

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Funcionalidad del nodo WebRTC en VIVAit Call	
Fecha : 05 de julio de 2022	Número de revisión: Versión 1.1
Objeto del documento : Descripción del nodo WebRTC en VIVAit Call	
Actores (empresas): • MDTEL	

ÍNDICE

ÍNDICE.....	1
1. Introducción.....	3
2. Arquitectura de WebRTC.....	3
3. Instalación y configuración del NodoRTC.....	5
3.1 Instalación del repositorio.....	5
Subir los repositorios a la máquina y configurar los diferentes elementos.....	5
3.2 Configuración de los diferentes elementos.....	11
3.2.1 Configuración de libnice.....	11
3.2.2 Configuración de serCen.....	14
3.2.3 Configuración de Janus.....	18
3.2.4 Configuración de APACHE.....	25
3.2.5 Instalación y configuración de Baikal.....	29
3.2.6 Instalación y configuración del certificado.....	33
3.2.7 Configuración de firewall (iptables).....	38
3.2.8 Instalación y configuración de fail2ban.....	43
3.2.9 Subir FonBo.war a Tomcat.....	51
3.2.10 Proteger Apache con Let's Encrypt (Instalando el paquete Cerbot).....	52
Comprobar la configuración del host virtual de Apache.....	53
3.2.11 Crear script escobatoken.pl.....	59
4. Comprobaciones.....	63
4.1 Logrotate.d y cron.d.....	63
4.2 Grabaciones.....	63
4.3 Ramdisk.....	63
4.4 Certbot.....	64
5. Diagnósticos.....	65
5.1 Herramientas de diagnóstico.....	65
5.1.1 Funcionamiento de envío de emails.....	65

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.1.2 Verificar puertos abiertos de un equipo.....	68
5.1.3 Monitorizar el tráfico de un determinado puerto.....	68
5.1.4 Capturar tráfico con tcpdump.....	68
5.1.5 Comprobar que hay usuarios logados en Webfon.....	69
5.1.6 Gestionar certificados de Let's Encrypt.....	69
5.1.7 Comprobar Conexión server LDAP.....	70
5.2 Logs y comandos.....	71
5.2.1 serCen.....	71
5.2.3 vivait-webfon.....	74
5.2.3 Janus.....	74
5.2.4 Configurar sesiones, dispositivos y conexiones simultáneas.....	75
5.2.5 Estado de APACHE.....	78
5.2.5 Fail2ban.....	79
5.2.6 IPTABLES.....	83
5.3 Otros errores y soluciones.....	84
5.2.1 Error: Webfon muestra Caller ID de la llamada anterior.....	84
5.2.2 Error: serCen no envía correo.....	85
5.2.3 Error al introducir la clave de LDAP en login.....	86
5.2.3 Modificación de serCen en glxcli000013.....	87
6. Arquitectura Gestión multidispositivo.....	88
6.1 Principales elementos funcionales.....	90
7. Pasos para la configuración de Web Call.....	93
7.1 Nodos.....	94
7.2 Plantilla.....	95
7.3 Licencias.....	96
7.4 Extensión física (opcional).....	97
7.5 Extensión Webfon.....	98
7.6 Usuarios.....	103
7.6.1 Validación de usuarios.....	103
7.6.2 Usuario con Extensión Webfon fija.....	106
7.6.3 Usuario con Pool de extensiones Webfon.....	109
7.6.4 Usuario con Doble Factor.....	111
7.6.3 Usuario con Pool de extensiones Webfon.....	114
7.6.5 Usuario con extensión física (multidispositivo).....	117
7.6.5 Usuario con extensión física (multidispositivo).....	120

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



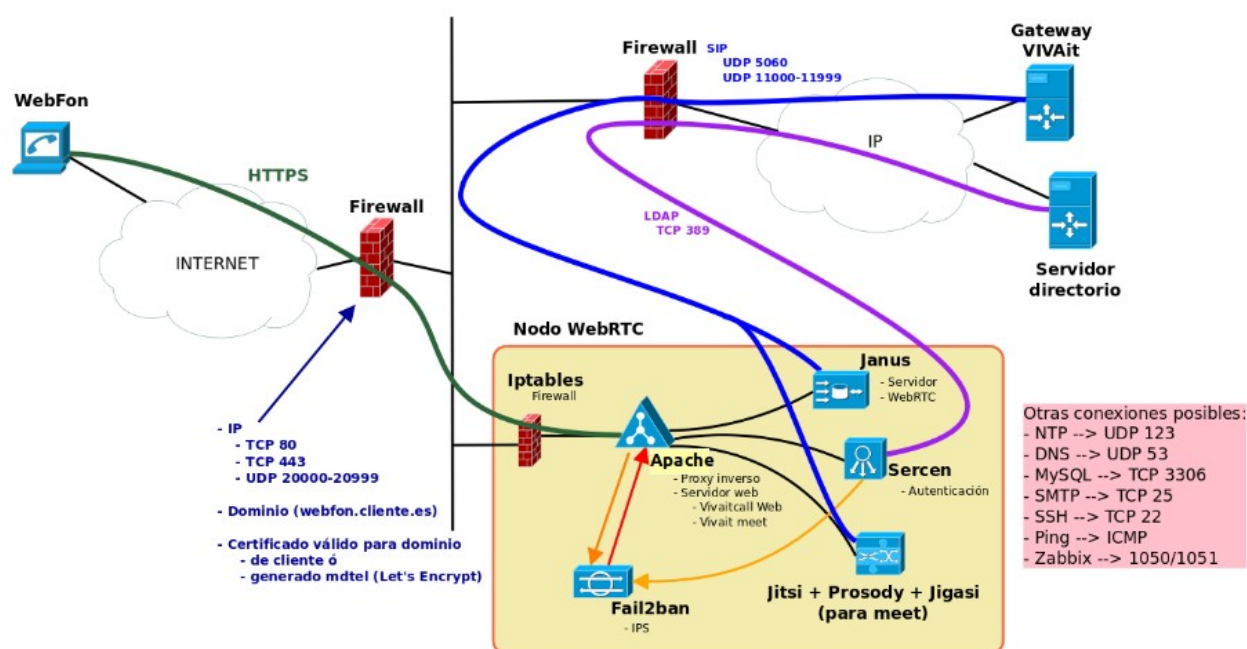
1. Introducción

Este documento tiene como objetivo definir el nodo webRTC de **VIVAit Call**, sus funcionalidades principales, arquitectura, piezas fundamentales, instalación, configuración y diagnósticos.

El nodo Webrtc es el nodo donde se dan de alta los servicios necesarios para la funcionalidad multidispositivo, como son: serCen, janus, apache, FonBo, seguridad y certificados, entre otros.

2. Arquitectura de WebRTC

La arquitectura de WebRTC en **VIVAit Call** es la siguiente:



Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Componentes Principales:

- **Apache**: Apache es un servidor web HTTP de código abierto. La funcionalidad principal de este servicio web es servir a los usuarios todos los ficheros necesarios para visualizar la web. Las solicitudes de los usuarios se hacen normalmente mediante un navegador (Chrome, Firefox, Safari, etc.).
- **SERCEN**: Sirve para identificar a los usuarios y garantizar que los mismos sean quienes dicen ser.
- **JANUS**: *Es un servidor WebRTC de propósito general. Como tal, no proporciona ninguna funcionalidad específica más que implementar la manera que posibilita configurar una comunicación multimedia WebRTC entre un navegador, intercambiando mensajes JSON con él, y la lógica de la aplicación implementada en un servidor, retransmitiendo paquetes RTP / RTCP y mensajes entre navegadores.*
- **IPTABLES**: Es un programa que se encarga de filtrar los paquetes de red, es decir, es la parte que se encarga de determinar qué paquetes de datos queremos que lleguen hasta el servidor y cuáles no.
- **FAIL2BAN**: Es una aplicación de Linux que permite evitar accesos no autorizados al servidor. Funciona bloqueando o baneando (prohibiendo) las IP que realicen varios intentos de acceso incorrectos al servidor según unas reglas previamente configuradas.
- **CLIENTE**: El cliente debe proporcionar una ip pública, un dns para esa ip y un certificado válido.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3. Instalación y configuración del NodoRTC

3.1 Instalación del repositorio

Subir los repositorios a la máquina y configurar los diferentes elementos.

Paso 1	Instalar apache2
Comando	<code>apt install apache2</code>
Resultado esperado	<pre> bartosz@prueba:~\$ sudo apt install apache2 [sudo] contraseña para bartosz: Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes adicionales: apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0 Paquetes sugeridos: apache2-doc apache2-suexec-pristine apache2-suexec-custom Se instalarán los siguientes paquetes NUEVOS: </pre>

Paso 2	Instalar paquete php libapache2
Comando	<code>apt install php libapache2-mod-php</code>
Resultado esperado	<pre> root@prueba:/# apt install php libapache2-mod-php Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes adicionales: apache2 apache2-bin apache2-data apache2-utils libapache2-mod-php7.4 libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0 php7.4 Paquetes sugeridos: apache2-doc apache2-suexec-pristine apache2-suexec-custom Se instalarán los siguientes paquetes NUEVOS: apache2 apache2-bin apache2-data apache2-utils libapache2-mod-php libapache2-mod-php7.4 libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0 php php7.4 0 actualizados, 13 nuevos se instalarán, 0 para eliminar y 0 no actualizados. Se necesita descargar 2.896 kB/3.197 kB de archivos. </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 3	Instalar paquete php-mysql
Comando	<code>apt install php-mysql</code>
Resultado esperado	<pre> root@prueba:/# apt install php-mysql Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes adicionales: php7.4-cli php7.4-common php7.4-curl php7.4-json php7.4-mysql php7.4-opcache php7.4-readline php7.4-xml Se instalarán los siguientes paquetes NUEVOS: php-mysql php7.4-mysql Se actualizarán los siguientes paquetes: php7.4-cli php7.4-common php7.4-curl php7.4-json php7.4-opcache php7.4-readline php7.4-xml 7 actualizados, 2 nuevos se instalarán, 0 para eliminar y 423 no actualizados. Se necesita descargar 2.885 kB de archivos. Se utilizarán 494 kB de espacio de disco adicional después de esta operación. ¿Desea continuar? [S/n] s </pre>

Paso 4	Instalar paquete php-xml
Comando	<code>apt install php-xml</code>
Resultado esperado	<pre> root@prueba:/# apt install php-xml Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho php-xml ya está en su versión más reciente (2:7.4+75). fijado php-xml como instalado manualmente. </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 5	Instalar paquete composer
Comando	<code>apt install composer</code>
Resultado esperado	<pre> root@prueba:/# apt install composer Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes adicionales: git git-man jsonlint libonig5 php-composer-ca-bundle php-composer-semver php-composer-spdx-licenses php-composer-xdebug-handler php-json-schema php-mbstring php-psr-container php-psr-log php-symfony-console php-symfony-filesystem php-symfony-finder php-symfony-process php-symfony-service-contracts php7.4-mbstring Paquetes sugeridos: </pre>
Paso 6	Instalar paquete php-pdo-mysql
Comando	<code>apt install php-pdo-mysql</code>
Resultado esperado	<pre> root@prueba:/# apt install php-pdo-mysql Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Nota, seleccionando «php7.4-mysql» en lugar de «php-pdo-mysql» php7.4-mysql ya está en su versión más reciente (7.4.3-4ubuntu2.12). fijado php7.4-mysql como instalado manualmente. </pre>
Paso 7	Habilitar ssl
Comando	<code>a2enmod ssl</code>
Resultado esperado	<pre> root@prueba:/# a2enmod ssl Considering dependency setenvif for ssl: Module setenvif already enabled Considering dependency mime for ssl: Module mime already enabled Considering dependency socache_shmcb for ssl: Enabling module socache_shmcb. Enabling module ssl. See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates. To activate the new configuration, you need to run: systemctl restart apache2 </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 8	Habilitar proxy http
Comando	<code>a2enmod proxy_http</code>
Resultado esperado	<pre> root@prueba:/# a2enmod proxy_http Considering dependency proxy for proxy_http: Enabling module proxy. Enabling module proxy_http. To activate the new configuration, you need to run: systemctl restart apache2 </pre>

Paso 9	Instalar paquete libmicrohttpd-dev libjansson-dev
Comando	<code>apt install libmicrohttpd-dev libjansson-dev</code>
Resultado esperado	<pre> root@VC-WEB-Debian-V10-02:/home/sat/repositorio_VIVAit_Meet_1.0# apt install libmicrohttpd-dev libjansson-dev Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho libjansson-dev ya está en su versión más reciente (2.12-1build1). fijado libjansson-dev como instalado manualmente. libmicrohttpd-dev ya está en su versión más reciente (0.9.66-1). fijado libmicrohttpd-dev como instalado manualmente. 0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 7 no actualizados. </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 10	Instalar paquete lua-lpty libconfig-dev
Comando	<code>apt install lua-lpty libconfig-dev</code>
Resultado esperado	<pre>root@PVC: ~# ssh -i /home/sat/repositorio_VIVait_Meet_1.0# apt install lua-lpty libconfig-dev Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho libconfig-dev ya está en su versión más reciente (1.5-0.4build1). fijado libconfig-dev como instalado manualmente. Se instalarán los siguientes paquetes NUEVOS: lua-lpty 0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 7 no actualizados. Se necesita descargar 9.744 B de archivos. Se utilizarán 67,6 kB de espacio de disco adicional después de esta operación. ¿Desea continuar? [S/n] s</pre>

Paso 11	Instalar paquete libsofia-sip-ua-dev libglib2.0-dev libldap2-dev
Comando	<code>apt install libsofia-sip-ua-dev libglib2.0-dev libldap2-dev</code>
Resultado esperado	<pre>root@PVC: ~# ssh -i /home/sat/repositorio_VIVait_Meet_1.0# apt install libsofia-sip-ua-dev libglib2.0-dev libldap2-dev Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes adicionales: libblkid-dev libffi-dev libglib2.0-dev-bin libmount-dev libpcrc16-3 libpcrc2-16-0 libpcrc2-32-0 libpcrc2-dev libpcrc2-posix2 libperl libsofia-sip-ua0 uid-dev Paquetes sugeridos: libgirepository1.0-dev libglib2.0-doc libxml2-utils sofia-sip-doc Se instalarán los siguientes paquetes NUEVOS: libblkid-dev libffi-dev libglib2.0-dev libglib2.0-dev-bin libldap2-dev libmount-dev libpcrc16-3 libpcrc2-16-0 libpcrc2-32-0 libpcrc2-dev libpcrc2-posix2 libperl libsofia-sip-ua0 uid-dev 0 actualizados, 19 nuevos se instalarán, 0 para eliminar y 7 no actualizados. Se necesita descargar 5.956 kB de archivos. Se utilizarán 31,4 MB de espacio de disco adicional después de esta operación. ¿Desea continuar? [S/n] s</pre>

Paso 12	Instalar paquete libopus-dev libogg-dev gengetopt
Comando	<code>apt install libopus-dev libogg-dev gengetopt</code>
Resultado esperado	<pre>root@PVC: ~# ssh -i /home/sat/repositorio_VIVait_Meet_1.0# apt install libopus-dev libogg-dev gengetopt Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes NUEVOS: gengetopt libogg-dev libopus-dev 0 actualizados, 3 nuevos se instalarán, 0 para eliminar y 7 no actualizados. Se necesita descargar 567 kB de archivos. Se utilizarán 2.020 kB de espacio de disco adicional después de esta operación. Des:1 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 gengetopt amd64 2.23+dfsg1-1 [159 kB] Des:2 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libogg-dev amd64 1.3.4-0ubuntu1 [161 kB] Des:3 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libopus-dev amd64 1.3.1-0ubuntu1 [236 kB] Descargados 567 kB en 0s (2.060 kB/s) Seleccionando el paquete gengetopt previamente no seleccionado.</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 13	Instalar paquete libavutil-dev libavformat-dev libavcodec-dev
Comando	<code>apt install libavutil-dev libavformat-dev libavcodec-dev</code>
Resultado esperado	<pre>root@sat: ~# cd /home/sat/repositorio_VIVAit_Meet_1.0# apt install libavutil-dev libavformat-dev libavcodec-dev Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes adicionales: libswresample-dev Se instalarán los siguientes paquetes NUEVOS: libavcodec-dev libavformat-dev libavutil-dev libswresample-dev 0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 7 no actualizados. Se necesita descargar 7.033 kB de archivos. Se utilizarán 29,2 MB de espacio de disco adicional después de esta operación. ¿Desea continuar? [S/n] s</pre>

Paso 14	Instalar paquete libwebsockets-dev libusrctp-dev
Comando	<code>apt install libwebsockets-dev libusrctp-dev</code>
Resultado esperado	<pre>root@sat: ~# cd /home/sat/repositorio_VIVAit_Meet_1.0# apt install libwebsockets-dev libusrctp-dev Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes adicionales: libcap-dev libev-dev libev4 libusrctp1 libuv1-dev libwebsockets15 Se instalarán los siguientes paquetes NUEVOS: libcap-dev libev-dev libev4 libusrctp-dev libusrctp1 libuv1-dev libwebsockets-dev libwebsockets15 0 actualizados, 6 nuevos se instalarán, 0 para eliminar y 7 no actualizados. Se necesita descargar 1.331 kB de archivos. Se utilizarán 4.984 kB de espacio de disco adicional después de esta operación. ¿Desea continuar? [S/n] s</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3.2 Configuración de los diferentes elementos

