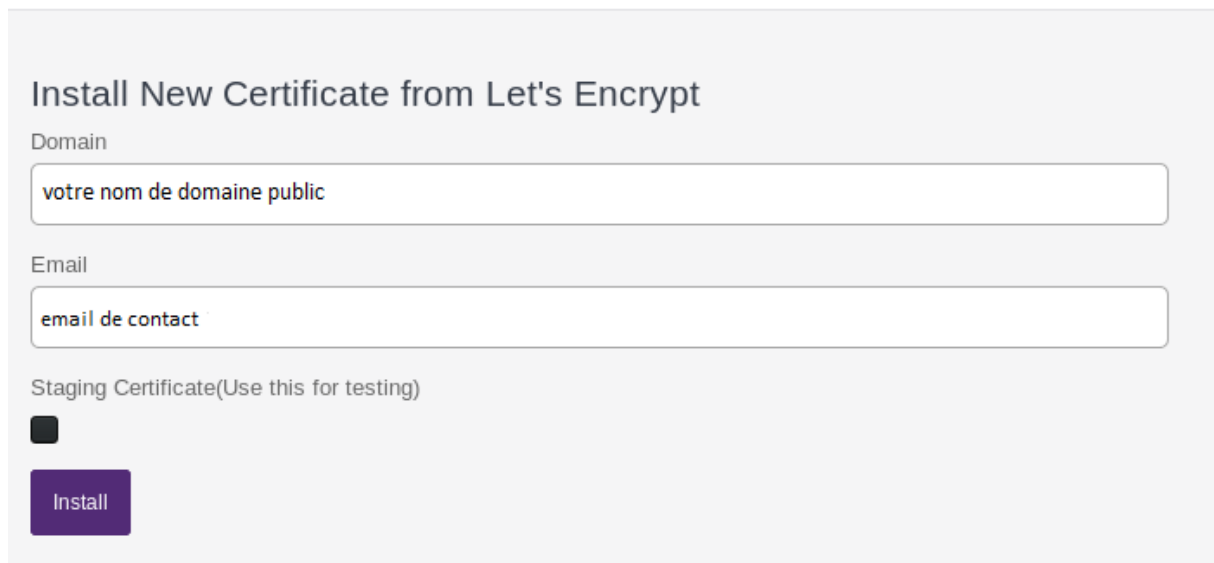
- VmX Locator

PASSAGE DU SITE EN HTTPS

Nous allons maintenant mettre en place un certificat sur notre serveur afin de remplacer le certificat autosigné actuellement utilisé.

Pour cela, nous allons avoir besoin d'un nom de domaine public. Il est possible d'en obtenir via le site <https://www.noip.com>.

De retour sur Issabel, dans *Security > HTTPS certificate*, saisir le nom de domaine obtenu ainsi qu'une adresse email et cliquer sur *Install*.



Le résultat de l'opération :

```
Output log:
-----
Congratulations! You have successfully enabled https://issabel.hopto.org

You should test your configuration at:
https://www.ssllabs.com/ssltest/analyze.html?d=issabel.hopto.org
-----
IMPORTANT NOTES:
- Congratulations! Your certificate and chain have been saved at:
/etc/letsencrypt/live/issabel.hopto.org/fullchain.pem
Your key file has been saved at:
/etc/letsencrypt/live/issabel.hopto.org/privkey.pem
Your cert will expire on 2019-10-06. To obtain a new or tweaked
version of this certificate in the future, simply run certbot again
with the "certonly" option. To non-interactively renew *all* of
your certificates, run "certbot renew"
- If you like Certbot, please consider supporting our work by:

Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate
Donating to EFF: https://eff.org/donate-le
```

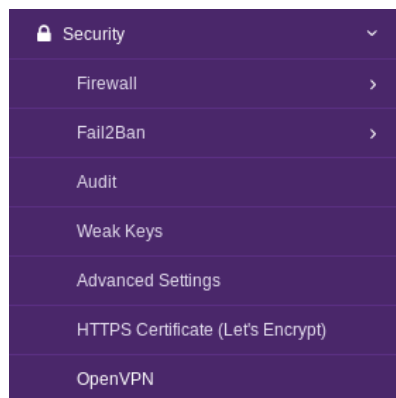
MISE EN PLACE DU VPN

Côté serveur

Tout d'abord, il faut s'assurer qu'OpenVPN soit bien installé sur notre serveur (l'installer si ce n'est pas le cas) :

```
user@issabel /h/user> yum install openvpn-2.4.7-1.el7.x86_64
Modules complémentaires chargés : fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: centos.quelquesmots.fr
* commercial-addons: repo.issabel.org
* epel: fr2.rpmfind.net
* extras: centos.quelquesmots.fr
* issabel-base: repo.issabel.org
* issabel-extras: repo.issabel.org
* issabel-updates: repo.issabel.org
* issabel-updates-sources: repo.issabel.org
* updates: centos.mirrors.proxad.net
Le paquet openvpn-2.4.7-1.el7.x86_64 est déjà installé dans sa dernière version
Rien à faire
```

Une fois OpenVPN installé, un onglet apparaît dans la section *Security* de l'interface Web



Pour commencer la configuration du VPN, entrer les informations demandées :

OpenVPN Settings

OpenVPN Configuration
Create Client Certificates
OpenVPN Status

1. Create Vars File
2. Clean All
3. Build CA
4. Build Server's Keys
5. Build Server Configuration

Common Values for Vars File

This is your actual "vars" configuration

Country Name ?	FR
State or Province ?	FR
Locality ?	France
Organization Name ?	Nom de l'entreprise
Organization Unit Name ?	Students
Common Name ?	IssabelServer
Name ?	nom de domaine public
Email ?	email de contact

Previous
Next

Cliquer sur *Suivant* jusqu'à l'onglet 5 :

OpenVPN Settings

OpenVPN Configuration
Create Client Certificates
OpenVPN Status

1. Create Vars File
2. Clean All
3. Build CA
4. Build Server's Keys
5. Build Server Configuration

Server Configuration

This is your actual Server configuration

IP or HOST ?	IP PUBLIC DU SERVEUR	Set Your Public IP
Listening Port ?	1194	
Protocol ?	UDP	
Dev ?	TUN	
Server Network ?	192.168.0.0	Server Mask ? 255.255.255.0
Keep Alive ?	10	Timeout ? 120

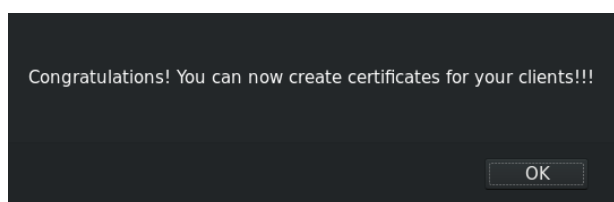
Advanced Settings(optional) ?

Previous
Finish

On renseigne ici :

- Le port d'écoute : 1194
- Le réseau du VPN : 192.168.0.0
- Le masque du réseau VPN : 255.255.255.0

Une fois configuré, un message pop-up apparaît, indiquant que l'on peut passer à la création de certificats pour nos clients.



Côté client

Security / OpenVPN

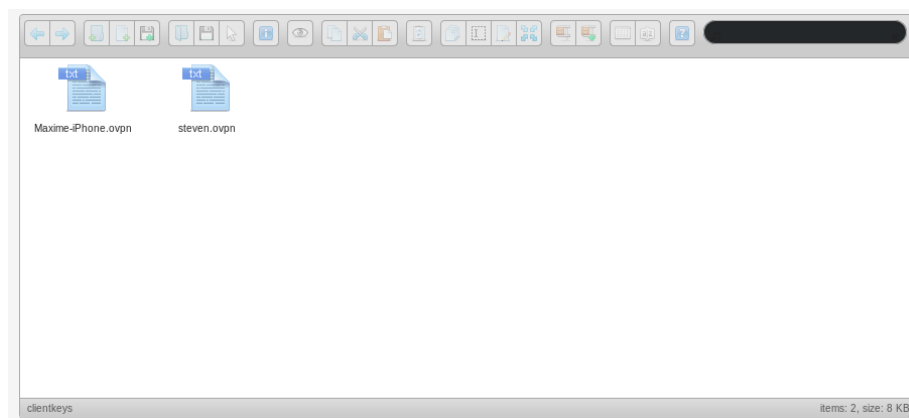
Here you can create predefined keys and certificates for clients, so they can connect to the vpn.

Client Type ? Embedded Windows Client (One file) Client Name ? ClientVPN

[Generate Configs](#)

Se rendre dans *Create Certificate Clients*. Ici, on va choisir “Embedded Windows Client” en *Client Type*. Cela nous permettra de générer un **fichier.ovpn** qui est utilisable par n’importe quel client OpenVPN (pas seulement Windows mais aussi Linux, Android ou iOS).

Remplir le champ *Client Name* et cliquer sur *Generate Configs*.



Il ne reste donc plus qu’à télécharger notre fichier (*Clic droit > Download*).

Une fois récupéré (le fichier.ovpn peut être envoyé par mail par exemple), il est possible de se connecter au VPN depuis un système linux de la façon suivante :

user@PC:\$ sudo openvpn fichier.ovpn

```

steven@pc-420:~/Documents/uxperiment/HOPIOS$ sudo openvpn steven.ovpn
[sudo] Mot de passe de steven:
Thu Jul 11 15:06:21 2019 openvpn 2.4.0 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/TKINTFO] [AEAD] built on Oct 14 2018
Thu Jul 11 15:06:21 2019 library versions: OpenSSL 1.0.2s  28 May 2019, LZO 2.08
Thu Jul 11 15:06:21 2019 WARNING: No server certificate verification method has been enabled. See http://openvpn.net/howto.html#mitm for more info.
Thu Jul 11 15:06:21 2019 TCP/UDP: Preserving recently used remote address: [AF_INET]192.168.0.6:1194
Thu Jul 11 15:06:21 2019 Socket Buffers: R=[212992->212992] S=[212992->212992]
Thu Jul 11 15:06:21 2019 UDP link local: (not bound)
Thu Jul 11 15:06:21 2019 UDP link remote: [AF_INET]192.168.0.6:1194
Thu Jul 11 15:06:21 2019 TLS: Initial packet from [AF_INET]192.168.0.6:1194, sid=5efbc978 ca0bd514
Thu Jul 11 15:06:21 2019 VERIFY OK: depth=1, C=FR, ST=FR, L=France, O=IT-TEK, OU=Students, CN=IssabelServer, name=steven, email=steven@pc-420.org, emailAddress=steven@pc-420.org
Thu Jul 11 15:06:21 2019 VERIFY OK: depth=0, C=FR, ST=FR, L=France, O=IT-TEK, OU=Students, CN=server, name=server, email=server@pc-420.org, emailAddress=server@pc-420.org
Thu Jul 11 15:06:21 2019 Control Channel: TLSv1.2, cipher TLSv1/SSLv3 ECDHE-RSA-AES256-GCM-SHA384, 1024 bit RSA
Thu Jul 11 15:06:21 2019 [server] Peer Connection Initiated with [AF_INET]192.168.0.6:1194
Thu Jul 11 15:06:23 2019 SENT CONTROL [server]: 'PUSH_REQUEST' (status=1)
Thu Jul 11 15:06:23 2019 PUSH: Received control message: 'PUSH_REPLY,route 192.168.0.0 255.255.255.0,topology net30,ping 10,ping-restart 120,ifconfig 192.168.0.6 192.168.0.5,peer-id 0,cipher AES-256-GCM'
Thu Jul 11 15:06:23 2019 OPTIONS IMPORT: timers and/or timeouts modified
Thu Jul 11 15:06:23 2019 OPTIONS IMPORT: --ifconfig/up options modified
Thu Jul 11 15:06:23 2019 OPTIONS IMPORT: route options modified
Thu Jul 11 15:06:23 2019 OPTIONS IMPORT: peer-id set
Thu Jul 11 15:06:23 2019 OPTIONS IMPORT: adjusting link_mtu to 1625
Thu Jul 11 15:06:23 2019 OPTIONS IMPORT: data channel crypto options modified
Thu Jul 11 15:06:23 2019 Data Channel Encrypt: Cipher 'AES-256-GCM' initialized with 256 bit key
Thu Jul 11 15:06:23 2019 Data Channel Decrypt: Cipher 'AES-256-GCM' initialized with 256 bit key
Thu Jul 11 15:06:23 2019 ROUTE GATEWAY 192.168.0.0 255.255.0.0 IFACE=wlx350 HWADDR=28:c2:dd:85:f3:b1
Thu Jul 11 15:06:23 2019 TUN/TAP device tun0 opened
Thu Jul 11 15:06:23 2019 TUN/TAP TX queue length set to 100
Thu Jul 11 15:06:23 2019 do ifconfig, tt->did_ifconfig_ipv6_setup=0
Thu Jul 11 15:06:23 2019 /sbin/ip link set dev tun0 up mtu 1500
Thu Jul 11 15:06:23 2019 /sbin/ip addr add dev tun0 local 192.168.0.6 peer 192.168.0.5
Thu Jul 11 15:06:23 2019 /sbin/ip route add 192.168.0.0/24 via 192.168.0.5
Thu Jul 11 15:06:23 2019 Initialization Sequence Completed

```

Un client Windows devra cliquer sur le fichier.ovpn puis effectuer un clic droit sur le fichier et enfin choisir « Start OpenVPN on this config file ».

Pour se connecter via un smartphone, utiliser l'application OpenVPN. Créer un profil et importer le fichier.ovpn.

Une fois connecté, le client remonte sur le serveur et une adresse IP sur le réseau VPN configuré est attribuée :

The screenshot shows the Issabel OpenVPN Settings interface on the left and a Windows command prompt on the right.