El primer paso es subir los archivos que se necesitan a la máquina (**install-webfon.tar: en el manual está nombrado como "instal" así que conviene renombrar**). No es necesario subirlos a una carpeta específica (esos archivos se moverán a otros directorios)

Enlace para descargar "install-webfon.tar" :

<http://viva-gestion00.mdnova.local/redmine/documents/223>

3.2.1 Configuración de libnice

Paso 1	Entrar en el directorio <code>/usr/lib/x86_64-linux-gnu :</code>
Comando	<code>cd /usr/lib/x86_64-linux-gnu</code>
Resultado esperado	<pre>~# cd /usr/lib/x86_64-linux-gnu/ ~/usr/lib/x86_64-linux-gnu#</pre>

Paso 2	Cambiar nombre del directorio a <code>x86_64-linux-gnu</code>
Comando	<code>mv -vi /root/instal/usr/lib/x86_64-linux-gnu/libnice.so.10.11.0 /usr/lib/x86_64-linux-gnu/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# mv -vi /root/instal/usr/lib/x86_64-linux-gnu/libnice.so.10.11.0 /usr/lib/x86_64-linux-gnu/ renamed '/root/instal/usr/lib/x86_64-linux-gnu/libnice.so.10.11.0' -> '/usr/lib/x86_64-linux-gnu/libnice.so.10.11.0'</pre>

Paso 3	Entrar en el directorio donde hemos copiado el fichero <code>libnice.so</code>
Comando	<code>cd /usr/lib/x86_64-linux-gnu/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# cd /usr/lib/x86_64-linux-gnu/ root@VC-WebP-AytoArganda-MAD-02:/usr/lib/x86_64-linux-gnu#</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 4	Crear enlaces simbólicos
Comando	<code>ln -svi libnice.so.10.11.0 libnice.so.10</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/usr/lib/x86_64-linux-gnu# ln -svi libnice.so.10.11.0 libnice.so.10 'libnice.so.10' -> 'libnice.so.10.11.0'</pre>

Paso 5	Crear enlaces simbólicos
Comando	<code>ln -svi libnice.so.10 libnice.so</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/usr/lib/x86_64-linux-gnu# ln -svi libnice.so.10 libnice.so 'libnice.so' -> 'libnice.so.10'</pre>

Paso 6	Verificar los enlaces simbólicos
Comando	<code>ls -ltr</code>
Resultado Esperado	<pre>root@vivait-meet-install:/usr/lib/x86_64-linux-gnu# ls -ltr grep libnice -rwxr-xr-x 1 root root 1708776 nov 26 2020 libnice.so.10.11.0 lrwxrwxrwx 1 root root 18 abr 6 2021 libnice.so.10 -> libnice.so.10.11.0 lrwxrwxrwx 1 root root 13 abr 6 2021 libnice.so -> libnice.so.10 root@vivait-meet-install:/usr/lib/x86_64-linux-gnu#</pre>

Configuración del directorio /var/www

Paso 1	Entrar en el directorio /www
Comando	<code>cd /var/www</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/usr/lib/x86_64-linux-gnu# cd /var/www root@VC-WebP-AytoArganda-MAD-02:/var/www#</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 2	Eliminar la carpeta /var/www/html
Comando	<code>rm -R html</code>
Resultado esperado	<pre> root@VC-WebP-AytoArganda-MAD-02:/var/www# rm -R html/ root@VC-WebP-AytoArganda-MAD-02:/var/www# root@VC-WebP-AytoArganda-MAD-02:/var/www# root@VC-WebP-AytoArganda-MAD-02:/var/www# root@VC-WebP-AytoArganda-MAD-02:/var/www# root@VC-WebP-AytoArganda-MAD-02:/var/www# ls -l total 0 drwxr-xr-x 2 root root 4096 feb 1 14:08 html </pre>

Paso 3	Mover fichero html a /www
Comando	<code>mv -vi /root/instal/var/www/html</code>
Resultado esperado	<pre> root@VC-WebP-AytoArganda-MAD-02:/var/www# mv -vi /root/instal/var/www/html . renamed '/root/instal/var/www/html' -> './html' root@VC-WebP-AytoArganda-MAD-02:/var/www# root@VC-WebP-AytoArganda-MAD-02:/var/www# root@VC-WebP-AytoArganda-MAD-02:/var/www# root@VC-WebP-AytoArganda-MAD-02:/var/www# root@VC-WebP-AytoArganda-MAD-02:/var/www# ls -l total 4 drwxr-xr-x 2 root root 4096 feb 1 14:08 html </pre>

Paso 4	Mover el directorio vivait a /var/www/vivait/webs
Comando	<code>mv -vi /root/instal/var/www/vivait/webs</code>
Resultado esperado	<pre> root@VC-WebP-AytoArganda-MAD-02:/var/www# mv -vi /root/instal/var/www/vivait . renamed '/root/instal/var/www/vivait' -> './vivait' </pre>

Paso 5	Crear directorio Mdtel
Comando	<code>mkdir /etc/Mdtel/</code>
Resultado esperado	<pre> root@VC-WebP-AytoArganda-MAD-02:~# mkdir /etc/Mdtel root@VC-WebP-AytoArganda-MAD-02:~# </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3.2.2 Configuración de serCen

Paso 1	Crear el grupo sercen
Comando	<code>addgroup --system --gid 990 sercen</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# addgroup --system --gid 990 sercen Adding group 'sercen' (GID 990) ... Done.</pre>

Paso 2	Crear el usuario sercen
Comando	<code>sudo adduser --system --shell /usr/sbin/nologin --uid 990 --gid 990 --home /nonexistent --no-create-home sercen</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# adduser --system --shell /usr/sbin/nologin --uid 990 --gid 990 --home /nonexistent --no-create-home sercen Warning: The home dir /nonexistent you specified can't be accessed: No such file or directory Adding system user 'sercen' (UID 990) ... Adding new user 'sercen' (UID 990) with group 'sercen' ... Not creating home directory '/nonexistent'.</pre>

Paso 3	Mover el fichero serCen a /etc/sbin
Comando	<code>mv -vi /instal/SerCen/usr/sbin/serCen /usr/sbin/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# mv -vi /root/instal/serCen/usr/sbin/serCen /usr/sbin/ renamed '/root/instal/serCen/usr/sbin/serCen' -> '/usr/sbin/serCen'</pre>

Paso 4	Mover el fichero serCen a /etc/init.d
Comando	<code>mv -vi /instal/SerCen/etc/init.d/serCen /etc/init.d/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# mv -vi /root/instal/serCen/etc/init.d/serCen /etc/init.d/ renamed '/root/instal/serCen/etc/init.d/serCen' -> '/etc/init.d/serCen'</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 5	Mover el fichero serCen a /etc/logrotate.d
Comando	<code>mv -vi /instal/SerCen/etc/logrotate.d/serCen /etc/logrotate.d/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# mv -vi /root/instal/serCen/etc/logrotate.d/serCen /etc/logrotate.d/ renamed '/root/instal/serCen/etc/logrotate.d/serCen' -> '/etc/logrotate.d/serCen'</pre>

Paso 6	Mover el fichero serCen.conf a /etc/Mdtel/
Comando	<code>mv -vi /instal/SerCen/etc/MDtel/serCen.conf /etc/MDtel/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# mv -vi /root/instal/serCen/etc/MDtel/serCen.conf /etc/MDtel/ renamed '/root/instal/serCen/etc/MDtel/serCen.conf' -> '/etc/MDtel/serCen.conf'</pre>

Paso 7	Entrar en el directorio /etc/rc2.d
Comando	<code>cd /etc/rc2.d</code>
Resultado esperado	<pre>rc0.d/ rc1.d/ rc2.d/ rc3.d/ rc4.d/ rc5.d/ rc6.d/ rcS.d/ root@VC-WebP-AytoArganda-MAD-02:~# cd /etc/rc2.d/</pre>

Paso 8	Crear enlace simbolico de arranque
Comando	<code>ln -s ../init.d/serCen S10serCen</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/rc2.d# ln -s ../init.d/serCen S10serCen</pre>

Paso 9	Dar permisos de ejecución a /usr/sbin/serCen
Comando	<code>chmod +x /usr/sbin/serCen</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/usr/sbin# chmod +x serCen root@VC-WebP-AytoArganda-MAD-02:/usr/sbin#</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 10	Dar permisos de ejecución a /etc/init.d/serCen
Comando	<code>chmod +x /etc/init.d/serCen</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# chmod +x serCen root@VC-WebP-AytoArganda-MAD-02:/etc/init.d#</pre>

Paso 11	Dar permisos de ejecución a /etc/logrotate/serCen
Comando	<code>chmod +x /etc/logrotate/serCen</code>
Resultado esperado	<pre>chmod +x serCen</pre>

Modificar el fichero SerCen.conf

Hay que revisar ciertos parámetros y modificar los que hagan falta dependiendo de las características del cliente.

Paso 12	Revisar y modificar el fichero /etc/MDtel/serCen.conf
Comando	<code>vi /etc/MDtel/serCen.conf</code>

En el fichero se tienen que comprobar y o modificar los siguientes parámetros:

```
smtps_usa = false
smtp_host = "correoejemplo.mdnova.local"
smtp_puerto = 0
smtp_usuario = ""
smtp_clave = ""
```

En la línea de f1factor_xx si se pone una IP, no pide pin de acceso para esa IP. Esta característica depende del cliente (Sí el cliente no quiere que se le pida el doble factor habría que poner la red del cliente, si no dejar por defecto en blanco). Si se deja en blanco, pedirá doble factor

Ejemplo:

```
f1factor_00_ip = "172.25.0.0"
f1factor_00_msk = "255.255.0.0"
```

```
f1factor_01_ip = ""
f1factor_01_msk = ""
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Los siguientes parámetros definen el nombre de las salas con posibilidad de acceso telefónico:

```
vivait_meet_prefijo_min = 5
vivait_meet_prefijo = "vivaitcall_"
```

Parámetro VDN para acceder a la sala por teléfono:

```
vivait_meet_dnis = "9999"
```

Número de días en que una clave expirada puede ser cambiada antes de su caducidad. Un valor de 0 indica que no caduca.

```
.. ws_implementacion / clave_caduca_dias.
```

Paso 13	Arrancar serCen
Comando	<code>/etc/init.d/serCen start</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# /etc/init.d/serCen start Starting serCen</pre>

Paso 14	Comprobar que SerCen está arrancado
Comando	<code>/etc/init.d/serCen status</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# /etc/init.d/serCen status serCen is running</pre>

Paso 15	Comprobar que SerCen está funcionando correctamente
Comando	<code>nc localhost 1125</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# nc localhost 1125 serCen: sis ver='00:01:04' inic='20220202 101100' alarmas=0 ultAlar='00000000 000000' serCen: gmp msj=254/256 buf=256/256 tarea=97/102 serCen: tmp uptime=44 (0d 0h 0m 44s) serCen: wvs mysql=1 conxNum=0 conxMaxPeriodo=0 serCen: wvc numCacheLibre=3 numColaPend=0 serCen: wpc numReq=0/0 numGet=0/0 numPost=0/0 numPut=0/0 numDelete=0/0 serCen: smt numCacheLibre=5 numColaPend=0 enPeriodo=0/0 numMsj=0/0</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3.2.3 Configuración de Janus

Paso 1	Crear el grupo de janus
Comando	<code>addgroup --system --gid 991 janus</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# addgroup --system --gid 991 janus Adding group 'janus' (GID 991) ... Done.</pre>
Paso 2	Crear el usuario janus
Comando	<code>sudo adduser --system --shell /usr/sbin/nologin --uid 991 --gid 991 --home /nonexistent --no-create-home janus</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# adduser --system --shell /usr/sbin/nologin --uid 991 --gid 991 --home /nonexistent --no-create-home janus Warning: The home dir /nonexistent you specified can't be accessed: No such file or directory Adding system user 'janus' (UID 991) ... Adding new user 'janus' (UID 991) with group 'janus' ...</pre>
Paso 3	Mover directorio scripts a /root
Comando	<code>mv -vi /root/instal/scripts /root/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# mv -vi /root/instal/scripts /root/ renamed '/root/instal/scripts' -> '/root/scripts'</pre>
Paso 4	Mover vivait-meet/opt a /opt/
Comando	<code>mv -vi /instal/vivait-meet/opt /opt/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# mv -vi /root/instal/vivait-meet/opt /opt/ renamed '/root/instal/vivait-meet/opt' -> '/opt/opt'</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 5	Mover janus a /etc/init.d
Comando	<code>mv -vi /instal/janus/etc/init.d/janus /etc/init.d/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# mv -vi /root/instal/janus/etc/init.d/janus /etc/init.d/ renamed '/root/instal/janus/etc/init.d/janus' -> '/etc/init.d/janus'</pre>

Paso 6	Mover janus a /etc/
Comando	<code>mv -vi /instal/janus/etc/janus /etc/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# mv -vi /root/instal/janus/etc/janus /etc/ renamed '/root/instal/janus/etc/janus' -> '/etc/janus'</pre>

Paso 7	Mover janus a /etc/logrotate.d
Comando	<code>mv -vi /instal/janus/etc/logrotate.d/janus /etc/logrotate.d/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# mv -vi /root/instal/janus/etc/logrotate.d/janus /etc/logrotate.d/ renamed '/root/instal/janus/etc/logrotate.d/janus' -> '/etc/logrotate.d/janus'</pre>

Paso 8	Mover lib a /usr/lib
Comando	<code>mv -vi /instal/janus/usr/lib/ /usr/lib/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# mv -vi /root/instal/janus/usr/lib /usr/lib/ renamed '/root/instal/janus/usr/lib' -> '/usr/lib/lib'</pre>

Paso 9	Mover janus a /usr/sbin
Comando	<code>mv -vi /instal/janus/usr/sbin/janus /usr/sbin/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# mv -vi /root/instal/janus/usr/sbin/janus /usr/sbin/ renamed '/root/instal/janus/usr/sbin/janus' -> '/usr/sbin/janus'</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 10	Mover safe_janus a /usr/sbin
Comando	<code>mv -vi /instal/janus/usr/sbin/safe_janus /usr/sbin/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# mv -vi /root/instal/janus/usr/sbin/janus /usr/sbin/ renamed '/root/instal/janus/usr/sbin/janus' -> '/usr/sbin/janus'</pre>
Paso 11	Dar permisos de ejecución /usr/sbin/janus
Comando	<code>chmod +x janus</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/usr/sbin# chmod +x janus root@VC-WebP-AytoArganda-MAD-02:/usr/sbin#</pre>
Paso 12	Dar permisos de ejecución /usr/sbin/janus
Comando	<code>chmod +x /usr/sbin/safe_janus</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/usr/sbin# chmod +x safe_janus root@VC-WebP-AytoArganda-MAD-02:/usr/sbin#</pre>
Paso 13	Entrar en el directorio /etc/rc2.d
Comando	<code>cd /etc/rc2.d</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/usr/sbin# cd /etc/rc2.d/ root@VC-WebP-AytoArganda-MAD-02:/etc/rc2.d#</pre>
Paso 14	Crear enlace simbólico entre dos ficheros
Comando	<code>ln -s ../init.d/janus S10janus</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/rc2.d# ln -s ../init.d/janus S10janus root@VC-WebP-AytoArganda-MAD-02:/etc/rc2.d#</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Modificar el fichero janus.jcfg

Hay que revisar ciertos parámetros y modificar los que hagan falta dependiendo de las características del cliente.

Paso 15	Revisar y modificar el fichero /etc/janus.jcfg
Comando	<code>vi /etc/janus/janus.jcfg</code>

En el fichero se tienen que comprobar y o modificar los siguientes parámetros:

`server_name = "labmeet.mdnova.local"` → **Sustituir todos los "labmeet.mdnova.local" por nombre de dominio del servidor**

```
certificates: {
    # dtls
    cert_pem = "" → Dejar en blanco
    cert_key = "" → Dejar en blanco
```

Modificar el fichero janus.consola.jcfg

Paso 16	Modificar el fichero /etc/janus/janus.consola.jcfg
Comando	<code>vi /etc/janus/janus.consola.jcfg</code>

REVISAR:

`server_name = "labmeet.mdnova.local"` → **Sustituir todos los "labmeet.mdnova.local" por nombre de dominio del servidor**

```
certificates: {
    # dtls
    cert_pem = "" → Dejar en blanco
    cert_key = "" → Dejar en blanco
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Modificar el fichero janus.transport.http.jcfg

Paso 17	Modificar el fichero janus.transport.http.jcfg
Comando	<code>vi /etc/janus/janus.transport.http.jcfg</code>

REVISAR:

cert_pem = "" → **Dejar en blanco**
cert_key = "" → **Dejar en blanco**

Modificar el fichero janus.consola.jcfg

Paso 18	Modificar el fichero janus.transport.websockets.jcfg
Comando	<code>vi /etc/janus/janus.transport.websockets.jcfg</code>

REVISAR:

cert_pem = "" → **Dejar en blanco**
cert_key = "" → **Dejar en blanco**

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Modificar el fichero vivait.plugin.webfon.jcfg

Paso 19	Modificar el fichero vivait.plugin.webfon.jcfg
Comando	<code>vi /etc/janus/vivait.plugin.webfon.jcfg</code>

REVISAR:

`local_nodo_id = 4` → **ID NODO EN LA BASE DE DATOS EN LA QUE ESTE CREADO EL WEBPHONE *******

`colas_limpiar_siempre = false` → **SI SE TRABAJA CON POOL DE EXTENSIONES A TRUE**

`local_ip = "172.25.0.22"` → **IP PRIVADA EN LA RED DE VOZ**

`max_digi_llam_sal = 10` → **NÚMERO MÁXIMO DE DÍGITOS EN LLAMADAS SALIENTES DE LA EXTENSIÓN**



Se configura `max_digi_llam_sal` porque son pequeños mecanismos de seguridad por si un cliente solo va a hacer llamadas nacionales , para evitar que no haga llamadas internacionales.