Issabel OpenVPN Settings:

- OpenVPN Configuration | Create Client Certificates | OpenVPN Status
- The Server IP Address is: 192.168.0.1
- OpenVPN is Running
- Stop OpenVPN Service | Restart OpenVPN Service
- Connected Clients:

Virtual IP	Common Name	Real IP	Revoke
192.168.0.14	1001	[REDACTED]	Revoke
192.168.0.6	steven	[REDACTED]	Revoke
- Created Certificates: 4

Name	Revoke
server	Revoke
steven	Revoke
Maxime-IPhone	Revoke
1001	Revoke
- Revoked Clients: 0

Windows Command Prompt:

```

C:\Windows\system32\cmd.exe
Microsoft Windows [version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Tous droits réservés.

C:\Users\Maxime>ipconfig

Configuration IP de Windows

Carte Ethernet Connexion au réseau local 3 :
    Suffixe DNS propre à la connexion. . . : fe80::558:319f:171:903c%23
    Adresse IP de liaison locale. . . . . : 192.168.0.14
    Adresse IP4. . . . . : 192.168.0.14
    Masque de sous-réseau. . . . . : 255.255.255.252
    Passerelle par défaut. . . . . :

Carte Ethernet Connexion Réseau Bluetooth :
  
```

Un client Linux crée alors une interface tun0 (interface dans VPN) qui obtient elle aussi une adresse du réseau VPN 192.168.0.0/24.

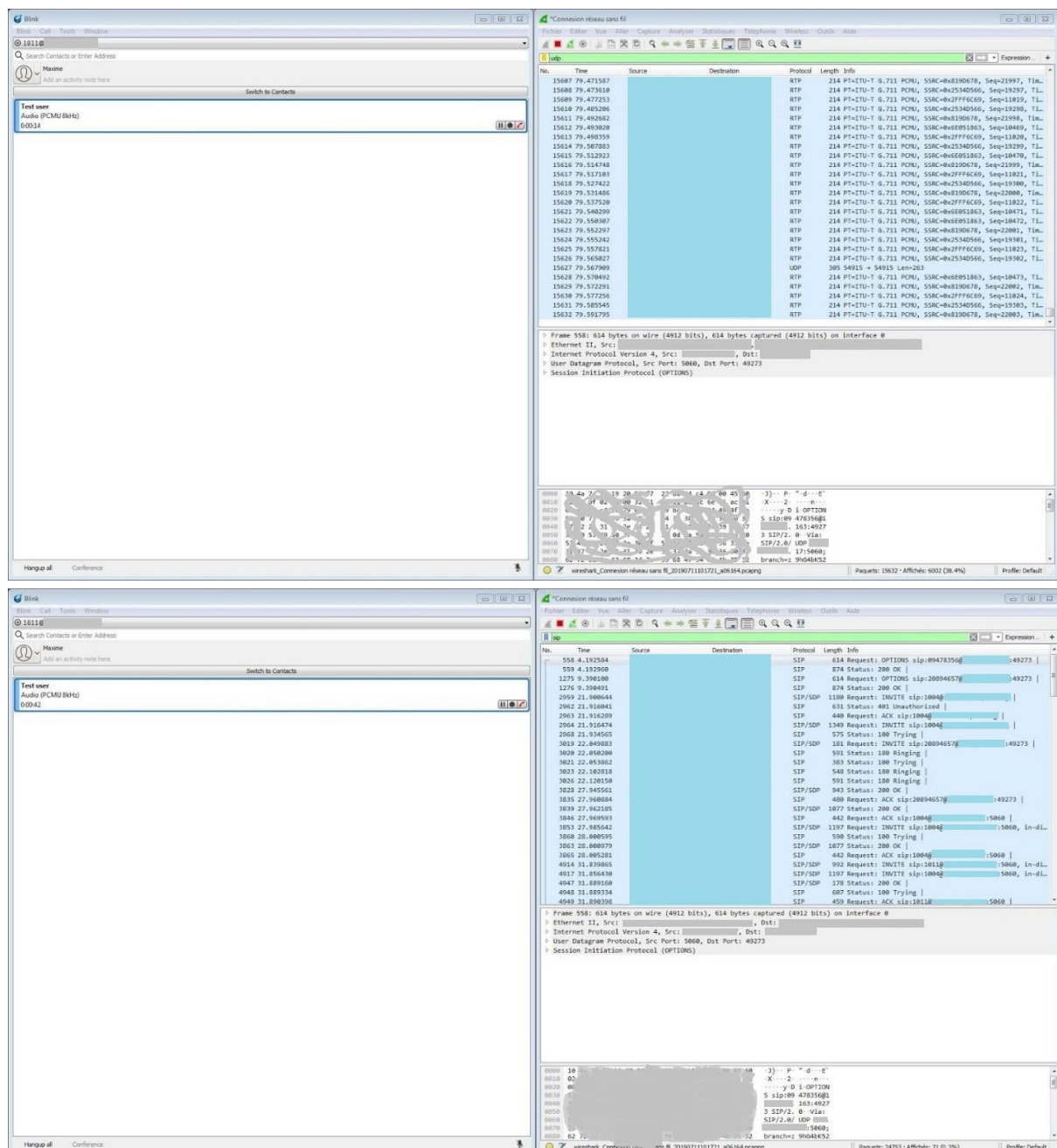
```

tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
    inet 192.168.0.6 netmask 255.255.255.255 destination 192.168.0.5
    inet6 fe80::294b:21d0:99d1:8d17 prefixlen 64 scopeid 0x20<link>
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 100 (UNSPEC
)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4 bytes 204 (204.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
  