Paso 20	Parar el servicio janus
Comando	<code>/etc/init.d/janus stop</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/usr/lib/vivait# /etc/init.d/janus stop Stopping janus: janus.</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 21	Arrancar janus
Comando	<code>/etc/init.d/janus start</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/rc2.d# /etc/init.d/janus start Starting janus: janus.</pre>

Paso 22	Comprobar que janus se está ejecutando
Comando	<code>/etc/init.d/janus status</code>
Resultado esperado	<pre>root@vivait-meet-install:/etc/janus# /etc/init.d/janus status Status of janus: janus is running.</pre>

Paso 23	Comprobar que janus funciona correctamente
Comando	<code>ps aux grep janus</code>
Resultado esperado	<pre>Starting janus: janus. root@VC-WebP-AytoArganda-MAD-02:~/install-vebfon/janus/usr/sbin# ps aux grep janus root 3111121 0.0 0.0 6892 1708 ? S 10:43 0:00 /bin/bash /usr/sbin/safe_janus root 3111128 0.0 0.0 7996 3260 ? S 10:43 0:00 runuser -u janus -g janus -- nice -n 0 /usr/sbin/janus --pid-file=/var/run/janus/janus.pid --config-folder=/etc/janus --log-file=/var/log/janus.log janus 3111129 0.6 0.2 1106928 13284 ? Sl 10:43 0:00 /usr/sbin/janus --pid-file=/var/run/janus/janus.pid --config-folder=/etc/janus --log-file=/var/log/janus/janus.log root 3111154 0.0 0.0 6432 672 pts/1 S+ 10:43 0:00 grep --color=auto janus</pre>

*ps aux devuelve la línea de comandos completa de cada proceso de janus

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3.2.4 Configuración de APACHE

Paso 1	Entrar en el directorio <code>/etc/apache2/sites-available</code>
Comando	<code>cd /etc/apache2/sites-available/</code>

Modificar el fichero labmeet2.mdnova.local.conf

Paso 2	Modificar fichero <code>/etc/apache2/sites-available/labmeet2.mdnova.local.conf</code>
Comando	<code>vi labmeet2.mdnova.local.conf</code>
Sustituir en el fichero las siguientes líneas:	<pre> <IfModule mod_ssl.c> <VirtualHost _default_:443> #ServerAdmin webmaster@localhost DocumentRoot /var/www/vivait ServerName labmeet2.mdnova.local # Available loglevels: trace8, ..., trace1, debug, info, notice, warn, # error, crit, alert, emerg. # It is also possible to configure the loglevel for particular # modules, e.g. #LogLevel info ssl:warn ErrorLog \${APACHE_LOG_DIR}/error.log CustomLog \${APACHE_LOG_DIR}/access.log combined # For most configuration files from conf-available/, which are # enabled or disabled at a global level, it is possible to </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
```

```
# SSL Engine Switch:
# Enable/Disable SSL for this virtual host.
SSLEngine on
```

```
# A self-signed (snakeoil) certificate can be created by installing
# the ssl-cert package. See
# /usr/share/doc/apache2/README.Debian.gz for more info.
# If both key and certificate are stored in the same file, only the
# SSLCertificateFile directive is needed.
SSLCertificateFile /etc/apache2/ssl/labmeet2.mdnova.local.crt
SSLCertificateKeyFile /etc/apache2/ssl/labmeet2.mdnova.local.key
```

```
# Server Certificate Chain:
# Point SSLCertificateChainFile at a file containing the
# concatenation of PEM encoded CA certificates which form the
# certificate chain for the server certificate. Alternatively
# the referenced file can be the same as SSLCertificateFile
# when the CA certificates are directly appended to the server
# certificate for convinience.
#SSLCertificateChainFile /etc/apache2/ssl.crt/server-ca.crt
```

```
# Certificate Authority (CA):
# Set the CA certificate verification path where to find CA
# certificates for client authentication or alternatively one
# huge file containing all of them (file must be PEM encoded)
# Note: Inside SSLCACertificatePath you need hash symlinks
# to point to the certificate files. Use the provided
# Makefile to update the hash symlinks after changes.
#SSLCACertificatePath /etc/ssl/certs/
#SSLCACertificateFile /etc/apache2/ssl.crt/ca-bundle.crt
```

```
# Certificate Revocation Lists (CRL):
# Set the CA revocation path where to find CA CRLs for client
# authentication or alternatively one huge file containing all
# of them (file must be PEM encoded)
# Note: Inside SSLCARevocationPath you need hash symlinks
# to point to the certificate files. Use the provided
# Makefile to update the hash symlinks after changes.
#SSLCARevocationPath /etc/apache2/ssl.crl/
#SSLCARevocationFile /etc/apache2/ssl.crl/ca-bundle.crl
```

```
# Client Authentication (Type):
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```
# Client certificate verification type and depth. Types are
# none, optional, require and optional_no_ca. Depth is a
# number which specifies how deeply to verify the certificate
# issuer chain before deciding the certificate is not valid.
#SSLVerifyClient require
#SSLVerifyDepth 10

# SSL Engine Options:
# Set various options for the SSL engine.
# o FakeBasicAuth:
# Translate the client X.509 into a Basic Authorisation. This means that
# the standard Auth/DBMAuth methods can be used for access control. The
# user name is the 'one line' version of the client's X.509 certificate.
# Note that no password is obtained from the user. Every entry in the user
# file needs this password: 'xxj31ZMTZzkVA'.
# o ExportCertData:
# This exports two additional environment variables: SSL_CLIENT_CERT and
# SSL_SERVER_CERT. These contain the PEM-encoded certificates of the
# server (always existing) and the client (only existing when client
# authentication is used). This can be used to import the certificates
# into CGI scripts.
# o StdEnvVars:
# This exports the standard SSL/TLS related 'SSL_*' environment variables.
# Per default this exportation is switched off for performance reasons,
# because the extraction step is an expensive operation and is usually
# useless for serving static content. So one usually enables the
# exportation for CGI and SSI requests only.
# o OptRenegotiate:
# This enables optimized SSL connection renegotiation handling when SSL
# directives are used in per-directory context.
#SSLOptions +FakeBasicAuth +ExportCertData +StrictRequire

ProxyPass "/sercen" "http://127.0.0.1:7669/sercen"
ProxyPass "/janus" "http://127.0.0.1:8088/janus"
ProxyPass "/Vivait-FonBO" "http://FONBO:8180/Vivait-FonBO"

<FilesMatch "\.(cgi|shtml|phtml|php)$">
    SSLOptions +StdEnvVars
</FilesMatch>
<Directory /usr/lib/cgi-bin>
    SSLOptions +StdEnvVars
</Directory>

# SSL Protocol Adjustments:
# The safe and default but still SSL/TLS standard compliant shutdown
# approach is that mod_ssl sends the close notify alert but doesn't wait for
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```
# the close notify alert from client. When you need a different shutdown
# approach you can use one of the following variables:
# o ssl-unclean-shutdown:
#   This forces an unclean shutdown when the connection is closed, i.e. no
#   SSL close notify alert is send or allowed to received. This violates
#   the SSL/TLS standard but is needed for some brain-dead browsers. Use
#   this when you receive I/O errors because of the standard approach where
#   mod_ssl sends the close notify alert.
# o ssl-accurate-shutdown:
#   This forces an accurate shutdown when the connection is closed, i.e. a
#   SSL close notify alert is send and mod_ssl waits for the close notify
#   alert of the client. This is 100% SSL/TLS standard compliant, but in
#   practice often causes hanging connections with brain-dead browsers. Use
#   this only for browsers where you know that their SSL implementation
#   works correctly.
# Notice: Most problems of broken clients are also related to the HTTP
# keep-alive facility, so you usually additionally want to disable
# keep-alive for those clients, too. Use variable "nokeepalive" for this.
# Similarly, one has to force some clients to use HTTP/1.0 to workaround
# their broken HTTP/1.1 implementation. Use variables "downgrade-1.0" and
# "force-response-1.0" for this.
# BrowserMatch "MSIE [2-6]" \
#     nokeepalive ssl-unclean-shutdown \
#     downgrade-1.0 force-response-1.0

</VirtualHost>
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Paso 3	Entrar en el directorio /etc/apache2/sites-enabled
Comando	<code>cd /etc/apache2/sites-enabled/</code>
Paso 4	Crear enlace simbólico
Comando	<code>ln -s ../sites-available/labmeet2.mdnova.local.conf labmeet2.mdnova.local.conf</code>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 5	Parar apache
Comando	<code>systemctl stop apache2</code>

Paso 6	Iniciar apache
Comando	<code>systemctl start apache2</code>

Paso 7	Comprobar funcionamiento de apache
Comando	<code>systemctl status apache2</code>

3.2.5 Instalación y configuración de Baikal

Para definir las programaciones de desvíos se ha utilizado una solución de servidor estándar de calendarios; se ha optado por el servidor Baikal, por disponer de todas las funcionalidades necesarias y disponer de BBDD Mysql (especialmente adecuado en entornos [VIVAit](#)).

Paso 1	Instalar php 7.4
Comando	<code>apt install php7.4-curl</code>

Paso 2	Entrar en el directorio /tmp
Comando	<code>cd /tmp</code>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 3	Descargar el archivo comprimido de Baikal
Comando	<code>wget https://github.com/sabre-io/Baikal/archive/refs/tags/0.9.2.tar.gz</code>
Resultado esperado	<pre> root@prueba:/tmp# wget https://github.com/sabre-io/Baikal/archive/refs/tags/0.9.2.tar.gz --2022-07-05 12:05:21-- https://github.com/sabre-io/Baikal/archive/refs/tags/0.9.2.tar.gz Resolviendo github.com (github.com)... 140.82.121.4 Conectando con github.com (github.com)[140.82.121.4]:443... conectado. Petición HTTP enviada, esperando respuesta... 302 Found Ubicación: https://codeload.github.com/sabre-io/Baikal/tar.gz/refs/tags/0.9.2 [siguiente] --2022-07-05 12:05:21-- https://codeload.github.com/sabre-io/Baikal/tar.gz/refs/tags/0.9.2 Resolviendo codeload.github.com (codeload.github.com)... 140.82.121.10 Conectando con codeload.github.com (codeload.github.com)[140.82.121.10]:443... conectado. Petición HTTP enviada, esperando respuesta... 200 OK Longitud: no especificado [application/x-gzip] Guardando como: "0.9.2.tar.gz" 0.9.2.tar.gz [<=>] 556,96K 1,76MB/s en 0,3s 2022-07-05 12:05:22 (1,76 MB/s) - "0.9.2.tar.gz" guardado [570329] </pre>
Paso 4	Descomprimir el archivo que se ha descargado
Comando	<code>tar xvzf 0.9.2.tar.gz Baikal-0.9.2/</code>
Resultado esperado	<pre> root@prueba:/tmp# tar xvzf 0.9.2.tar.gz Baikal-0.9.2/ Baikal-0.9.2/ Baikal-0.9.2/.github/ Baikal-0.9.2/.github/ISSUE_TEMPLATE/ Baikal-0.9.2/.github/ISSUE_TEMPLATE/bug_report.md Baikal-0.9.2/.github/ISSUE_TEMPLATE/feature_request.md Baikal-0.9.2/.github/ISSUE_TEMPLATE/upgrading.md Baikal-0.9.2/.github/workflows/ Baikal-0.9.2/.github/workflows/ci.yml Baikal-0.9.2/.gitignore </pre>
Paso 5	Mover la carpeta de Baikal a /var/www/vivait
Comando	<code>mv Baikal-0.9.2 /var/www/vivait</code>
Paso 6	Entrar en el directorio /var/www/vivait
Comando	<code>cd /var/www/vivait</code>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 7	Renombrar Baikal-0.9.2 a baikal
Comando	<code>mv Baikal-0.9.2 baikal</code>

Paso 8	Asignar permisos a la carpeta baikal
Comando	<code>chown -R www-data:www-data baikal</code>

Paso 9	Entrar en el directorio <code>/var/www/vivait/baikal</code>
Comando	<code>cd baikal</code>

Paso 10	Instalar composer en el directorio baikal
Comando	<code>composer install</code>
Resultado esperado	<pre> root@prueba:/var/www/vivait/baikal# composer install Do not run Composer as root/super user! See https://getcomposer.org/root for details Loading composer repositories with package information Warning from https://repo.packagist.org: Support for Composer 1 is deprecated and some packages will not be available. You should upgrade to Composer 2. See https://blog.packagist.com/deprecating-composer-1-support/ info from https://repo.packagist.org: #StandWithUkraine Updating dependencies (including require-dev) Package operations: 39 installs, 0 updates, 0 removals </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 11	Cargar repositorios de composer en el directorio baikal
Comando	<code>composer fund</code>
Resultado esperado	<pre> root@prueba:/var/www/vivait/baikal# composer fund Do not run Composer as root/super user! See https://getcomposer.org/root for details Warning from https://repo.packagist.org: Support for Composer 1 is deprecated and some packages will not be available. You should upgrade to Composer 2. See https://blog.packagist.com/deprecating-composer-1-support/ Info from https://repo.packagist.org: #StandWithUkraine The following packages were found in your dependencies which publish funding information: composer pcre, semver, xdebug-handler https://packagist.com https://github.com/sponsors/composer https://tidelift.com/funding/github/packagist/composer/composer </pre>
Paso 12	Entrar en el directorio /var/www/vivait/baikal/vendor/sabre/dav/lib/CalDAV/
Comando	<code>cd /var/www/vivait/baikal/vendor/sabre/dav/lib/CalDAV/</code>
Paso 13	Crear un archivo de respaldo de Plugin.php
Comando	<code>mv -vi Plugin.php Plugin.php.orig</code>
Paso 14	Mover el archivo Plugin.php a /tmp "cogerlo de http://viva-gestion00/redmine/attachments/10508"
Comando	<code>mv /tmp/Plugin.php .</code>
Paso 15	Conceder permisos al archivo Plugin.php
Comando	<code>chown root:root /tmp/Plugin.php</code>
Paso 16	Conceder otros permisos al archivo Plugin.php
Comando	<code>chmod 644 /tmp/Plugin.php</code>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 17	Acceder a mysql (Pedirá contraseña del usuario root)
Comando	<code>mysql -u root -p</code>

Paso 18	Crear una base de datos con nombre baikal
Comando	<code>create database baikal;</code>

Paso 19	Crear usuario baikal en mysql
Comando	<code>create user 'baikal'@'localhost' identified by 'ivivabaikal';</code>

Paso 20	Conceder privilegios al usuario baikal de mysql
Comando	<code>grant all on baikal.* to 'baikal'@'localhost';</code>

3.2.6 Instalación y configuración del certificado

Un certificado digital no es más que una cadena de bits con la clave pública y una serie de atributos, todo, firmado digitalmente por una autoridad de certificación (CA) que actúa de garante de la información que contiene. Cuando la información ha sido verificada por la CA, utiliza su clave privada para firmar digitalmente el certificado emitido, lo que proporciona confianza a quien lo recibe.

La clave privada es un archivo electrónico con extensión KEY. Este mecanismo se compone de un software que sirve para el desarrollo de mecanismos de encriptación como los cifrados.

Existen diferentes extensiones en los certificados, estos pueden ser:

- .pfx: es la copia de seguridad con clave privada de un certificado (exportado desde Internet Explorer).

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



- .p12: es la copia de seguridad con clave privada de un certificado (exportado desde Firefox).
- .cer: es un formato de exportación de clave pública desde Internet Explorer, puede ser en formato DER o formato PEM (Base64)
- .crt: es un formato de exportación de clave pública desde Mozilla Firefox. Es en formato PEM (Base 64).

Es importante tener en cuenta que en cuanto a los certificados SSL/TLS, no importa la extensión del archivo. El certificado SSL/TLS puede estar colocado en forma textual (que predomina) es lo más habitual (Linux y apache, Unix y otros servidores) o en forma binaria (Java, Microsoft Server).

Dependiendo del cliente, la forma de instalar y configurar el certificado puede variar. A continuación se muestran los pasos para configurar un certificado con extensión .pfx :

Instalación de un certificado con extensión .pfx

Lo primero es obtener los certificados y copiarlos en una carpeta de la máquina.

Paso 1	Generar clave.key
Comando	<code>openssl pkcs12 -in CertWebphone.pfx -nocerts -out Clave.key</code>
Resultado esperado	<pre> root@VC-WebP-AytoArganda-MAD-02:~# openssl pkcs12 -in CertWebPhone.pfx -nocerts -out Clave.key Enter Import Password: Enter PEM pass phrase: Verifying - Enter PEM pass phrase: </pre>

Paso 2	Generar certificado CertWebphone.pfx
Comando	<code>openssl pkcs12 -in CertWebphone.pfx -cicerts -nokeys -out Certificado.crt</code>
Resultado esperado	<pre> root@VC-WebP-AytoArganda-MAD-02:~# openssl pkcs12 -in CertWebPhone.pfx -cicerts -nokeys -out Certificado.crt Enter Import Password: root@VC-WebP-AytoArganda-MAD-02:~# </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 3	Generar certificado
Comando	<code>openssl rsa -in webphone.ejemplo.es.key -out webphone.ejemplo.es.limpia.key</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# openssl rsa -in webphone.ayto-arganda.es.key -out webphone.ayto-arganda.es.limpia.key Enter pass phrase for webphone.ayto-arganda.es.key: writing RSA key</pre>
Paso 4	Copiar los certificados a /etc/apache2/ssl/
Comando	<code>cp Certificado.crt /etc/apache2/ssl/</code> <code>cp Clave.key /etc/apache2/ssl/</code>
Paso 5	Entrar en el directorio /etc/apache2/ssl
Comando	<code>cd /etc/apache2/ssl</code>
Paso 6	Renombrar el fichero que ya existe(webphone.ejemplo.es.crt)
Comando	<code>mv webphone.ejemplo.es.crt webphone.ejemplo.es.crt.orig</code>
Resultado esperado	<pre>mv webphone.ayto-arganda.es.crt webphone.ayto-arganda.es.crt.orig</pre>
Paso 7	Renombrar el fichero que ya existe (webphone.ejemplo.es.key)
Comando	<code>mv webphone.ejemplo.es.key webphone.ejemplo.es.key.orig</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# mv webphone.ayto-arganda.es.key webphone.ayto-arganda.es.key.orig root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet#</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 8	Renombrar Clave.key (Generado en el paso 1) a webphone.ejemplo.es.key
Comando	<code>mv Clave.key webphone.ejemplo.es.key</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# mv Clave.key webphone.ayto-arganda.es.key root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet#</pre>

Paso 9	Renombrar Certificado.crt (Generado en el paso 2) a webphone.ejemplo.es.crt
Comando	<code>mv Certificado.crt webphone.ejemplo.es.crt</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# mv Certificado.crt webphone.ayto-arganda.es.crt root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet#</pre>

Paso 10	Renombrar Certificado.crt a webphone.ejemplo.es.crt
Comando	<code>mv webphone.ejemplo.es.key webphone.ejemplo.es.key.cifrada</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# mv webphone.ayto-arganda.es.key webphone.ayto-arganda.es.key.cifrada root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet#</pre>

Paso 11	Renombrar Certificado.crt a webphone.ejemplo.es.crt
Comando	<code>mv webphone.ejemplo.es.limpia.key webphone.ejemplo.es.key</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# mv webphone.ayto-arganda.es.limpia.key webphone.ayto-arganda.es.key root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet#</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 12	Cambiar permisos a webphone.ejemplo.es.key y webphone.ejemplo.es.crt
Comando	<pre>chmod 644 webphone.ejemplo.es.crt</pre> <pre>chmod 644 webphone.ejemplo.es.key</pre>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# chmod 644 webphone.ayto-arganda.es.crt root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet# chmod 644 webphone.ayto-arganda.es.key root@VC-WebP-AytoArganda-MAD-02:/etc/jitsi/meet#</pre>

Modificar el fichero /etc/hosts

Paso 13	Modificar fichero /etc/hosts
Comando	<pre>vi /etc/hosts</pre>

En el fichero de /etc/hosts , hay que añadir la ip del tomcat , con sus configuraciones y por último añadir la palabra (FONBO)

Se muestra un ejemplo de una configuración del fichero hosts:

Ejemplo:

```
10.133.133.10 BDTR BDHIST VC-corp-zonaejemplo FONBO
```

3.2.7 Configuración de firewall (iptables)

Dependiendo de la arquitectura del cliente puede tener una configuración u otra.

Un **firewall**, también llamado cortafuegos, es un sistema cuya función es prevenir y proteger a nuestra red privada, de intrusiones o ataques de otras redes, bloqueándole el acceso. Permite el tráfico entrante y saliente que hay entre redes u ordenadores de una misma red.

Las **reglas de entrada** filtran el tráfico que pasa de la red al equipo local en función de las condiciones de filtrado especificadas en la regla. Por el contrario, las **reglas de salida** filtran el tráfico que pasa del equipo local a la red en función de las condiciones de filtrado especificadas en la regla.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Iptables es un módulo del núcleo de Linux que se encarga de filtrar los paquetes de red, es decir, es la parte que se encarga de determinar qué paquetes de datos queremos que lleguen hasta el servidor y cuáles no. Iptables gestiona, mantiene e inspecciona las reglas de filtrado de paquetes IPv4 a través de tablas. Estas tablas clasifican y organizan las reglas de acuerdo al tipo de decisiones que se deben tomar sobre los paquetes.

El primer paso es obtener los ficheros (**.sh, firewall**) y copiarlos en la máquina. Después hay que crear la carpeta firewall en **/etc**

firewall.sh -> Contiene todas las reglas iptables, **NO TOCAR**

vars.sh -> Fichero donde se configura la red del cliente así como las interfaces públicas y privadas de la máquina.

Paso 1	Crear la carpeta /etc/firewall
Comando	<code>mkdir /etc/firewall</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:~# mkdir /etc/firewall root@VC-WebP-AytoArganda-MAD-02:~#</pre>
Paso 2	Entrar en el directorio /etc/firewall
Comando	<code>cd /etc/firewall/</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc# cd firewall/ root@VC-WebP-AytoArganda-MAD-02:/etc/firewall#</pre>
Paso 3	Mover los ficheros .sh a /etc/firewall
Comando	<code>mv -vi /carpetaorigen/*.sh .</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/firewall# mv -vi /home/sat/*.sh . renamed '/home/sat/firewall.sh' -> './firewall.sh' renamed '/home/sat/vars.sh' -> './vars.sh'</pre>

*La carpeta origen es dónde se han copiado los ficheros a la máquina.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 4	Poner permisos de ejecución a los ficheros .sh
Comando	<code>chmod +x *.sh</code>
Resultado esperado	<pre> root@VC-WebP-AytoArganda-MAD-02:/etc/firewall# chmod +x * root@VC-WebP-AytoArganda-MAD-02:/etc/firewall# root@VC-WebP-AytoArganda-MAD-02:/etc/firewall# root@VC-WebP-AytoArganda-MAD-02:/etc/firewall# root@VC-WebP-AytoArganda-MAD-02:/etc/firewall# ls -l total 8 -rwxr-xr-x 1 root root 1322 feb 2 12:04 firewall.sh -rwxr-xr-x 1 root root 75 feb 2 12:01 varsh.sh </pre>

Paso 5	Poner permisos de ejecución a los ficheros .sh
Comando	<code>chown root:root *.sh</code>
Resultado esperado	<pre> root@VC-Web-20na-franca-VIGO-02:/etc/firewall# chown root:root *.sh root@VC-Web-20na-franca-VIGO-02:/etc/firewall# ls -l total 8 -rwxr-xr-x 1 root root 2034 ene 18 11:33 firewall.sh -rwxr-xr-x 1 root root 171 ene 18 11:33 varsh.sh </pre>

Paso 6	Editar el fichero /etc/firewall/vars.sh
Comando	<code>vi vars.sh</code>

Copiar el contenido que está debajo y pegarlo al final del fichero vars.sh

```
#!/bin/bash
```

```

export ETH_LAN=ens3
export IP_LAN=172.25.128.253
export RED_LAN=172.25.0.0/23
export ETH_DMZ=ens7
export IP_DMZ=89.140.51.149
export RED_DMZ=89.140.51.144/29

```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 1	Editar el fichero firewall.sh
Comando	<code>vi firewall.sh</code>

Copiar el contenido que está debajo y pegarlo al final del fichero firewall.sh

```
#!/bin/bash

DIR=$(dirname "$0")

. ${DIR}/vars.sh

# Se inician las cadenas
iptables -t filter -F
iptables -t filter -X
iptables -t nat -F
iptables -t nat -X

### INPUT
iptables -P INPUT ACCEPT

# DMZ filtrada
# privadas no
iptables -A INPUT -i $ETH_DMZ -s 192.168.0.0/16 -j DROP
iptables -A INPUT -i $ETH_DMZ -s 172.16.0.0/12 -j DROP
iptables -A INPUT -i $ETH_DMZ -s 10.0.0.0/8 -j DROP
# otras ip publicas no valen para RTP
# iptables -A INPUT -i $ETH_DMZ -s 213.97.161.214 -p udp -j DROP
# web
iptables -A INPUT -i $ETH_DMZ -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -i $ETH_DMZ -p tcp --dport 443 -j ACCEPT
# rtp / udp (jitsi videobridge)
iptables -A INPUT -i $ETH_DMZ -p udp --dport 10000 -j ACCEPT
# rtp / udp (/etc/janus/janus.cfg)
iptables -A INPUT -i $ETH_DMZ -p udp --dport 20000:20999 -j ACCEPT
# turn
# iptables -A INPUT -i $ETH_DMZ -p tcp --dport 3478 -j ACCEPT
# iptables -A INPUT -i $ETH_DMZ -p tcp --dport 5349 -j ACCEPT
# iptables -A INPUT -i $ETH_DMZ -p udp --dport 60000:60099 -j
ACCEPT
# xmpp
# iptables -A INPUT -i $ETH_DMZ -p tcp --dport 5222 -j ACCEPT
# acme
iptables -A INPUT -i $ETH_DMZ -p tcp --sport 443 -m state --state
RELATED,ESTABLISHED -j ACCEPT
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```
# stun STUN_MAPPING_HARVESTER_ADDRESSES en
/etc/jitsi/videobridge/sip-communicator.properties
iptables -A INPUT -i $ETH_DMZ -p udp --sport 443 -m state --state
RELATED,ESTABLISHED -j ACCEPT
# icmp pruebas
iptables -A INPUT -i $ETH_DMZ -p icmp -s 213.97.161.214 -j ACCEPT
iptables -A INPUT -i $ETH_DMZ -p icmp --icmp-type 0 -j ACCEPT
# iperf3
iptables -A INPUT -i $ETH_DMZ -p tcp --dport 4444 -j ACCEPT
# resto
iptables -A INPUT -i $ETH_DMZ -j DROP

#### OUTPUT
iptables -P OUTPUT ACCEPT
# otras ip publicas no valen para RTP
# iptables -A OUTPUT -o $ETH_DMZ -d 213.97.161.214 -p udp -j DROP
# no se permiten salidas con ip cruzadas
iptables -A OUTPUT -o $ETH_LAN -s $RED_DMZ -j DROP
iptables -A OUTPUT -o $ETH_DMZ -s $RED_LAN -j DROP

#### FORWARD
iptables -P FORWARD ACCEPT
iptables -A FORWARD -i $ETH_DMZ -j DROP
iptables -A FORWARD -o $ETH_DMZ -j DROP
~
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 2	Mover el fichero firewall (del directorio dónde se han pegado) a /etc/init.d
Comando	<code>mv -vi /carpetaorigen/ /etc/init.d/</code>
Resultado esperado	<pre>root@VC-Web-ZonaFranca-VIGO-02:/etc/firewall# mv -vi /home/sat/firewall /etc/init.d/ renamed '/home/sat/firewall' -> '/etc/init.d/firewall'</pre>

Paso 3	Entrar en el directorio /etc/init.d
Comando	<code>cd /etc/init.d</code>
Resultado esperado	<pre>root@VC-WebP-AytoArganda-MAD-02:/etc/firewall# cd /etc/init.d/ root@VC-WebP-AytoArganda-MAD-02:/etc/init.d#</pre>

Paso 4	Poner permisos de ejecución a al fichero firewall
Comando	<code>chmod +x firewall</code> <code>chown root:root firewall</code>
Resultado esperado	<pre>root@VC-Web-ZonaFranca-VIGO-02:/etc/init.d# chown root:root firewall root@VC-Web-ZonaFranca-VIGO-02:/etc/init.d# chmod +x firewall</pre>

Paso 5	Entrar en el directorio /etc/rc2.d
Comando	<code>cd /etc/rc2.d/</code>
Resultado esperado	<pre>root@VC-Web-ZonaFranca-VIGO-02:/etc/init.d# cd /etc/rc2.d/</pre>

Paso 6	Crear enlace simbólico
Comando	<code>ln -s ../init.d/firewall s10firewall</code>
Resultado esperado	<pre>root@VC-Web-ZonaFranca-VIGO-02:/etc/rc2.d# ln -s ../init.d/firewall S10firewall</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3.2.8 Instalación y configuración de fail2ban

Fail2ban supervisa las entradas en los archivos de registro de los distintos servicios (por ejemplo, SSH o FTP). Demasiados intentos fallidos hacen que la dirección IP atacante se bloquee durante un cierto período de tiempo para que no se establezcan más conexiones.

Paso 1	Instalar fail2ban
Comando	<code>apt install fail2ban</code>
Resultado esperado	<pre> root@VC-WebP-AytoArganda-MAD-02:/etc/init.d# apt-get install fail2ban Leyendo lista de paquetes... Hecho Creando árbol de dependencias Leyendo la información de estado... Hecho Se instalarán los siguientes paquetes adicionales: python3-pyinotify whois Paquetes sugeridos: mailx monit python-pyinotify-doc Se instalarán los siguientes paquetes NUEVOS: fail2ban python3-pyinotify whois 0 actualizados, 3 nuevos se instalarán, 0 para eliminar y 0 no actualizados. Se necesita descargar 444 kB de archivos. Se utilizarán 2,400 kB de espacio de disco adicional después de esta operación. ¿Desea continuar? [S/n] s </pre>

Paso 2	Entrar en el directorio /etc/fail2ban
Comando	<code>cd /etc/fail2ban/</code>
Resultado esperado	<pre> root@VC-Web-IonaFrancs-VIGO-02:/etc/firewall# cd /etc/fail2ban/ root@VC-Web-IonaFrancs-VIGO-02:/etc/fail2ban# ls -l total 64 drwxr-xr-x 2 root root 4096 ene 18 11:45 action.d -rw-r--r-- 1 root root 2817 ene 11 2020 fail2ban.conf drwxr-xr-x 2 root root 4096 mar 2 2020 fail2ban.d drwxr-xr-x 3 root root 4096 ene 18 11:45 filter.d -rw-r--r-- 1 root root 25740 ene 11 2020 jail.conf drwxr-xr-x 2 root root 4096 ene 18 11:45 jail.d -rw-r--r-- 1 root root 645 ene 11 2020 paths-arch.conf -rw-r--r-- 1 root root 2827 ene 11 2020 paths-common.conf -rw-r--r-- 1 root root 573 ene 11 2020 paths-debian.conf -rw-r--r-- 1 root root 738 ene 11 2020 paths-opensuse.conf root@VC-Web-IonaFrancs-VIGO-02:/etc/fail2ban# mv jail.conf jail </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 3	Hacer una copia de seguridad del fichero jail.conf
Comando	<code>cp jail.conf jail.conf.orig</code>

Modificar el fichero jail.conf

Paso 5	Editar el fichero jail.conf
Comando	<code>vi /etc/fail2ban/jail.conf</code>

En el fichero jail.conf , hay que editar las siguientes líneas :

Se configura la línea de **ignoreip** si no se quiere banear a las ip que estén dentro de la red del cliente.(Comentar la línea con `#wkwignoreip` y añadirla abajo para modificarla)

`ignoreip = 127.0.0.1/8 ::1 redcliente`

Ejemplo:

```
# wkw ignoreip = 127.0.0.1/8 ::1
ignoreip = 127.0.0.1/8 ::1 172.20.0.0/16
```

Se configura la línea de **bantime** para poner el tiempo de baneo.
(Comentar la línea con `#wkwbantime` y añadirla abajo para modificarla)

```
#wkwbantime
bantime = 15 m
```

Se configura la línea de **Usedns** poniendo un `no` .
(Comentar la línea con `#wkwUsedns` y añadirla abajo para modificarla)

```
#wkw Usedns = warn
usedns = no
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 6	Entrar en el directorio jail.d
Comando	<code>cd jail.d/</code>
Resultado esperado	<pre>root@VC-WEB-ZonaFranca-VIG0-02:/etc/fail2ban# cd jail.d/ root@VC-WEB-ZonaFranca-VIG0-02:/etc/fail2ban/jail.d# ls -l total 4 -rw-r--r-- 1 root root 22 mar 2 2020 defaults-debian.conf</pre>

Paso 7	Hacer una copia de seguridad del fichero defaults-debian.conf
Comando	<code>cp -vi defaults-debian.conf defaults-debian.conf.orig</code>

Modificar el fichero defaults-debian.conf

Paso 8	Editar el fichero defaults-debian.conf
Comando	<code>vi defaults-debian.conf</code>

En el fichero defaults-debian.conf hay que poner la palabra "true".

```
[sshd]
enabled = true
```

Paso 9	Entrar en el directorio action
Comando	<code>cd /etc/fail2ban/action</code>
Resultado esperado	<pre>root@VC-WEB-ZonaFranca-VIG0-02:/etc/fail2ban/jail.d# cd .. root@VC-WEB-ZonaFranca-VIG0-02:/etc/fail2ban# cd action.d/ root@VC-WEB-ZonaFranca-VIG0-02:/etc/fail2ban/action.d#</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 10	Hacer una copia de seguridad del fichero iptables-common.conf
Comando	<code>cp iptables-common.conf iptables-common.conf.orig</code>