```


CONFIGURATION DU TLS SUR LE SERVEUR ET MISE EN PLACE DU SRTP

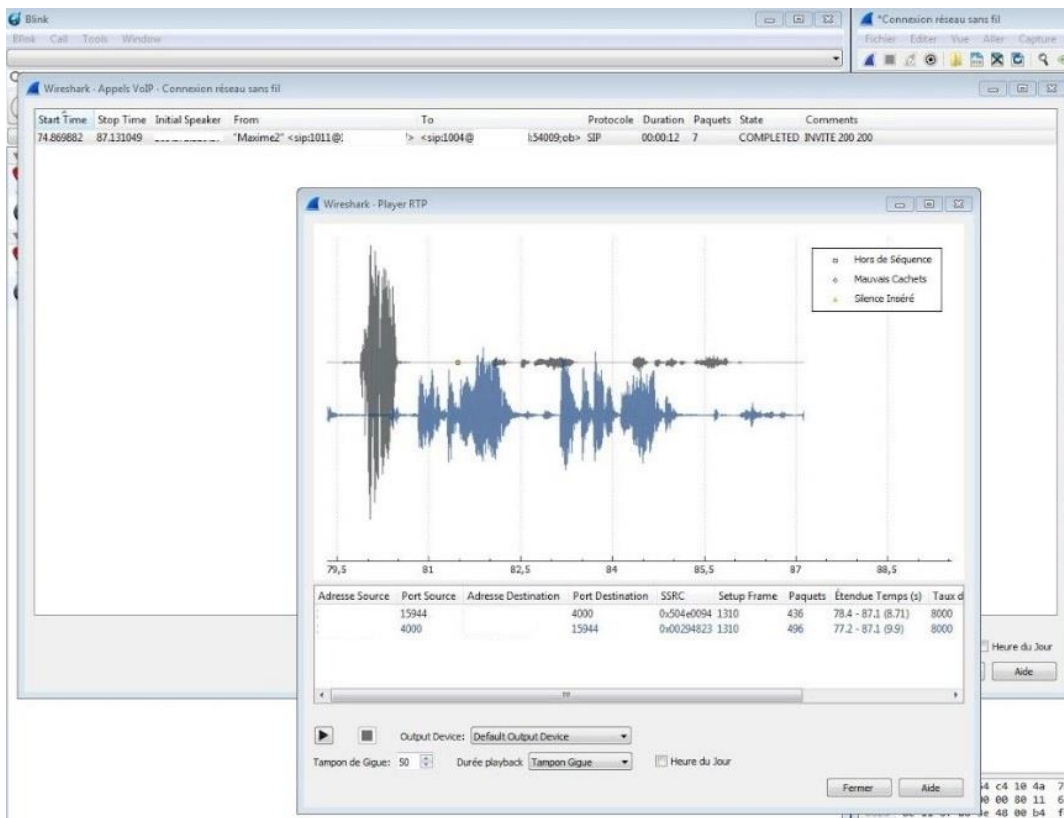
Pourquoi est-il important de sécuriser les communications passant par la VoIP ?

Les appels passés entre nos différents appareils utilisent les protocoles SIP et RTP. En utilisant Wireshark, nous avons surveillé un appel entre deux extensions. Voici ce que nous avons obtenu.



Différentes informations apparaissent en clair : le protocole, le codec utilisé, les extensions concernées par l'appel.

Il est même possible de rejouer la conversation dans son intégralité.



Passons à la configuration du serveur et à la mise en place du TLS pour chiffrer les flux.

Générer les certificats nécessaires au chiffrement des appels

SERVEUR

Nous avons utilisé le script présent sur un serveur Asterisk 11 sous `/usr/share/doc/asterisk/asterisk.../ast_tls_cert` (non disponible sur Asterisk 13). Il est aussi disponible ici : https://github.com/asterisk/asterisk/blob/master/contrib/scripts/ast_tls_cert.

Après l'avoir importé sur le serveur, il faut créer un dossier `keys` s'il n'existe pas déjà sous `/etc/asterisk/`.

```
mkdir /etc/asterisk/keys
```

Lancer le script pour générer les certificats relatifs au serveur.

```
.ast_tls_cert -C issabel.hopto.org -O "Les11Commandements" -d /etc/asterisk/keys/
[root@issabel keys]# ls -la /etc/asterisk/keys
total 88
drwxrwx--x 3 asterisk asterisk 4096 Jul 10 11:43 .
drwxrwxr-x 3 asterisk asterisk 12288 Jul 10 09:45 ..
-rw----- 1 asterisk asterisk 1237 Jul 10 09:31 asterisk.crt
-rw----- 1 asterisk asterisk 586 Jul 10 09:31 asterisk.csr
-rw----- 1 asterisk asterisk 887 Jul 10 09:31 asterisk.key
-rw-r--r-- 1 asterisk asterisk 2124 Jul 10 09:31 asterisk.pem
-rwxr-xr-x 1 asterisk asterisk 5031 Jul 9 16:51 ast_tls_cert
-rw----- 1 asterisk asterisk 164 Jul 10 09:30 ca.cfg
-rw----- 1 asterisk asterisk 1777 Jul 10 09:31 ca.crt
-rw----- 1 asterisk asterisk 3311 Jul 10 09:31 ca.key
-rw----- 1 asterisk asterisk 131 Jul 10 11:38 tmp.cfg
```

Sur la plateforme web, il faut se rendre dans *PBX > PBX Configuration > Asterisk SIP Settings* puis tout en bas de la page, dans la rubrique *Other SIP Settings*, ajouter des champs en cliquant sur *Add Field* puis configurer comme suit.

tlsenable	=	yes
tlsbindaddr	=	0.0.0.0
tls_certfile	=	/etc/asterisk/keys/asterisk.crt
tlscafile	=	/etc/asterisk/keys/ca.crt
tlscipher	=	ALL
tlsclientmethod	=	tlsv1
tls_certfile	=	/etc/asterisk/keys/VPN/ca.crt
tlscafile	=	/etc/asterisk/keys/VPN/ca.crt

Add Field

Submit Changes

Les champs en doublon `tls_certfile` et `tlscafile` correspondent aux clients se connectant directement au domaine dans le premier cas, et dans le second cas à ceux se connectant en VPN.

Ces paramètres se retrouvent dans le fichier `/etc/asterisk/sip_general_additional.conf`.

Modifier le fichier `/etc/asterisk/pjsip.conf` en ajoutant à la fin du fichier les paramètres suivants :

```
type=transport
```

```
protocol=tls
bind=0.0.0.0:5061
cert_file=/etc/asterisk/keys/asterisk.crt
priv_key_file=/etc/asterisk/keys/asterisk.key
method=tlsv1
```

Sauvegarder.

Sur la plateforme web, se rendre dans *PBX > PBX Configuration > Extensions* (dans la rubrique *Basic*).

Sélectionner l'extension à modifier puis modifier les champs suivants pour mettre en place le TLS ainsi que le SRTP :

type	friend
nat	Yes
port	5061
qualify	yes
qualifyfreq	60
transport	TLS Only
avpf	No
force_avp	No
icesupport	No
dtlsenable	No
dtlsverify	No
dtlssetup	Incoming and Outgoing
dtlscertfile	
dtlsprivatekey	
rtcp_mux	No
encryption	Yes (SRTP only)
callgroup	
pickupgroup	
disallow	
allow	
dial	SIP/1001
accountcode	
mailbox	1001@device
vmexten	
deny	
permit	0.0.0.0/0.0.0.0
Class of Service	ALLOW ALL (Default)

Puis sauvegarder et appliquer les changements.

Relancer le service asterisk sur le serveur pour prendre la nouvelle configuration en compte.

```
/etc/init.d/asterisk restart
```

CLIENT (CAS DE BLINK UNIQUEMENT)

Si le client utilise Blink en tant que Softphone il faut générer les certificats à attribuer au client.