Modificar el fichero iptables-common.conf

Paso 11	Editar el fichero iptables-common.conf
Comando	<code>vi iptables-common.conf</code>

Se configura la línea de **blocktype** para poner la palabra "DROP"
(Comentar la línea con #wkwbantime y añadirla abajo para modificarla)

```
#wkw blocktype = REJECT --reject-with icmp6-port-unreachable
blocktype = DROP
```

Modificar el fichero /etc/fail2ban/jail.d/serCen.conf

Paso 12	Editar el fichero /etc/fail2ban/jail.d/serCen.conf
Comando	<code>vi /etc/fail2ban/jail.d/serCen.conf</code>

Copiar el contenido que está debajo y pegarlo al final del fichero serCen.conf

```
[serCen]
enabled = true
port = 80,443
protocol = tcp
filter = serCen
blocktype = DROP
action = %(action_)s[blocktype=%(blocktype)s]
logpath = /var/log/serCen/serCen.log
bantime = 900
findtime = 600
maxretry = 10
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Modificar el fichero /etc/fail2ban/jail.d/vivait.conf

Paso 13	Editar el fichero vivait.conf
Comando	<code>vi /etc/fail2ban/jail.d/vivait.conf</code>

Copiar el contenido que está debajo y pegarlo al final del fichero vivait.conf

```
[vivait-janus]
enabled = true
port = 80,443
filter = vivait-janus
blocktype = DROP
action = %(action_)s[blocktype=%(blocktype)s]
logpath = /var/log/janus/janus.log
bantime = 900
findtime = 600
maxretry = 10
```

```
[vivait-apache]
enabled = true
port = 80,443
filter = vivait-apache
blocktype = DROP
action = %(action_)s[blocktype=%(blocktype)s]
logpath = /var/log/apache2/access.log
# hay bastantes falsos positivos
bantime = 900
findtime = 600
maxretry = 100
~
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Modificar el fichero /etc/fail2ban/filter.d/vivait-janus.conf

Paso 14	Editar el fichero /etc/fail2ban/filter.d/vivait-janus.conf
Comando	<code>vi /etc/fail2ban/filter.d/vivait-janus.conf</code>

Copiar el contenido que está debajo y pegarlo al final del fichero vivait-janus.conf

```
[INCLUDES]
before = common.conf

[Definition]
_daemon = janus

failregex = TRUCO_KO \[<HOST>\]

ignoreregex =
~
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Modificar el fichero /etc/fail2ban/filer.d/vivait-apache.conf

Paso 15	Editar el fichero /etc/fail2ban/filter.d/vivait-apache.conf
Comando	<code>vi /etc/fail2ban/filter.d/vivait-apache.conf</code>

Copiar el contenido que está debajo y pegarlo al final del fichero vivait-apache.conf

[INCLUDES]

before = botsearch-common.conf

[Definition]

_daemon = apache2

failregex = ^<HOST> \- \S+ \[\] \"(GET|POST|HEAD|CONNECT) \/?\S* \S+\" .+\$

```
ignoreregex = ^<HOST> \- .*\/http-bind\/?.*\" 200
               ^<HOST> \- .*\/janus\/?.*\" (200|304)
               ^<HOST> \- .*\/webfon2\/?.*\" (200|304)
               ^<HOST> \- .*\/webrtc-test\/?.*\" (200|304)
               ^<HOST> \- .*\/Vivait-FonB0\/?.*\" (200|304)
               ^<HOST> \- .*\/css\/.*\" (200|304)
               ^<HOST> \- .*\/images\/.*\" (200|304)
               ^<HOST> \- .*\/lang\/.*\" (200|304)
               ^<HOST> \- .*\/libs\/.*\" (200|304)
               ^<HOST> \- .*\/fonts\/.*\" (200|304)
               ^<HOST> \- .*\/static\/.*\" (200|304)
```

~

~

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Iniciar fail2ban

Paso 16	Iniciar fail2ban
Comando	<code>/etc/init.d/fail2ban start</code>
Resultado esperado	<pre>root@VC-ZonaFranca-VIG0-02:/etc/fail2ban/filter.d# /etc/init.d/fail2ban start Starting fail2ban (via systemctl): fail2ban.service.</pre>
Paso 17	Comprobar que fail2ban funciona correctamente
Comando	<code>/etc/init.d/fail2ban status</code>
Resultado esperado	<pre>root@VC-ZonaFranca-VIG0-02:/etc/fail2ban/filter.d# /etc/init.d/fail2ban status fail2ban.service - Fail2Ban Service Loaded: loaded (/lib/systemd/system/fail2ban.service; enabled; vendor preset: enabled) Active: active (running) since Tue 2022-01-18 12:17:49 UTC; 4s ago Docs: man:fail2ban(1) Process: 3118581 ExecStartPre=/bin/mkdir -p /run/fail2ban (code=exited, status=0/SUCCESS) Main PID: 3118230 (f2b/server) Tasks: 0 (limit: 9447) Memory: 248.0K CGroup: /system.slice/fail2ban.service └─ 3118230 /usr/bin/python3 /usr/bin/fail2ban-server --async -b -s /var/run/fail2ban/fail2ban.sock -p /var/run/fail2ban/fail2ban.pid --loglevel INFO --logtarget /var/log/fail2ban.log ene 18 12:17:49 VC-ZonaFranca-VIG0-02 systemd[1]: Starting Fail2Ban Service... ene 18 12:17:49 VC-ZonaFranca-VIG0-02 systemd[1]: Started fail2ban Service. ene 18 12:17:49 VC-ZonaFranca-VIG0-02 fail2ban-server[3118582]: 2022-01-18 12:17:49,585 fail2ban [3118582]: ERROR Server already running ene 18 12:17:49 VC-ZonaFranca-VIG0-02 fail2ban-server[3118582]: 2022-01-18 12:17:49,586 fail2ban [3118582]: ERROR Async configuration of server failed</pre>
Paso 18	Reiniciar fail2ban-client
Comando	<code>fail2ban-client restart</code>
Resultado esperado	<pre>root@VC-ZonaFranca-VIG0-02:/etc/fail2ban/filter.d# fail2ban-client restart Shutdown successful Server ready</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022

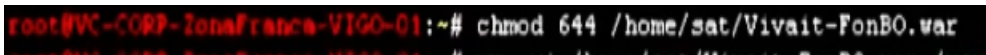


3.2.9 Subir FonBo.war a Tomcat

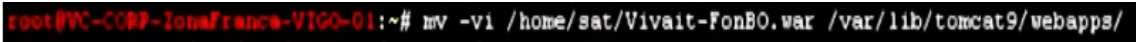
Importar la aplicación web empaquetada (Vivait-Fonbo.war) y copiarla en la máquina

Subir el archivo Vivait-FonBo.war a la máquina de Tomcat. Poner el archivo en una carpeta como por ejemplo /home/sat (Después se moverá ese archivo a otra carpeta)

Paso 1	Cambiar los permisos del archivo Vivait-FonBo.war
Comando	<code>chown root:root /home/sat/Vivait-FonBo.war</code>
Resultado esperado	

Paso 2	Cambiar los permisos del archivo FonBo.war
Comando	<code>chmod 644 /home/sat/Vivait-FonBo.war</code>
Resultado esperado	

Paso 3	Copiar el archivo Vivait-FonBo.war y renombrarlo a Vivait-Fonbo.war.orig (Por Seguridad)
Comando	<code>cp Vivait-FonBo.war Vivait-Fonbo.war.orig</code>

Paso 4	Mover el archivo FonBo.war a var/lib/tomcat9/webapps/
Comando	<code>mv -vi /home/sat/Vivait-FonBo.war /var/lib/tomcat9/webapps/</code>
Resultado esperado	

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3.2.10 Proteger Apache con Let's Encrypt (Instalando el paquete Cerbot)

*******(Solo para cuando el cliente no proporciona certificado)*******

Instalar certbot para apache

Paso 1	Instalar certbot para apache
Comando	<code>sudo apt install certbot python3-certbot-apache</code>

Parar el servicio de apache.

Paso 2	Parar el servicio de apache
Comando	<code>sudo /etc/init.d/apache2 stop</code>
Resultado esperado	<pre> root@vivait-meet-install:/# /etc/init.d/apache2 stop Stopping apache2 (via systemctl): apache2.service. root@vivait-meet-install:/# </pre>

Comprobar el estado del servicio apache.

Paso 3	Comprobar el estado del servicio apache
Comando	<code>sudo service apache2 status</code>
Resultado esperado	<pre> root@vivait-meet-install:/# service apache2 status ● apache2.service - The Apache HTTP Server Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled) Active: inactive (dead) since Wed 2022-09-07 12:47:52 CEST; 10s ago Docs: https://httpd.apache.org/docs/2.4/ Process: 302297 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS) Process: 302548 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/SUCCESS) Main PID: 302316 (code=exited, status=0/SUCCESS) sep 07 12:41:21 vivait-meet-install systemd[1]: Starting The Apache HTTP Server... sep 07 12:41:21 vivait-meet-install apachectl[302311]: AH00558: apache2: Could not reliably determine sep 07 12:41:21 vivait-meet-install systemd[1]: Started The Apache HTTP Server. sep 07 12:47:51 vivait-meet-install systemd[1]: Stopping The Apache HTTP Server... sep 07 12:47:52 vivait-meet-install apachectl[302550]: AH00558: apache2: Could not reliably determine sep 07 12:47:52 vivait-meet-install systemd[1]: apache2.service: Succeeded. sep 07 12:47:52 vivait-meet-install systemd[1]: Stopped The Apache HTTP Server. root@vivait-meet-install:/# </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Comprobar la configuración del host virtual de Apache

Para poder obtener y configurar un certificado SSL de forma automática para su servidor web, Certbot debe poder encontrar el host virtual adecuado entre los archivos de configuración de Apache. Los nombres de dominio de su servidor se obtendrán de las directivas `ServerName` y `ServerAlias` definidas en su bloque de configuración de `VirtualHost`.

Si siguió el [paso de configuración del host virtual en el tutorial de instalación de Apache](#), debe tener un bloque `VirtualHost` establecido para su dominio en `/etc/apache2/sites-available/.conf` con las directivas `ServerName` y `ServerAlias` ya configuradas de forma adecuada.

Para comprobarlo, abra el archivo de host virtual de su dominio con `nano` o su editor de texto preferido:

Paso 4	Abrir el archivo de host virtual de su dominio
Comando	<code>sudo vi /etc/apache2/sites-available/your_domain.conf</code>

Busque las líneas de `ServerName` y `ServerAlias` existentes. Deberían tener el siguiente aspecto:

```
/etc/apache2/sites-available/your_domain.conf
...
ServerName your_domain
ServerAlias www.your_domain
...
```

Si ya tiene su `ServerName` y `ServerAlias` configuradas de esta manera, puede salir del editor de texto y continuar con el siguiente paso. Si utiliza `nano`, puede salir escribiendo `CTRL+X` y, luego, `Y` y `ENTER` para confirmar.

Si la configuración del host virtual no coincide con la del ejemplo, actualícela correspondientemente. Cuando termine, guarde el archivo y salga del editor. Luego, ejecute el siguiente comando para validar sus cambios:

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 5	Validar los cambios (si procede)
Comando	<code>sudo vi /etc/apache2/sites-available/your_domain.conf</code>
Resultado esperado	Debería obtener Syntax OK como respuesta.

Si encuentra un mensaje de error, vuelva a abrir el archivo de host virtual y verifique que no haya errores ortográficos y que no falten caracteres.

Paso 6	Cuando la sintaxis del archivo de configuración sea correcta, vuelva a cargar Apache para que los cambios surtan efecto
Comando	<code>sudo systemctl start apache2</code>
Resultado esperado	Con estos cambios, certbot podrá encontrar el bloque VirtualHost correcto y actualizarlo.

#si el servicio no está activo no se puede recargar, sería necesario iniciar apache con (systemctl start apache2)

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Obtener un certificado SSL

Certbot ofrece varias alternativas para obtener certificados SSL a través de complementos. El complemento de Apache se encargará de reconfigurar Apache y volver a cargar la configuración siempre que sea necesario. Para utilizar este complemento, escriba lo siguiente:

Paso 7	Para utilizar este complemento, escriba lo siguiente
Comando	<code>sudo certbot --apache</code>

Esta secuencia de comandos le solicitará que responda a una serie de preguntas para configurar su certificado SSL.

Acción1: Primero, le solicitará una dirección de correo electrónico válida. Esta dirección se utilizará para las notificaciones de renovación y los avisos de seguridad:

Output

```
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Plugins selected: Authenticator apache, Installer apache
Enter email address (used for urgent renewal and security notices) (Enter 'c' to
cancel): you@your_domain
```

Acción2 : Después de proporcionar una dirección de correo electrónico válida, presione ENTER para continuar con el siguiente paso. Luego, se le solicitará que confirme si acepta las condiciones de servicio de Let's Encrypt. Puede confirmar pulsando A y, luego, ENTER:

```
- - - - -
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf. You must
agree in order to register with the ACME server at
https://acme-v02.api.letsencrypt.org/directory
- - - - -
(A)gree/(C)ancel: A
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Acción 3: A continuación, se le solicitará que confirme si desea compartir su dirección de correo electrónico con Electronic Frontier Foundation para recibir noticias y otra información. Si no desea suscribirse a su contenido, escriba N. De lo contrario, escriba Y. Luego, presione ENTER para continuar con el siguiente paso.

```

- - - - -
Would you be willing to share your email address with the Electronic Frontier
Foundation, a founding partner of the Let's Encrypt project and the non-profit
organization that develops Certbot? We'd like to send you email about our work
encrypting the web, EFF news, campaigns, and ways to support digital freedom.
- - - - -
(Y)es/(N)o: N

```

Acción 4: En el siguiente paso, se le solicitará que informe a Certbot los dominios para los que desea activar HTTPS. Los nombres de dominio enumerados se obtienen de forma automática de la configuración del host virtual de Apache, por lo tanto, es importante que se asegure de que los ajustes de ServerName y ServerAlias estén configurados correctamente en su host virtual.

Si desea habilitar HTTPS para todos los nombres de dominio enumerados (recomendado), puede dejar la solicitud en blanco y presionar ENTER para continuar. De lo contrario, seleccione los dominios para los que desea habilitar HTTPS enumerando cada número correspondiente, separado por comas o espacios, y, luego, presionando ENTER.

```

Which names would you like to activate HTTPS for?
- - - - -
1: your_domain
2: www.your_domain
- - - - -
Select the appropriate numbers separated by commas and/or spaces, or leave input
blank to select all options shown (Enter 'c' to cancel):

```

Verá un resultado como este:

```

Obtaining a new certificate
Performing the following challenges:
http-01 challenge for your_domain
http-01 challenge for www.your_domain
Enabled Apache rewrite module
Waiting for verification...
Cleaning up challenges
Created an SSL vhost at /etc/apache2/sites-available/your_domain-le-ssl.conf
Enabled Apache socache_shmcb module
Enabled Apache ssl module
Deploying Certificate to VirtualHost /etc/apache2/sites-available/your_domain-le-ssl.conf
Enabling available site: /etc/apache2/sites-available/your_domain-le-ssl.conf
Deploying Certificate to VirtualHost /etc/apache2/sites-available/your_domain-le-ssl.conf

```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Acción 5 : A continuación, se le solicitará que seleccione si desea que el tráfico de HTTP se redirija a HTTPS o no. En la práctica, esto significa que cuando alguien visite su sitio web a través de canales no cifrados (HTTP), se lo redirigirá automáticamente a la dirección HTTPS de su sitio web. Seleccione 2 para habilitar el redireccionamiento o 1 si desea mantener HTTP y HTTPS como métodos de acceso al sitio web separados.

```
Please choose whether or not to redirect HTTP traffic to HTTPS, removing HTTP access.
- - - - -
1: No redirect - Make no further changes to the webserver configuration.
2: Redirect - Make all requests redirect to secure HTTPS access. Choose this for
new sites, or if you're confident your site works on HTTPS. You can undo this
change by editing your web server's configuration.
- - - - -
Select the appropriate number [1-2] then [enter] (press 'c' to cancel): 2
```

Con este paso, se completa la configuración de Certbot y se le presentarán los comentarios finales sobre su certificado nuevo e información sobre dónde localizar los archivos generados y cómo probar su configuración utilizando una herramienta externa que analiza la autenticidad del certificado:

```

- - - - -
Congratulations! You have successfully enabled https://your_domain and
https://www.your_domain

You should test your configuration at:
https://www.ssllabs.com/ssltest/analyze.html?d=your_domain
https://www.ssllabs.com/ssltest/analyze.html?d=www.your_domain
- - - - -

IMPORTANT NOTES:
- Congratulations! Your certificate and chain have been saved at:
/etc/letsencrypt/live/your_domain/fullchain.pem
Your key file has been saved at:
/etc/letsencrypt/live/your_domain/privkey.pem
Your cert will expire on 2020-07-27. 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"
- Your account credentials have been saved in your Certbot
configuration directory at /etc/letsencrypt. You should make a
secure backup of this folder now. This configuration directory will
also contain certificates and private keys obtained by Certbot so
making regular backups of this folder is ideal.
- 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
```

Ahora, su certificado está instalado y cargado en la configuración de Apache. Intente volver a cargar su sitio web utilizando https:// y observe el indicador de seguridad de su navegador. Debe indicar que su sitio está protegido debidamente, en general, con un icono de candado en la barra de direcciones.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Verificar la renovación automática de Certbot

Los certificados de Let's Encrypt son válidos únicamente por noventa días. El propósito de esto es incentivar a los usuarios a automatizar sus procesos de renovación de certificados. El paquete certbot que instalamos se ocupa de esto por nosotros añadiendo un temporizador systemd que se ejecutará dos veces al día y renovará automáticamente cualquier certificado que vaya a vencer en los próximos 30 días.