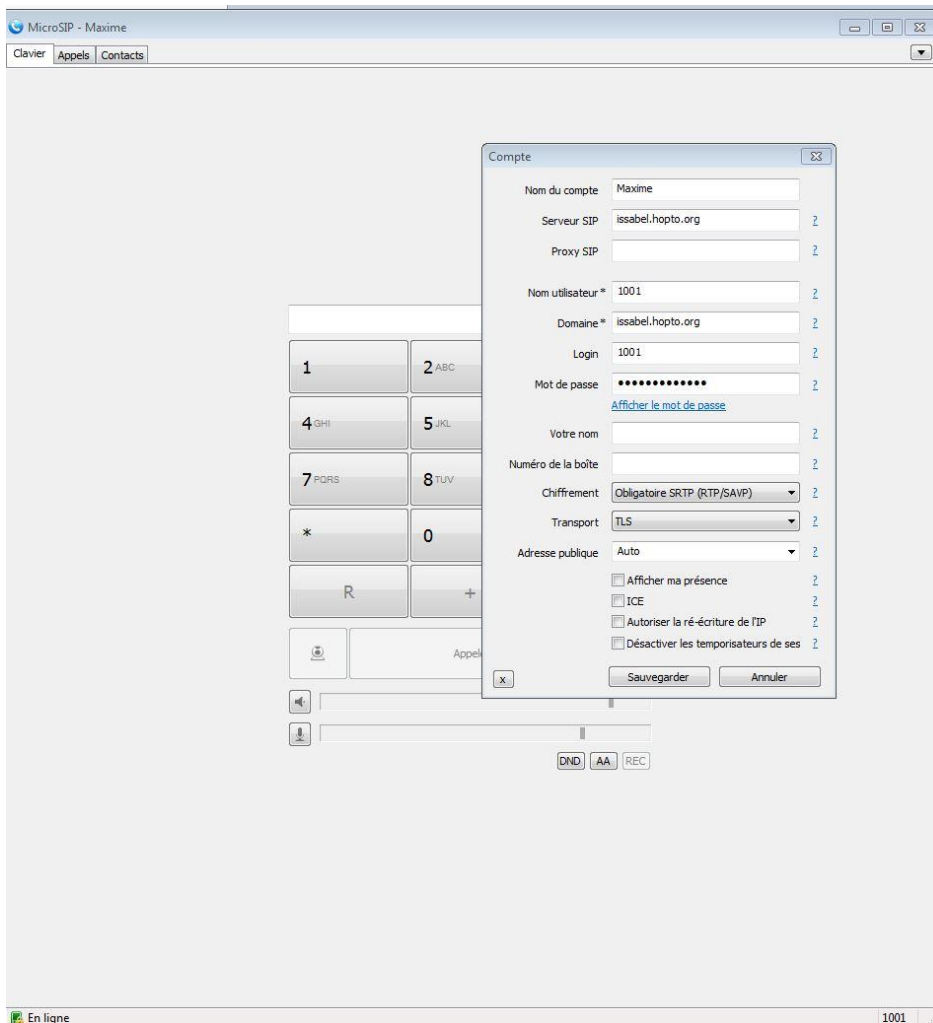
```
./ast_tls_cert -m client -C @IP_client_ou_FQDN -O "Les11Commandements" -c ca.crt -k ca.key -d /etc/asterisk/keys -o <nom_du_fichier>
```

```
[root@issabel keys]# ls -la
total 88
drwxrwx--x 3 asterisk asterisk 4096 Jul 10 11:43 .
drwxrwxr-x 3 asterisk asterisk 12288 Jul 10 09:45 ..
-rw----- 1 asterisk asterisk 1241 Jul 10 09:53 nom_du_fichier.crt
-rw----- 1 asterisk asterisk 590 Jul 10 09:53 nom_du_fichier.csr
-rw----- 1 asterisk asterisk 887 Jul 10 09:53 nom_du_fichier.key
-rw----- 1 asterisk asterisk 2128 Jul 10 09:53 nom_du_fichier.pem
-rw----- 1 asterisk asterisk 1237 Jul 10 09:31 asterisk.crt
-rw----- 1 asterisk asterisk 586 Jul 10 09:31 asterisk.csr
-rw----- 1 asterisk asterisk 887 Jul 10 09:31 asterisk.key
-rw-r--r-- 1 asterisk asterisk 2124 Jul 10 09:31 asterisk.pem
-rw-rw-r-- 1 asterisk asterisk 3619 Jul 9 03:23 asterisk.pem.old
-rwxr-xr-x 1 asterisk asterisk 5031 Jul 9 16:51 ast_tls_cert
-rw----- 1 asterisk asterisk 164 Jul 10 09:30 ca.cfg
-rw----- 1 asterisk asterisk 1777 Jul 10 09:31 ca.crt
-rw----- 1 asterisk asterisk 3311 Jul 10 09:31 ca.key
-rw----- 1 asterisk asterisk 131 Jul 10 11:38 tmp.cfg
```

Il faudra alors transférer au client les fichiers **ca.crt** et **nom_du_fichier.pem** (correspondant au FQDN ou @IP du client).

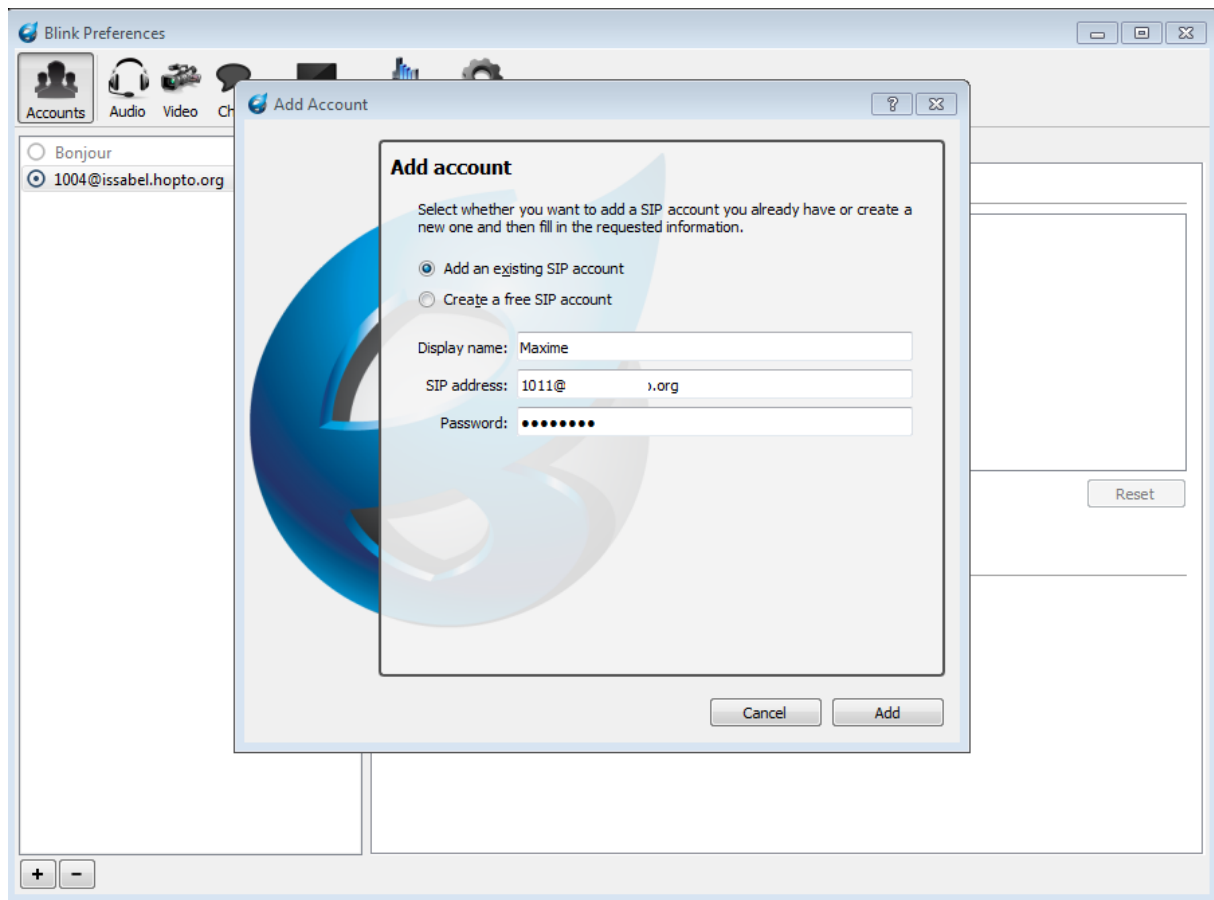
Configuration des clients Softphones

PC WINDOWS : MICROSIP

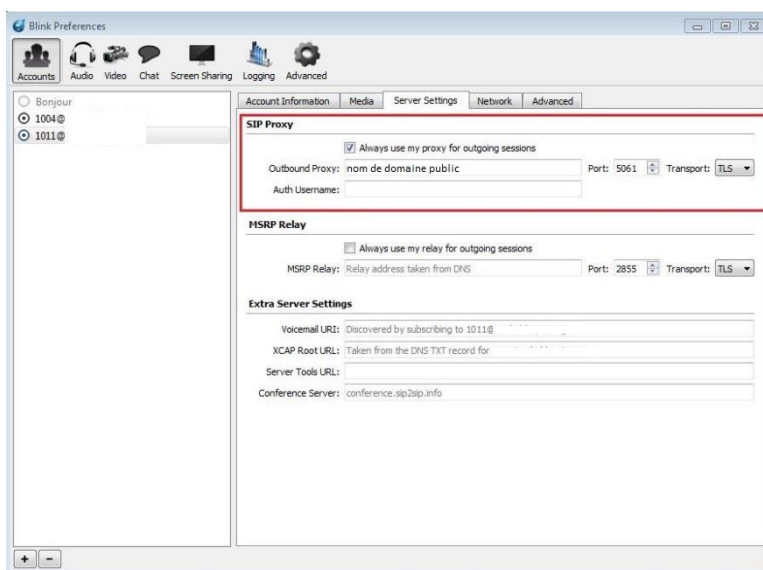
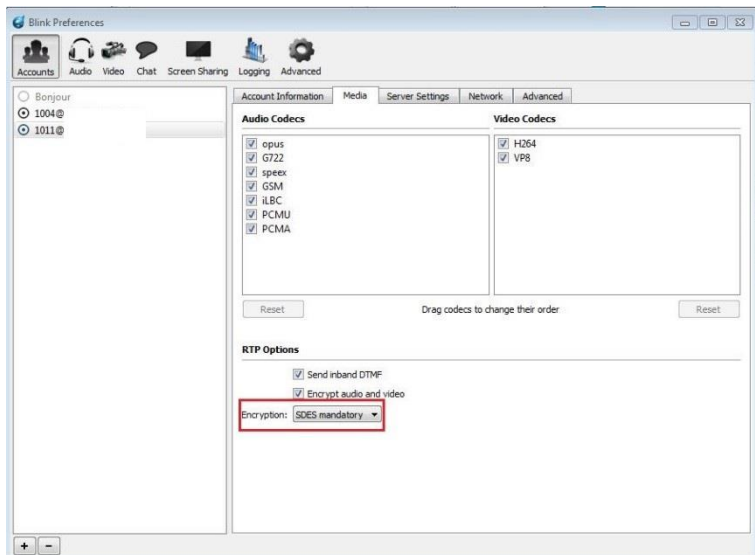


Il faut entrer les identifiants permettant de se connecter au serveur SIP et forcer le chiffrement obligatoire en SRTP ainsi que le transport en TLS. Le client se connecte et un cadenas apparaît sur le téléphone vers en bas de la fenêtre indiquant que le chiffrement est en place.

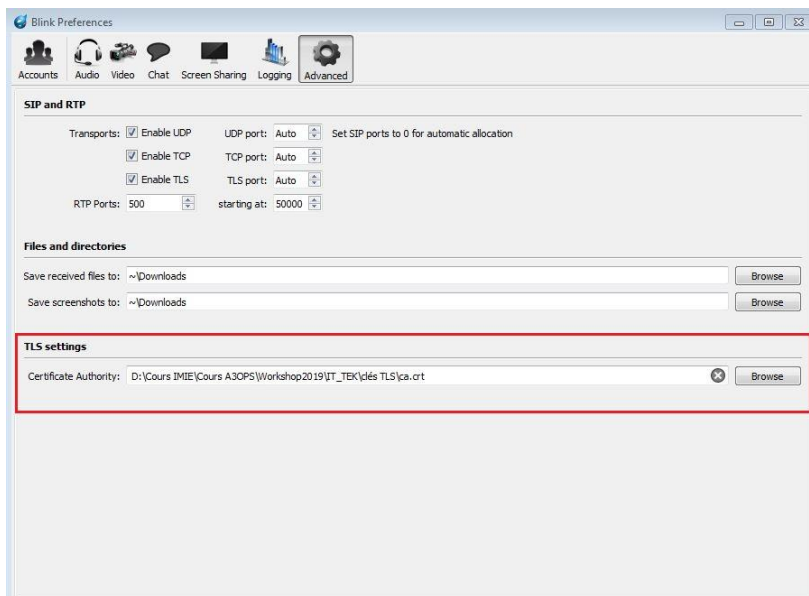
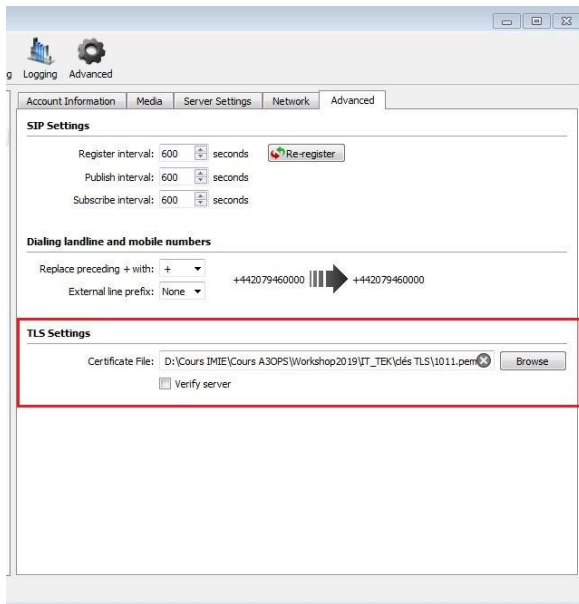
PC : BLINK



Voici comment configurer le client Blink pour sécuriser les communications (SIP over TLS et SRTP).