Paso 8	Consultar el estado del temporizador de certbot
Comando	<code>sudo systemctl status certbot.timer</code>
Resultado esperado	<pre>certbot.timer - Run certbot twice daily Loaded: loaded (/lib/systemd/system/certbot.timer; enabled; vendor preset: enabled) Active: active (waiting) since Mon 2020-05-04 20:04:36 UTC; 2 weeks 1 days ago Trigger: Thu 2020-05-21 05:22:32 UTC; 9h left Triggers: certbot.service</pre>

Paso 9	Probar el proceso de renovación, haciendo un simulacro con certbot
Comando	<code>sudo certbot renew --dry-run</code>
Resultado esperado	Si no ve errores, estará listo

Cuando sea necesario, Certbot renovará sus certificados y volverá a cargar apache para registrar los cambios. Si el proceso de renovación automática falla, Let's Encrypt enviará un mensaje a la dirección de correo electrónico que especificó en el que se le advertirá cuándo se aproxime la fecha de vencimiento de sus certificados.

Paso 10	Enumerar el listado de los temporizadores
Comando	<code>systemctl list-timers</code>
Resultado esperado	Si no ve errores, estará listo

Los temporizadores se pueden usar como una alternativa a cron . Los temporizadores tienen soporte incorporado para ejecutar eventos basados en el calendario, eventos de tiempo monotónicos y se pueden ejecutar de forma asíncrona.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3.2.11 Crear script escobatoken.pl

Crear el archivo escobaToken.pl en /usr/local/sbin

Este script será el encargado de borrar los registros de token de la base de datos que son ya inservibles, haciendo así la tabla más manejable.

Contenido del script:

```
#!/usr/bin/perl

use DBI;
use strict;
use warnings;
use Getopt::Long;
use Log::Log4perl;
#use Switch;

use feature qw/switch/;

no warnings 'uninitialized';

# Inicializamos el log
#
my $log_conf = q(
    log4perl.rootLogger          = DEBUG, LOG1
    log4perl.appender.LOG1       = Log::Log4perl::Appender::File
    log4perl.appender.LOG1.filename = /var/log/escobaToken.log
    log4perl.appender.LOG1.mode    = append
    log4perl.appender.LOG1.layout  = Log::Log4perl::Layout::attributionLayout
    log4perl.appender.LOG1.layout.ConversionPattern = %d %p %m %n
);
Log::Log4perl::init(\$log_conf);

my $logger = Log::Log4perl->get_logger();

my $filas=1;
my $totFilas=0;
my $dbh;
my $sth;
my $query;
my $res;
# Lee los parámetros en una variable rh_params
my $rh_params = {};
#s Cadena obligatoria
#s Cadena opcional
#i Entero obligatorio
#i Entero opcional
GetOptions($rh_params,
    'hDB=s',
    'ayuda',
);

# Si se ha especificado el parámetro 'ayuda', imprimir la ayuda y abandonar
$rh_params->{ayuda} && imprime_ayuda( 0 );
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```

unless (defined $rh_params->{hDB})
{
    $logger->logdie("ERROR: Falta param --hDB");
    imprime_ayuda(1);
}

sub imprime_ayuda {
    my $exit_status = shift;

    print <<"END"

    Uso: escobamcan.pl [parametros]

    Lee un fichero y genera un informe
    Todos los parámetros son obligatorios si no se indica lo contrario:

        --hDB: Servidor BD Nimitz a la que conectarse
        --ayuda: imprime esta ayuda (opcional)

END
;
    exit $exit_status;
}

#Datos de la conexión
my $db="nimitz";
my $host=$rh_params->{hDB};
my $port="3306";
my $userid="root";
my $passwd="ivivamysql";
my $connectionInfo="DBI:mysql:database=$db;$host:$port";

sub handle_error {
    my $message = shift;
    $logger->error("Error al conectar a $host: $message");
    exit;
}

# Realizamos la conexión a la base de datos
$dbh = DBI->connect($connectionInfo,$userid,$passwd) or handle_error(DBI->errstr);
# Empieza el programa
$logger->info("Conectado a la base de datos $host:$db");

#Desactivamos la copia a la réplica
$query = "set SQL_LOG_BIN=0";
$sth = $dbh->prepare($query);
$res = $sth->execute();

$logger->info("Borrando tabla DAT_TOKENS");

$query = "DELETE FROM DAT_TOKENS WHERE (D_HORA_FIN < DATE_SUB(now(), INTERVAL
15 DAY)) OR ((D_HORA_INICIO < DATE_SUB(now(), INTERVAL 15 DAY)) AND
(E_TOKEN_ESTADO<>200)) LIMIT 100";

```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```
$logger->info("Query: $query");

while($filas>0)
{
    # Primero hay que "preparar" el query
    $sth = $dbh->prepare($query);

    # Ejecutamos el query
    $filas = $sth->execute();

    if (defined $filas)
    {
        $totFilas = $totFilas + $filas;
    }
    else
    {
        $logger->warn("Error al borrar de DAT_TOKEN");
    }
}

$logger->info("Ejecutada query: $query");
$filas=1;
$logger->info("Filas borradas: $totFilas");
$totFilas=0;

#Activamos la copia a la réplica
$query = "set SQL_LOG_BIN=1";
$sth = $dbh->prepare($query);
$res = $sth->execute();

$sth->finish();
$logger->info("Borrado finalizado");
$dbh->disconnect;
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



4. Comprobaciones

4.1 Logrotate.d y cron.d

Comprobar que todos los ficheros relacionados con este procedimiento que están contenidos en el `/etc/logrotate.d` y `/etc/cron.d` estén descomentadas sus líneas para que los logs roten y se ejecuten las tareas de forma correcta.

4.2 Grabaciones

Para que funcione las grabaciones hace falta ejecutar un ssh desde el usuario vivait de la máquina que corre el recordCentral con destino el usuarioSSH que está puesto en el nodo en el portal de administración.

4.3 Ramdisk

Solo válido nodos que tengan la propiedad de grabar, es decir, todas menos el ACD ya que no posee un dialplan para la grabación.

Paso 1	En el nodo en cuestión editar el fichero rmaDisk en el directorio <code>/etc/cron.d/</code>
Comando	<code># vi /etc/cron.d/ramDisk</code>

Descomentar la línea:

```
#@reboot    root    mount -t tmpfs -o
size=2000M,nr_inodes=10k,mode=0777,uid=asterisk,gid=asterisk tmpfs
/var/spool/asterisk/monitor
```

Y ajustar el tamaño (size) a la mitad del valor de la RAM física de la máquina, la cual se puede comprobar con el comando `free`:

Un ejemplo sería:

```
root@preproduccioncorp0:~# free -h
              total        used          free      shared  buff/cache   available
Mem:          1.9Gi         1.1Gi          98Mi        2.0Mi        738Mi        664Mi
Swap:         2.0Gi         62Mi          1.9Gi
root@preproduccioncorp0:~#
```

Lo que daría lugar a modificar la citada línea como sigue:

```
@reboot    root    mount -t tmpfs -o
size=1000M,nr_inodes=10k,mode=0777,uid=asterisk,gid=asterisk tmpfs
/var/spool/asterisk/monitor
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



4.4 Certbot

Se puede comprobar que certbot funciona correctamente, para ello:

Paso 1	Comprobar funcionamiento de cerbot																																																																								
Comando	systemctl list-timers																																																																								
Resultado esperado	<pre>root@preproduccioncorp0:~# systemctl list-timers</pre> <table><thead><tr><th>NEXT</th><th>LEFT</th><th>LAST</th><th>PASSED</th><th>UNIT</th><th>ACTIVATES</th></tr></thead><tbody><tr><td>Thu 2022-09-08 14:09:00 CEST</td><td>2min 28s left</td><td>Thu 2022-09-08 13:39:00 CEST</td><td>27min ago</td><td>phpsessionclean.timer</td><td>phpsessionclean.service</td></tr><tr><td>Thu 2022-09-08 18:45:47 CEST</td><td>4h 39min left</td><td>Thu 2022-09-08 11:55:46 CEST</td><td>2h 10min ago</td><td>ua-timer.timer</td><td>ua-timer.service</td></tr><tr><td>Fri 2022-09-09 00:00:00 CEST</td><td>9h left</td><td>Thu 2022-09-08 00:00:01 CEST</td><td>14h ago</td><td>logrotate.timer</td><td>logrotate.service</td></tr><tr><td>Fri 2022-09-09 00:00:00 CEST</td><td>9h left</td><td>Thu 2022-09-08 00:00:01 CEST</td><td>14h ago</td><td>man-db.timer</td><td>man-db.service</td></tr><tr><td>Fri 2022-09-09 01:44:16 CEST</td><td>11h left</td><td>Thu 2022-09-08 13:44:47 CEST</td><td>21min ago</td><td>fwupd-refresh.timer</td><td>fwupd-refresh.service</td></tr><tr><td>Fri 2022-09-09 03:18:50 CEST</td><td>13h left</td><td>Thu 2022-09-08 03:18:50 CEST</td><td>10h ago</td><td>systemd-tmpfiles-clean.timer</td><td>systemd-tmpfiles-clean.service</td></tr><tr><td>Fri 2022-09-09 05:34:06 CEST</td><td>15h left</td><td>Thu 2022-09-08 07:42:10 CEST</td><td>6h ago</td><td>apt-daily.timer</td><td>apt-daily.service</td></tr><tr><td>Fri 2022-09-09 06:24:18 CEST</td><td>16h left</td><td>Thu 2022-09-08 13:34:47 CEST</td><td>31min ago</td><td>motd-news.timer</td><td>motd-news.service</td></tr><tr><td>Fri 2022-09-09 06:41:41 CEST</td><td>16h left</td><td>Thu 2022-09-08 06:30:47 CEST</td><td>7h ago</td><td>apt-daily-upgrade.timer</td><td>apt-daily-upgrade.service</td></tr><tr><td>Sun 2022-09-11 03:10:27 CEST</td><td>2 days left</td><td>Sun 2022-09-04 03:10:54 CEST</td><td>4 days ago</td><td>e2scrub_all.timer</td><td>e2scrub_all.service</td></tr><tr><td>Mon 2022-09-12 00:00:00 CEST</td><td>3 days left</td><td>Mon 2022-09-05 00:00:02 CEST</td><td>3 days ago</td><td>fstrim.timer</td><td>fstrim.service</td></tr></tbody></table> <pre>Pass --all to see loaded but inactive timers, too. root@preproduccioncorp0:~#</pre>	NEXT	LEFT	LAST	PASSED	UNIT	ACTIVATES	Thu 2022-09-08 14:09:00 CEST	2min 28s left	Thu 2022-09-08 13:39:00 CEST	27min ago	phpsessionclean.timer	phpsessionclean.service	Thu 2022-09-08 18:45:47 CEST	4h 39min left	Thu 2022-09-08 11:55:46 CEST	2h 10min ago	ua-timer.timer	ua-timer.service	Fri 2022-09-09 00:00:00 CEST	9h left	Thu 2022-09-08 00:00:01 CEST	14h ago	logrotate.timer	logrotate.service	Fri 2022-09-09 00:00:00 CEST	9h left	Thu 2022-09-08 00:00:01 CEST	14h ago	man-db.timer	man-db.service	Fri 2022-09-09 01:44:16 CEST	11h left	Thu 2022-09-08 13:44:47 CEST	21min ago	fwupd-refresh.timer	fwupd-refresh.service	Fri 2022-09-09 03:18:50 CEST	13h left	Thu 2022-09-08 03:18:50 CEST	10h ago	systemd-tmpfiles-clean.timer	systemd-tmpfiles-clean.service	Fri 2022-09-09 05:34:06 CEST	15h left	Thu 2022-09-08 07:42:10 CEST	6h ago	apt-daily.timer	apt-daily.service	Fri 2022-09-09 06:24:18 CEST	16h left	Thu 2022-09-08 13:34:47 CEST	31min ago	motd-news.timer	motd-news.service	Fri 2022-09-09 06:41:41 CEST	16h left	Thu 2022-09-08 06:30:47 CEST	7h ago	apt-daily-upgrade.timer	apt-daily-upgrade.service	Sun 2022-09-11 03:10:27 CEST	2 days left	Sun 2022-09-04 03:10:54 CEST	4 days ago	e2scrub_all.timer	e2scrub_all.service	Mon 2022-09-12 00:00:00 CEST	3 days left	Mon 2022-09-05 00:00:02 CEST	3 days ago	fstrim.timer	fstrim.service
NEXT	LEFT	LAST	PASSED	UNIT	ACTIVATES																																																																				
Thu 2022-09-08 14:09:00 CEST	2min 28s left	Thu 2022-09-08 13:39:00 CEST	27min ago	phpsessionclean.timer	phpsessionclean.service																																																																				
Thu 2022-09-08 18:45:47 CEST	4h 39min left	Thu 2022-09-08 11:55:46 CEST	2h 10min ago	ua-timer.timer	ua-timer.service																																																																				
Fri 2022-09-09 00:00:00 CEST	9h left	Thu 2022-09-08 00:00:01 CEST	14h ago	logrotate.timer	logrotate.service																																																																				
Fri 2022-09-09 00:00:00 CEST	9h left	Thu 2022-09-08 00:00:01 CEST	14h ago	man-db.timer	man-db.service																																																																				
Fri 2022-09-09 01:44:16 CEST	11h left	Thu 2022-09-08 13:44:47 CEST	21min ago	fwupd-refresh.timer	fwupd-refresh.service																																																																				
Fri 2022-09-09 03:18:50 CEST	13h left	Thu 2022-09-08 03:18:50 CEST	10h ago	systemd-tmpfiles-clean.timer	systemd-tmpfiles-clean.service																																																																				
Fri 2022-09-09 05:34:06 CEST	15h left	Thu 2022-09-08 07:42:10 CEST	6h ago	apt-daily.timer	apt-daily.service																																																																				
Fri 2022-09-09 06:24:18 CEST	16h left	Thu 2022-09-08 13:34:47 CEST	31min ago	motd-news.timer	motd-news.service																																																																				
Fri 2022-09-09 06:41:41 CEST	16h left	Thu 2022-09-08 06:30:47 CEST	7h ago	apt-daily-upgrade.timer	apt-daily-upgrade.service																																																																				
Sun 2022-09-11 03:10:27 CEST	2 days left	Sun 2022-09-04 03:10:54 CEST	4 days ago	e2scrub_all.timer	e2scrub_all.service																																																																				
Mon 2022-09-12 00:00:00 CEST	3 days left	Mon 2022-09-05 00:00:02 CEST	3 days ago	fstrim.timer	fstrim.service																																																																				

Si no se muestran errores indica que funciona correctamente.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5. Diagnósticos

5.1 Herramientas de diagnóstico

5.1.1 Funcionamiento de envío de emails

Desde la máquina webfon se envían dos tipos de emails.

- Fail2ban utiliza "exim4" para la notificación de eventos.

Para verificar los correos enviados por exim4:

```
cat /var/log/exim4/mainlog
```

- serCen utiliza "curl" para notificar a los usuarios el número PIN necesario para la autenticación de doble factor.

Para verificar el funcionamiento de envío de correos de serCen, disponemos de un script en el que hay que sustituir FROM, TO, IP, PUERTO Y CLAVE:

Paso 1	Creamos el script enviar_correo
Comando	<pre># vi /tmp/enviar_correo.sh</pre>

Copiamos en el fichero el texto siguiente completando/sustituyendo los campos: FROM, TO, IP:PUERTO USU Y CLV:

```
#!/bin/bash
```

```
FROM=""
```

```
TO=""
```

```
ASUNTO="Pruebas de correo para '${TO}' desde '${FROM}'"
```

```
URL="smtp://IP:PUERTO"
```

```
USU=""
```

```
CLV=""
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```
TMP=$(mktemp /tmp/tmp.XXXXXX)
echo -ne "From: <${FROM}>\r\n" >${TMP}
echo -ne "To: <${TO}>\r\n" >>${TMP}
echo -ne "Subject: ${ASUNTO}\r\n" >>${TMP}
echo -ne "\r\n" >>${TMP}
echo -ne "Esto es un mensaje de pruebas:\r\n" >>${TMP}
echo -ne "Desde: ${FROM}\r\n" >>${TMP}
echo -ne "Para: ${TO}\r\n" >>${TMP}
echo -ne "Asunto: ${ASUNTO}\r\n" >>${TMP}
echo -ne "Url: ${URL}\r\n" >>${TMP}
echo -ne "Usuario: ${USU}\r\n" >>${TMP}
echo -ne "Archivo: ${TMP}\r\n" >>${TMP}
```

```
cat >"${TMP}" <<EOF
From: <${FROM}>
To: <${TO}>
Subject: ${ASUNTO}
```

```
Esto es un mensaje de pruebas:
Desde: ${FROM}
Para: ${TO}
Asunto: ${ASUNTO}
Url: ${URL}
Usuario: ${USU}
Archivo: ${TMP}
EOF
```