Il faut par ailleurs renseigner le chemin vers les certificats importés du serveur Issabel (nom_du_fichier.pem et ca.crt).

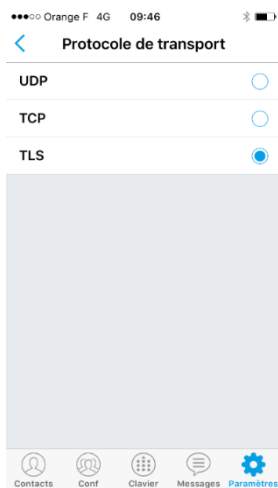


Le premier cadenas signifie que le SIP over TLS est bien activé, le second que le SRTP est activé.

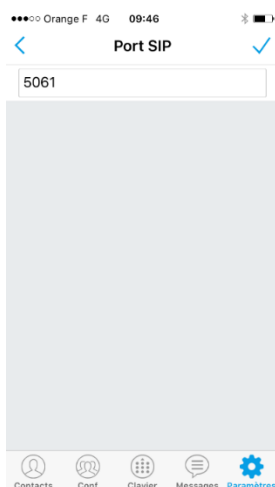
Configuration de GrandStream Wave sur iPhone (identique à Android)



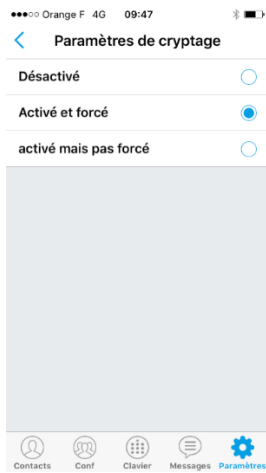
Renseigner le FQDN du serveur Issabel ainsi que les identifiants de connexion.



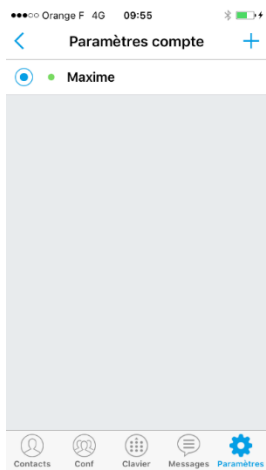
Indiquer le type de transport souhaité, ici TLS.



Indiquer le port sur lequel le client Softphone communiquera avec le serveur, ici 5061 (paramétré en amont sur le serveur).



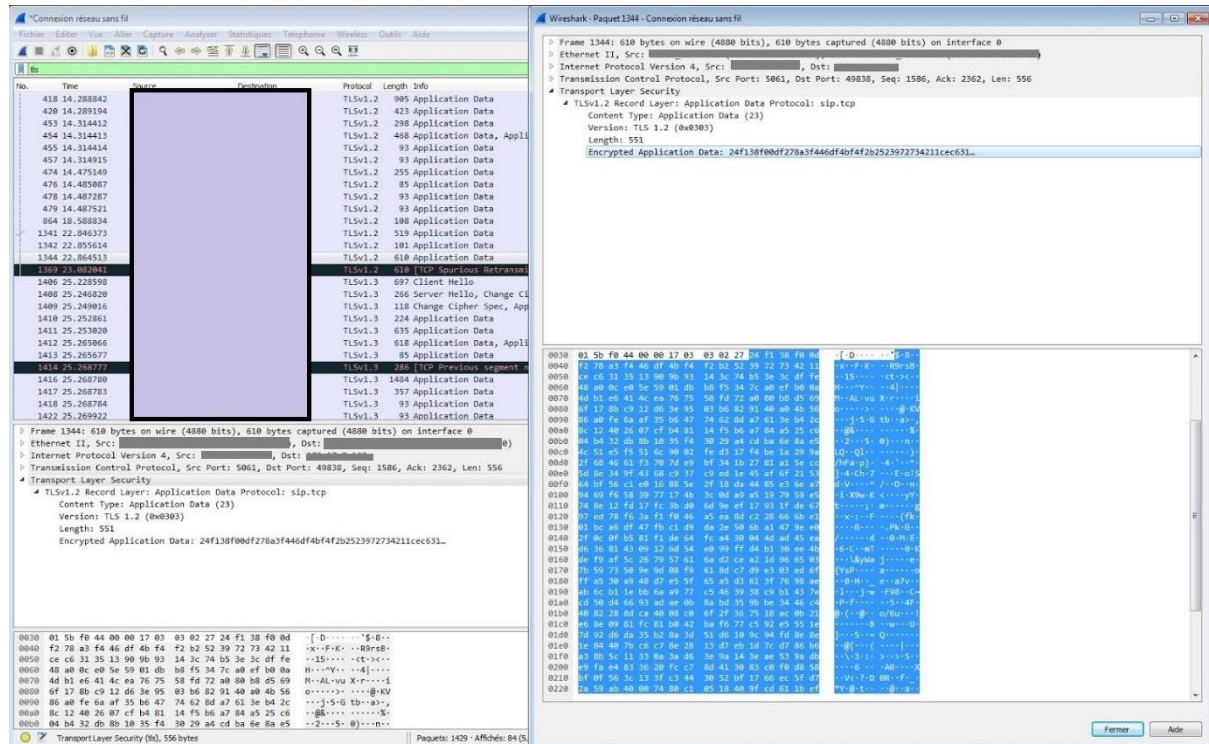
Forcer l'utilisation du chiffrement des communications (SRTP).



Le compte est bien connecté.

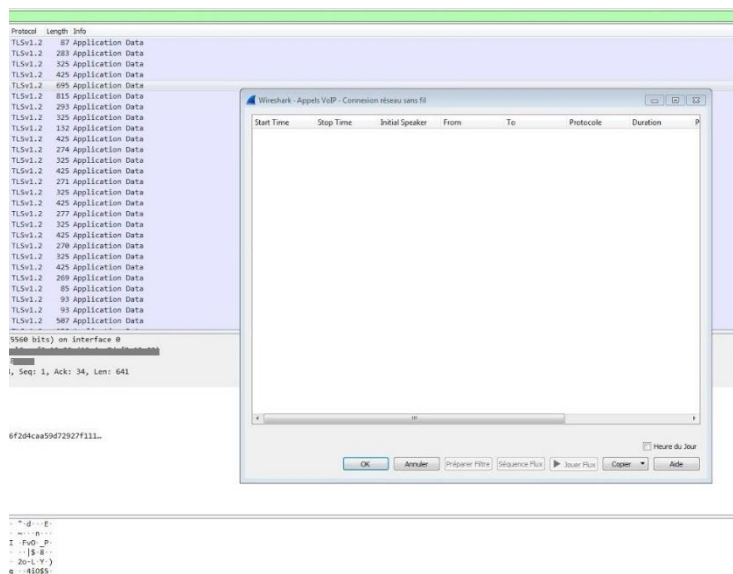
Vérification du chiffrement

Maintenant que le chiffrement est en place, voyons ce que nous obtenons en supervisant un appel avec Wireshark.



La sécurisation est bien effective, le protocole SIP est associé au TLS, la communication se fait bien sur le port 5061.

Par ailleurs, il est maintenant impossible de rejouer la conversation, elle n'apparaît plus dans les appels VoIP passés.



CONTRAINTES, LIMITES ET RECOMMANDATIONS :

Le premier temps du projet s'est construit de la façon suivante : veille technologique, mise en place de la solution choisie et tests.

Aucun d'entre nous ne maîtrisait le sujet de la téléphonie, la difficulté première était d'appréhender la technologie ainsi que les différents outils disponibles. Après une veille technologique relativement rapide due au planning imposé, nous avons décidé d'utiliser Issabel (Asterisk) selon les prérequis du cahier des charges.

Issabel est un projet qui reste jeune (2017), et même s'il s'agit d'une branche gratuite issue d'Elastix, il n'existe pas beaucoup de documentation sur cet outil. Par ailleurs, la communauté d'Issabel est principalement espagnole, et les documents et forums sont ainsi rédigés dans cette langue.

Lors de nos tests sur une première installation d'Issabel, nous avons dû procéder à une réinstallation pour utiliser une version d'Asterisk plus récente et ainsi se rapprocher de l'existant.

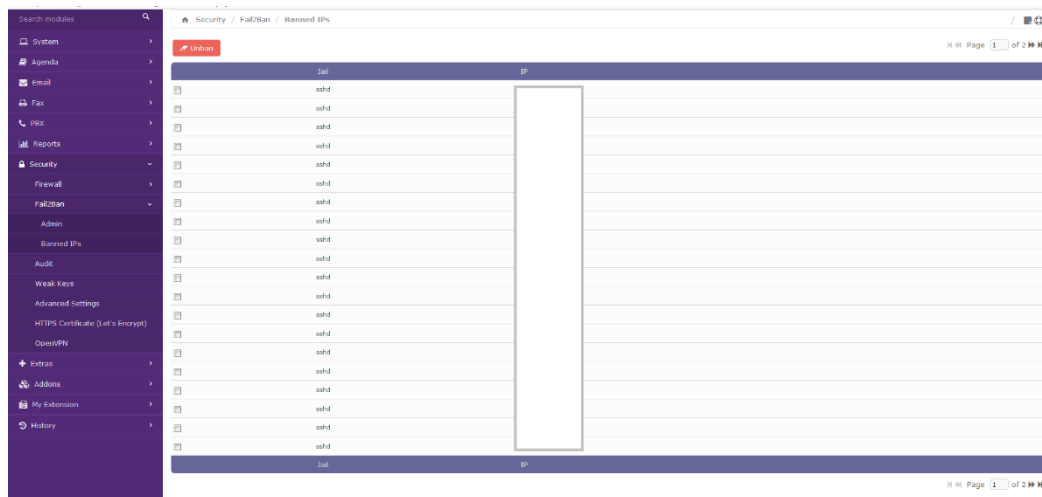
Après avoir installé Issabel sur un serveur local, est apparu un autre point contraignant. En effet, pour tester l'utilisation du VPN, nous avons dû mettre en place une infrastructure représentant deux réseaux séparés (schématiquement le LAN et le WAN). Cela permettant sans passer par un pare-feu, de se connecter au VPN et passer des appels dans des conditions s'approchant au mieux des conditions réelles.

Dans le second temps du projet, un serveur externalisé nous a été mis à disposition. Nous avons alors configuré à nouveau notre serveur Issabel.

La dernière contrainte que nous avons rencontrée a été la mise en place de la sécurisation et du chiffrement des échanges en VoIP. Les tests ainsi que le débogage nous ont pris du temps et après quelques jours nous avons réussi à l'implémenter.

Notre solution actuelle nécessite la configuration d'un softphone (en VPN ou non) sur les téléphones mobiles clients souhaitant communiquer via l'IPBX.

Par ailleurs, le fait d'exposer notre serveur sur Internet pour permettre aux clients Softphones de s'y connecter en 4G entraîne un risque d'attaque important. Nous pouvons voir ci-dessous les nombreuses adresses IP bannies après plusieurs tentatives de connexions infructueuses.



Une évolution possible serait d'interconnecter notre serveur à un opérateur de téléphonie. Un externe à l'entreprise n'ayant pas de client softphone et n'étant pas identifié sur le serveur Issabel pourrait alors joindre un téléphone de l'entreprise (une extension), un standard d'accueil ou un serveur vocal à choix multiples en tapant un numéro de téléphone.

La solution Issabel répond aux besoins émis par le client, ainsi qu'aux prérequis techniques (MicroSip ne fonctionnait cependant pas sur un client Debian 9, nécessitant peut-être une configuration supplémentaire pour la détection des périphériques).

Pour aller plus loin dans la sécurisation du serveur, il peut être intéressant de configurer le pare-feu de la machine ainsi que de modifier les ports associés aux différents services utilisés et les personnaliser.

SOURCES ET SITOGRAPHIE

Site officiel :

<https://www.issabel.org/>

Installation d'Issabel :

<http://vmgate.com/installing-issabel-version-4-training-issabel-v4-vmgate/>

Tutoriel configuration d'Issabel (en espagnol) :

https://www.youtube.com/watch?v=-dv2xp_QhtQ

Configuration VPN :

<https://www.youtube.com/watch?v=rtQq1dcFar8>

<https://asteriskmx.org/wp-content/uploads/sites/3/2014/06/Manual-EasyVPN-final-ES.pdf>

Sécurisation du serveur et des échanges :

<https://wiki.asterisk.org/wiki/display/AST/Secure+Calling+Tutorial>

<https://hub.packtpub.com/securing-your-elastix-system/>

Hacking PBX (podcast audio) :

<https://darknetdiaries.com/episode/1/>

Softphones :

<http://icanblink.com/>

<https://www.microsip.org/>

<https://www.zoiper.com/> (payant pour le TLS)

<http://www.grandstream.com/products/ip-voice-telephony/softphone-app/product/grandstream-wave>

Script générer des certificats :

https://github.com/asterisk/asterisk/blob/master/contrib/scripts/ast_tls_cert