```
curl -v "${URL}" \
--verbose \
--insecure \
--ssl \
--user "${USU}:${CLV}" \
--mail-from "${FROM}" \
--mail-rcpt "${TO}" \
--upload-file "${TMP}"
```

```
echo "Archivo temporal: ${TMP}"
rm -v "${TMP}"
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 2	Ejecutamos el script enviar_correo
Comando	<code># /tmp/enviar_correo.sh</code>

RESULTADO CORRECTO DEL ENVÍO DE CORREO MEDIANTE SCRIPT:

```

*   Trying 10.60.6.86:25...
*   TCP_NODELAY set
  % Total    % Received % Xferd  Average Speed   Time    Time
Time  Current                                Dload  Upload   Total
Spent     Left  Speed
  0      0    0     0    0     0     0     0  --:--:--
--:--:-- --:--:--          0* Connected to 10.60.6.86 (10.60.6.86)
port 25 (#0)
< 220 AytoExchHC1.aytoarga.es Microsoft ESMTP MAIL Service
ready at Fri, 11 Feb 2022 19:44:13 +0100
> EHLO tmp.hFg8HP
< 250-AytoExchHC1.aytoarga.es Hello [10.10.0.31]
< 250-SIZE 20971520
< 250-PIPELINING
< 250-DSN
< 250-ENHANCEDSTATUSCODES
< 250-AUTH
< 250-8BITMIME
< 250-BINARYMIME
< 250-CHUNKING
< 250 XEXCH50
> MAIL FROM:<vivait@ayto-arganda.es> SIZE=393
< 250 2.1.0 Sender OK
> RCPT TO:<vivait@ayto-arganda.es>
< 250 2.1.5 Recipient OK
> DATA
< 354 Start mail input; end with <CRLF>.<CRLF>
} [393 bytes data]
* We are completely uploaded and fine
< 250 2.6.0 <81786787-1dca-41b3-a7bb-
00a27dd69812@AytoExchHC1.aytoarga.es> Queued mail for
delivery
100    393    0    0  100    393    0  1097 --:--:--
--:--:-- --:--:--    1094
* Connection #0 to host 10.60.6.86 left intact
Archivo temporal: /tmp/tmp.hFg8HP
removed '/tmp/tmp.hFg8HP'

```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.1.2 Verificar puertos abiertos de un equipo

Paso 1	Verificar puertos abiertos de un equipo
Comando	En este caso es el puerto 25 del servidor de correo del cliente: <pre>nc 10.60.6.86 25 -nv</pre>
Resultado esperado	<pre>Connection to 10.60.6.86 25 port [tcp/*] succeeded! 220 toExchHC1.aytoarga.es Microsoft ESMTPL MAIL Service ready at Thu, 10 Feb 2022 19:35:40 +0100</pre>

5.1.3 Monitorizar el tráfico de un determinado puerto

Paso 1	Monitorizar el tráfico de un determinado puerto
Comando	En este caso es el puerto 25: <pre>ngrep -d any port 25 ngrep -d ens160 port 25 ngrep -d ens 160 port 25 and host 10.60.6.86</pre>

5.1.4 Capturar tráfico con tcpdump

La captura se puede guardar en un fichero para su posterior análisis con Wireshark.

Paso 1	Capturar tráfico con tcpdump
Comando	<pre>tcpdump -D (ver interfaces disponibles) tcpdump -i ens160 (especificar interfaz) tcpdump -i ens160 -w /tmp/prueba1.pcap (guardar para analizar)</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.1.5 Comprobar que hay usuarios logados en Webfon

Desde el propio portal de Vivait Call podemos comprobar qué usuarios están registrados en ese momento en el sistema Webfon:

En General → Usuarios → Administrar usuarios

Y comprobar que la extensión en "Movilidad" es una extensión webfon, y no una extensión normal:

VIVAIT CALL → Dispositivos → Extensiones

Cuando los usuarios hacen login, en el campo Extensión movilidad aparece reflejada su extensión de movilidad que tienen asignada (ya sea fija o mediante Pool). Por tanto, cuando este campo no esté vacío, significa que el usuario tiene la sesión iniciada. La extensión de movilidad también se puede verificar desde el propio portal de Vivait Call Web del usuario en Ajustes / Extensiones:

5.1.6 Gestionar certificados de Let's Encrypt

- Listar los certificados gestionados por Let's Encrypt:

```
sudo certbot certificates
```

- Renovar un certificado:

```
sudo certbot renew --cert-name example.org --force-renewal
```

- Expedir un nuevo certificado:

```
sudo certbot certonly --webroot -w /var/www/example.org -d  
exa
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.1.7 Comprobar Conexión server LDAP

Script para comprobar conexión correcta LDAP:

Paso 1	Creamos el script conexion_ldap
Comando	<code># vi /tmp/conexion_ldap.sh</code>

Copiamos en el fichero el texto siguiente completando/sustituyendo los campos necesario:

```
#!/bin/bash
curl --insecure -v -B \
--max-redirs 1 \
--max-time 1 \
-u \
'CN=vivait,OU=VivaIT,OU=DptoPoliciaLocalCentralizado,DC=aytoarga,DC=es' \ "ldap://10.60.6.11:389/dc=aytoarga,dc=es?*?sub?(sAMAccountName=vivait)"
```

#es necesario crear un script universal indicando las variables de los campos correspondientes

Paso 2	Ejecutamos el script conexion_ldap
Comando	<code># /tmp/conexion_ldap.sh</code>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.2 Logs y comandos

5.2.1 serCen

En serCen podemos monitorizar el funcionamiento del proceso de login de los usuarios.

Paso	Ver estado de serCen
Comando	<code>systemctl status serCen.service</code>
Resultado esperado	<pre>serCen.service - LSB: Start/stop serCen Loaded: loaded (/etc/init.d/serCen; generated) Active: active (exited) since Thu 2022-02-10 15:16:46 CET; 2h 38min ago Docs: man:systemd-sysv-generator(8) Process: 942 ExecStart=/etc/init.d/serCen start (code=exited, status=0/SUCCESS) feb 10 15:16:45 VC-WebP-AytoArganda-MAD-02 systemd[1]: Starting LSB: Start/stop serCen... feb 10 15:16:45 VC-WebP-AytoArganda-MAD-02 serCen[942]: Starting serCen feb 10 15:16:46 VC-WebP-AytoArganda-MAD-02 systemd[1]: Started LSB: Start/stop serCen.</pre>

Paso	Listar información de serCen
Comando	<code>nc localhost 1125</code>
Resultado esperado	<pre>serCen sis ver='00.01.04.1' inic='20220210 173436' alarmas=0 ultAlar='00000000 000000' serCen gmp msj=254/256 buf=256/256 tarea=97/102 serCen tmp uptime=1009 (0d 0h 16m 49s) serCen wws mysql=1 conxNum=0 conxMaxPeriodo=0 serCen wwc numCacheLibre=3 numColaPend=0 serCen wwc numReq=0/0 numGet=0/0 numPost=0/0 numPut=0/0 numDelete=0/0 serCen smt numCacheLibre=5 numColaPend=0 enPeriodo=0/0 numMsj=2/0</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 3	Monitorizar el log de serCen
Comando	<pre>cat /var/log/serCen/serCen.log</pre> <pre>tail -f /var/log/serCen/serCen.log</pre>

Validación positiva:

```
20220913 104513.429      WSI(13) usuarioPorCuenta ok cuenta=sat
id=1 modoLogin=10 usuMultiIp=0 tipo2Factor=no pinIntentosMax=0
clvExpiradaSegs=0 pinExpira=0
```

```
20220913 104513.431      WSI(13) postAutenticar1 bd_Ok cuenta=sat
id=1 clvExpiradaSegs=0
```

```
20220913 104513.438      WSI(13) tokenBdInsertar token=YwklAS
apli=webfon ipOrig=172.25.2.119 esta=200 tokenExpira=3600
pinExpira=0
```

```
20220913 104513.439      WSI(13) usuBdClaveComplejidad
diasValidezNueva=90 longiMin=4 requiereNumeros=0
requiereMayusculas=0 requiereMinusculas=0 requiereCarsEspeciales=0
```

Paso 4	Comprobar la configuración de serCen
Comando	<pre>/etc/MDtel/serCen.conf</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.2.2 serCen para LDAP

Paso 1	Monitorizar el log de serCen
Comando	<pre>cat /var/log/serCen/serCen.log</pre> <pre>tail -f /var/log/serCen/serCen.log</pre>

20220210 174843.708 WSI usuarioPorCuenta ok cuenta=vivait id=1001 modoLogin=20 multiIp=0 tipo2Factor=email pinIntentosMax=5 pinExpira=600 20220210 174843.713 LDP(vivait) authenticated_1 tipo=ldap_std_dos_pasos host=10.60.6.11 url=ldap://10.60.6.11/OU=DptoPoliciaLocalCentralizado,DC=aytoarga,DC=es?dn?sub?(sAMAccountName=vivait) 20220210 174843.716 LDP(vivait) authenticated_2 tipo=ldap_std_dos_pasos host=10.60.6.11 usuDN=CN=vivait,OU=VivaIT,OU=DptoPoliciaLocalCentralizado,DC=aytoarga,DC=es url=ldap://10.60.6.11/OU=DptoPoliciaLocalCentralizado,DC=aytoarga,DC=es?dn?base 20220210 174843.716 WSI postAutenticar1 ldap_Ok cuenta=vivait id=1001 20220210 174843.719 WSI tokenBdInsertar token=GGz6Xr apli=webfon ipOrig=95.169.224.135 esta=100 tokenExpira=3600 pinExpira=600 20220210 174843.723 SMT funEnviarTexto leidos=362/65536 soloAsunto='vivait(webfon) Información de acceso' servidor='10.60.6.86' from='vivait@ayto-arganda.es' to='vivait@ayto-arganda.es' 20220210 174918.919 WSI tokenBdLeer token=GGz6Xr apli=webfon multiIp=0 ipOrig=95.169.224.135 tipo2Factor=email pinIntentos=0/5 errores=0/10 20220210 174918.921 WSI tokenBdValidar token=GGz6Xr apli=webfon esta=200 tokenExpira=3600	Usuario existe en VivaitCall. Usuario existe en LDAP de cliente. Clave LDAP de usuario correcta Envío de correo Lectura de PIN introducido PIN correcto
---	---

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.2.3 vivait-webfon

Paso	Listar información de vivait-webfon
Comando	<code>nc localhost 1122</code>
Resultado esperado	<pre> vivait-webfon base ver='0.0.4' inic='20220210 173154' uptime=2914s errores=0 ultError=00000000 000000 vivait-webfon mensajes max=8 ocup=0 t_med_us=0 vivait-webfon mysql cnxMax=2 thrNumConx=2 thrNumDisp=2 thrNumOcup=0 cmdMax=8 cmdOcup=0 vivait-webfon sesion max=25 ocup=1 acum=2 vivait-webfon sesion linCont=0 hiloMediaCont=0 confOcup=0 confMax=100 vivait-webfon media max=500 ocup=0 vivait-webfon sip sipDispMax=25 sipDispOcup=1 s_root_cont=1 s_home_cont=1 s_nua_cont=1 </pre>

5.2.3 Janus

Paso	Ver estado de janus
Comando	<code>systemctl status janus.service</code>
Resultado esperado	<pre> janus.service - LSB: Start the Janus WEBRTC gateway Loaded: loaded (/etc/init.d/janus; generated) Active: active (exited) since Thu 2022-02-10 15:16:45 CET; 3h 15min ago Docs: man:systemd-sysv-generator(8) Process: 928 ExecStart=/etc/init.d/janus start (code=exited, status=0/SUCCESS) feb 10 15:16:45 VC-WebP-AytoArganda-MAD-02 systemd[1]: Starting LSB: Start the Janus WEBRTC gateway... feb 10 15:16:45 VC-WebP-AytoArganda-MAD-02 janus[928]: Starting janus: janus. feb 10 15:16:45 VC-WebP-AytoArganda-MAD-02 systemd[1]: Started LSB: Start the Janus WEBRTC gateway. feb 10 15:16:46 VC-WebP-AytoArganda-MAD-02 runuser[1045]: pam_unix(runuser:session): session opened for user janus by (uid=0) feb 10 17:31:49 VC-WebP-AytoArganda-MAD-02 runuser[1045]: pam_unix(runuser:session): session closed for user janus </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 2	Ver log de janus
Comando	<code>cat /var/log/janus/janus.log</code>
Resultado esperado	<pre> Logger plugins folder: /usr/lib/vivait/loggers ----- Starting Meetecho Janus (WebRTC Server) v0.10.10 ----- Checking command line arguments... Debug/log level is 3 Debug/log timestamps are enabled Debug/log colors are disabled [20220210 173154] [WARN] Reclaim session timeouts have been disabled, will cleanup immediately [20220210 173154] [WARN] mDNS resolution disabled, .local candidates will be ignored [20220210 173154] [WARN] MYQ[webfon] Conectado: "BDTR:nimitz:nimitz" [20220210 173154] [WARN] MYS[00] (vivait-webfon) Conectado con bd=nimitz [20220210 173154] [WARN] MYQ[webfon] Conectado: "BDTR:nimitz:nimitz" [20220210 173154] [WARN] MYS[01] (vivait-webfon) Conectado con bd=nimitz </pre>

Nota: un log de este tipo se visualiza después de parar y arrancar el servicio janus.

5.2.4 Configurar sesiones, dispositivos y conexiones simultáneas

Paso 1	Consultar/configurar entorno de webfon
Comando	<code>vi /etc/janus/vivait.plugin.webfon.jcfg</code>

Se obtiene un fichero de este tipo donde se pueden consultar/configurar algunos parámetros:

```

debug: {
    lock_debug = false
    refcount_debug = false
}

sesion: {

```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```

    local_nodo_id = 4
    sesiones_max = 25
concurrentes
    conferencias_max = 100
    lineas_disp_defecto = 2
    conferencias_disp_defecto = 1
    conferencias_beep_arch = "/etc/janus/beep.sln"
    conferencias_beep_defecto_segs = 30
    colas_limpiar_siempre = true
    message_control_dest = "control"
}

mensaje: {
    mensajes_max = 8
}

mysql: {
    num_conx = 2
    num_cmd = 8
    host = "BDTR"
    usuario = "nimitz"
    clave = "phikau3iwCe400PP5b09ng=="
    base_datos = "nimitz"
    charset = "utf8mb4"
}

supervision: {
    hay_spv = true
    local_ip = "0.0.0.0"
    puerto_escucha = 1122
}

sip: {
    video = false
    datos = false
    max_dispositivos = 25
(webfon) conectados simult.
    local_ip = "10.10.0.31"
    registro_ttl_segs = 3600
    latido_segs = 0
    max_digi_llam_sal = 10
    user_agent = "vivait-webfon"
    destino_buzon_dejar_msj = "***%s"
}

media: {
    media_max = 500
simultáneas
    rtp_min_puerto = 11000
    rtp_max_puerto = 11999
    audio_rtp_dscp = 0
    video_rtp_dscp = 0

```

☞ Número de sesiones máximas

☞ Número máximo de dispositivos

☞ Número máximo de conexiones

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```
    datos_rtp_dscp = 0
    dtmf_info = false
    dtmf_rtp_pt = 101
    dtmf_info_durac_ms = 100
    audio_codec = "pcma"
    video_codec_0 = "h264"
    video_rtp_pt_codec_0 = 107
}
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.2.5 Estado de APACHE

Paso 1	Consultar el log de apache
Comando	<pre>tail -f /var/log/apache2/access.log</pre> <pre>cat /var/log/apache2/access.log</pre>

Paso 2	Consultar el log de errores de apache
Comando	<pre>tail -f /var/log/apache2/error.log</pre> <pre>cat /var/log/apache2/error.log</pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.2.5 Fail2ban

Paso 1	Consultar el log de fail2ban
Comando	<pre>tail -f /var/log/fail2ban.log</pre> <pre>cat /var/log/fail2ban.log</pre>

Ejemplo de visualización:

Filtro aplicado (serCen) para actuar a los 10 fallos de autenticación

```
tail -f /var/log/fail2ban.log
```

```

2022-02-10 19:02:23,978 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:02:23
2022-02-10 19:02:28,116 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:02:28
2022-02-10 19:02:28,117 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:02:28
2022-02-10 19:04:46,953 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:04:46
2022-02-10 19:04:46,954 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:46
2022-02-10 19:04:47,721 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:04:47
2022-02-10 19:04:47,721 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:47
2022-02-10 19:04:48,423 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:04:48
2022-02-10 19:04:48,424 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:48
2022-02-10 19:04:48,749 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:04:48
2022-02-10 19:04:48,749 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:48
2022-02-10 19:04:49,199 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:04:49
2022-02-10 19:04:49,200 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:49
2022-02-10 19:04:49,605 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:04:49
2022-02-10 19:04:49,605 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:49
2022-02-10 19:04:49,944 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:04:49
2022-02-10 19:04:49,945 fail2ban.filter          [955]: INFO    [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:49
2022-02-10 19:04:50,239 fail2ban.filter          [955]: INFO    [serCen] Found
95.169.224.135 - 2022-02-10 19:04:50

```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



```

2022-02-10 19:04:50,239 fail2ban.filter [955]: INFO [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:50
2022-02-10 19:04:50,525 fail2ban.filter [955]: INFO [serCen] Found
95.169.224.135 - 2022-02-10 19:04:50
2022-02-10 19:04:50,526 fail2ban.filter [955]: INFO [vivait-nginx]
Found 95.169.224.135 - 2022-02-10 19:04:50
2022-02-10 19:04:50,692 fail2ban.actions [955]: NOTICE [serCen] Ban
95.169.224.135

```

Para contabilizar los errores, fail2ban lee el número de TRUCO_KO generados desde una determinada IP.

Podemos verlo también con el siguiente comando:

Paso 2	Consulta selectiva del log de serCen
Comando	<code>cat /var/log/serCen/serCen.log grep TRUCO_KO</code>

```

!20220210 190228.116 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190446.953 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190447.720 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190448.298 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190448.748 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190449.199 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190449.604 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190449.944 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190450.238 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave
!20220210 190450.525 WSI TRUCO_KO [95.169.224.135]
postAutenticar1_ldap_mal_clave

```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 3	Ver estado de fail2ban
Comando	<code>fail2ban-client status</code>
Resultado esperado	<pre> root@vivait-meet-install: # fail2ban-client status Status - Number of jail: 4 `- Jail list: serCen, sshd, vivait-apache, vivait-janus </pre>
Paso 4	Ver estado de fail2ban para serCen
Comando	<code>fail2ban-client status serCen</code>
Resultado esperado	<pre> Status for the jail: serCen - Filter - Currently failed: 0 - Total failed: 24 `-- File list:      /var/log/serCen/serCen.log `- Actions - Currently banned: 1 - Total banned: 2 `-- Banned IP list: 95.169.224.135 </pre>
Paso 5	Ver estado de fail2ban para janus
Comando	<code>fail2ban-client status vivait-janus</code>
Resultado esperado	<pre> Status for the jail: vivait-janus - Filter - Currently failed: 0 - Total failed: 0 `-- File list:      /var/log/janus/janus.log `- Actions - Currently banned: 0 - Total banned: 0 `-- Banned IP list: </pre>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 6	Ver estado de fail2ban para apache
Comando	<code>fail2ban-client status vivait-apache</code>
Resultado esperado	<pre>Status for the jail: vivait-apache - Filter - Currently failed: 2 - Total failed: 88 `-- File list:        /var/log/apache2/access.log `- Actions - Currently banned: 0 - Total banned: 0 `-- Banned IP list:</pre>

Paso 7	Desbanear una IP
Comando	<code>fail2ban-client set serCen unbanip 95.169.224.135</code>

Paso 8	Desbanear todo
Comando	<code>fail2ban-client unban -all</code>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.2.6 IPTABLES

Paso 1	Consulta de la configuración de IPTABLES
Comando	<code>iptables -L -n</code>

```
Chain INPUT (policy ACCEPT)
target     prot opt source                               destination
f2b-serCen  tcp  --  0.0.0.0/0                             0.0.0.0/0
multiport dports 80,443
ACCEPT     all  --  10.0.0.0/8                             0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0                             0.0.0.0/0      tcp
dpt:80
ACCEPT     tcp  --  0.0.0.0/0                             0.0.0.0/0      tcp
dpt:443
ACCEPT     udp  --  0.0.0.0/0                             0.0.0.0/0      udp
dpts:20000:20999
ACCEPT     tcp  --  0.0.0.0/0                             0.0.0.0/0      tcp
spt:443 state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0                             0.0.0.0/0      tcp
spt:80 state RELATED,ESTABLISHED
ACCEPT     icmp --  0.0.0.0/0                             0.0.0.0/0
icmp type 0
ACCEPT     tcp  --  0.0.0.0/0                             0.0.0.0/0      tcp
dpt:4444
ACCEPT     tcp  --  213.97.161.214                       0.0.0.0/0      tcp
dpt:22
ACCEPT     tcp  --  62.82.27.192/29                      0.0.0.0/0      tcp
dpt:22
DROP       all  --  0.0.0.0/0                             0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source                               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                               destination

Chain f2b-serCen (1 references)
target     prot opt source                               destination
DROP       all  --  95.169.224.135                       0.0.0.0/0
RETURN     all  --  0.0.0.0/0                             0.0.0.0/0
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.3 Otros errores y soluciones

5.2.1 Error: Webfon muestra Caller ID de la llamada anterior

ERROR

Se ha detectado un error en el webfon, al recibir una llamada con `caller id num` vacío, en este caso se muestra el de la llamada anterior.

SOLUCIÓN

Para resolver este fallo basta con cambiar el plugin que se encuentra en el fichero:
`/usr/lib/vivait/plugins/vivait_webfon_plugin.so`

Para realizar el cambio se debe seguir el siguiente procedimiento:

Paso 1	Parar janus
Comando	<code>/etc/init.d/janus stop</code>
Paso 2	Renombrar el antiguo antes de poner el nuevo
Comando	<code>cp /usr/lib/vivait/plugins/vivait_webfon_plugin.so /usr/lib/vivait/plugins/vivait_webfon_plugin.so.orig</code>
Paso 3	Sustituir el antiguo
Comando	<code>cp /directorio_origen/vivait_webfon_plugin.so /usr/lib/vivait/plugins/</code>

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Paso 4	Comprobar que una vez puesto el nuevo plugin tienen los mismos permisos que el antiguo
Comando	<code>ll /usr/lib/vivait/plugins/</code>
Resultado esperado	<pre> root@vivait-meet-install:/usr/lib# ll /usr/lib/vivait/plugins/ total 3908 drwxr-xr-x 2 root root 4096 sep 13 12:58 ./ drwxr-xr-x 6 root root 4096 mar 31 2021 ../ -rwxr-xr-x 1 root root 386024 feb 3 2021 libjanus_echotest.so* -rwxr-xr-x 1 root root 1801456 jul 18 18:27 vivait_webfon_plugin.so* -rwxr-xr-x 1 root root 1801456 sep 13 12:58 vivait_webfon_plugin.so.orig* </pre>

Paso 5	Arrancar janus
Comando	<code>/etc/init.d/janus start</code>

5.2.2 Error: serCen no envía correo

Error:	Desde la máquina se envían correos desde el prompt y no hay problemas, fail2ban también envía correos, solo falla cuando se envía desde el serCen.
Log de serCen:	SMT funEnviarTexto code='64/Requested SSL level failed' servidor='10.60.6.86' usuario='vivait'
Solución:	El servidor de correo de cliente no acepta el cifrado ssl (que en la versión anterior estaba forzada) . En la nueva versión no es obligatorio.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.2.3 Error al introducir la clave de LDAP en login

Error:	El sistema no permite hacer login.
Log de serCen:	<pre>cat /var/log/serCen/serCen.log</pre> <pre>** TRUKO_KO [X.X.X.X] **</pre> <pre>20220210 174805.114 WSI usuarioPorCuenta ok cuenta=vivait id=1001 modoLogin=20 multiIp=0 tipo2Factor=email pinIntentosMax=5 pinExpira=600</pre> <pre>20220210 174805.117 LDP(vivait) authenticated_1 tipo=ldap_std_dos_pasos host=10.60.6.11 url=ldap://10.60.6.11/OU=DptoPoliciaLocalCentralizado,DC=aytoarga,DC=es ?dn?sub?(SAMAccountName=vivait)</pre> <pre>20220210 174805.178 LDP(vivait) NOT_authenticated_2 tipo=ldap_std_dos_pasos usuDN=10.60.6.11 host=CN=vivait,OU=VivaIT,OU=DptoPoliciaLocalCentralizado,DC=aytoarga,DC=es url=ldap://10.60.6.11/OU=DptoPoliciaLocalCentralizado,DC=aytoarga,DC=es ?dn?base</pre> <pre>!20220210 174805.178 WSI TRUKO_KO [95.169.224.135] postAutenticar1_ldap_mal_clave</pre> <pre>20220210 174805.178 WSI postAutenticar1 ldap_no_clave cuenta=vivait id=1001</pre>
Solución:	fail2ban baneará la IP origen después de contabilizar el número de fallos de acceso

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



5.2.3 Modificación de serCen en glxcli000013

Que es glxcli

Se ha modificado serCen en glxcli000013. (versión actual 00.03.01.)

Los archivos modificados son:

<i>Fichero</i>	<i>Observaciones</i>
<code>/usr/sbin/serCen</code>	Este fichero soporta correctamente "modoLogin_deshabilitado". El antiguo está renombrado a <code>serCen.old</code>
<code>/etc/init.d/serCen</code>	Se le ha añadido la línea "export LD_PRELOAD=libtcpnat30.so" en este caso comentada. A partir de ahora, este archivo con la línea sin comentar se emplea cuando serCen se conecte con mysql a través de un NAT. En este caso, se requiere además del archivo <code>/usr/lib/libtcpnat30.so</code>
<code>/usr/lib/libtcpnat30.so</code>	Debería estar en todas las instalaciones para su posible uso.

NOTA: en las instalaciones "todo en uno" y donde no exista NAT, se comentará la línea "export LD_PRELOAD=libtcpnat30.so" del fichero `/usr/sbin/serCen`.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022

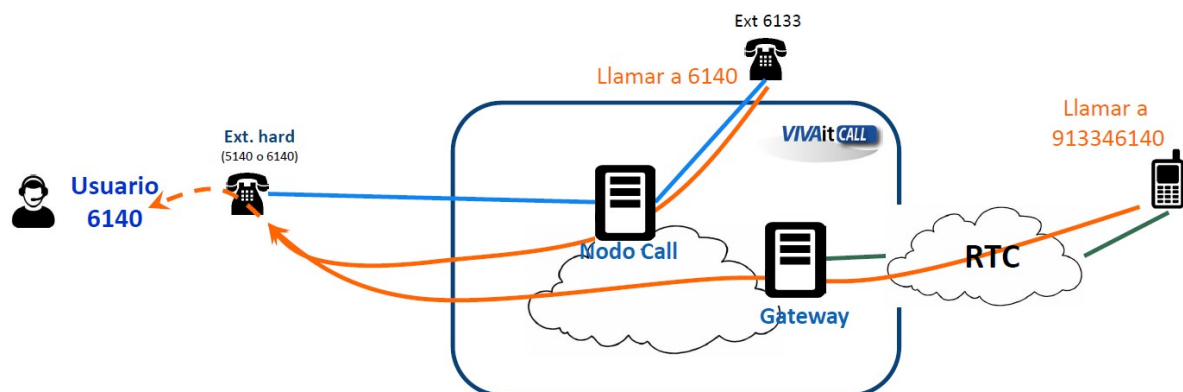


6. Arquitectura Gestión multidispositivo

La arquitectura general de **VIVAit** para la gestión multidispositivo se basa en la utilización de:

- Al menos un nodo tipo "corporativo" donde se registraran las Arquitectura extensiones físicas y se dan de alta las extensiones webfon, este será el encargado de cursar el tráfico telefónico, y
- De un nodo tipo "Webrtc" que será el encargado de albergar los servicios correspondientes que se invocaran desde un navegador con el que el usuario interactuara con la extensión web.

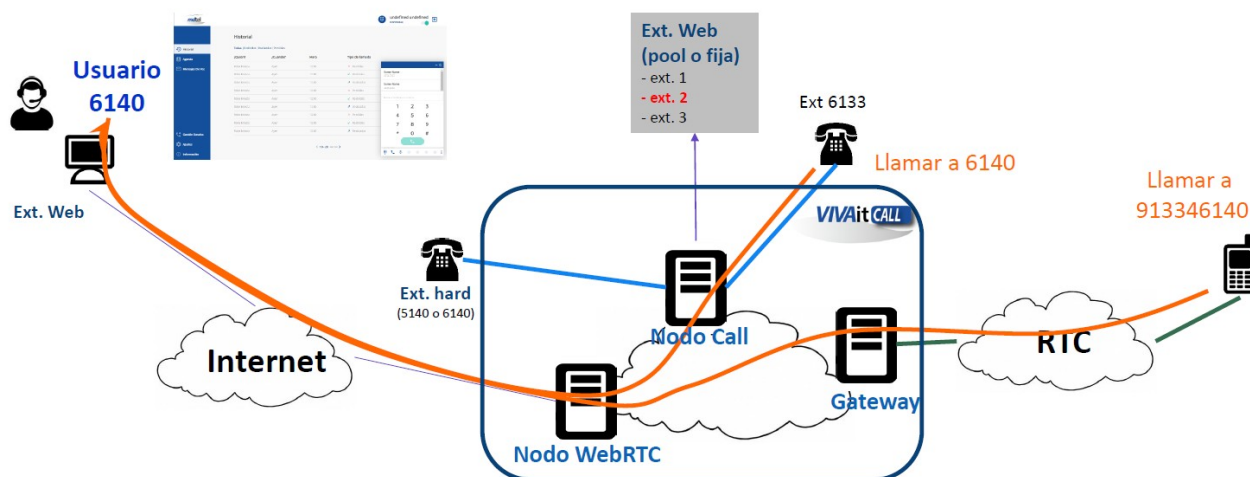
Esquema de funcionamiento con usuario en extensión física



Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022

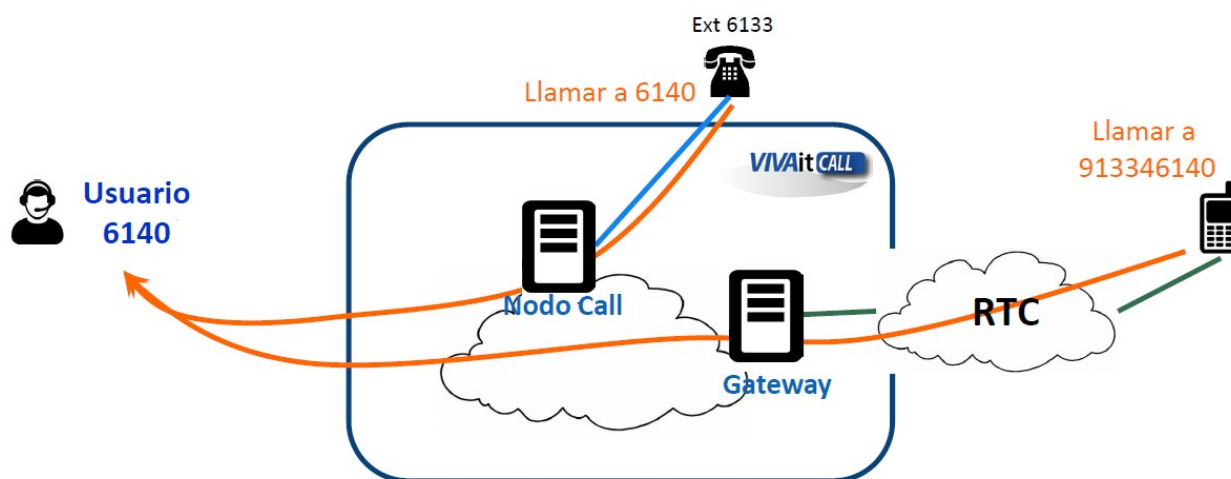


Esquema de funcionamiento con usuario en teletrabajo



Para el sistema es indiferente que la extensión Web este la oficina o fuera de ella (teletrabajo).

Esquema de funcionamiento con usuario sin extensión física



Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



6.1 Principales elementos funcionales

Aunque pueden existir otras configuraciones a analizar para propósitos específicos se van a considerar los siguientes elementos y su parametrización:

- **Usuario:** es el núcleo principal de la funcionalidad multidispositivo de **VIVAit**.
- **Extensión web:** es el "Número de login" en la pestaña "Centralita" del menú "General > Usuarios > Administrar usuarios" será el número que identifica al usuario que utiliza el servicio. No necesitan licencia
- **Extensión física:** también llamada "extensión hard" o "extensión SIP" será el terminal en el que el usuario puede interactuar; se trata de una extensión asignada a un terminal (hardphone o softphone) que se provisiona en **VIVAit** según corresponda. Necesita licencia.



En lo que sigue se va a considerar que el número de "extensión física" y el "número de extensión web" tienen el mismo valor y que en el plan de marcación (prerutas) esta numeración tiene como tipo de destino saliente 1 "Usuario o extensión".

- **Extensión webfon:** se trata de una extensión, con su plan de marcación exclusivo (no es relevante ya que nunca se va a marcar de forma interactiva esta numeración) a la que se dota de la plantilla de "Plantilla para Extensiones webfon" ya definida en el sistema. Necesita licencia.
- **Pool de extensiones:** se trata de un grupo de extensiones webfon que se compartirán entre los usuarios a los que se asigna el pool de extensiones webfon, en lugar de asignarles una extensión webfon fija.
- **Doble factor:** es un método de autenticación adicional en el que el sistema además de usuario y contraseña exige un pin que se envía a la cuenta de correo o al teléfono del usuario, este pin se debe introducir cada vez que el usuario se loga y puede condicionarse el entorno de red en el que es necesaria esta doble validación.
- **Nodo corporativo:** se trata de uno o varios nodos del tipo "Corporativo" en el que se dan de alta y se registran las extensiones físicas y se dan de alta las extensiones webfon.
- **Nodo Webrtc:** se trata del nodo donde se dan de alta los servicios necesarios para la funcionalidad multidispositivo, como son: serCen, janus, nginx, FonBo, seguridad y certificados, entre otros.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



- **Licencias:** contrala la cantidad de extensiones dadas de alta en el sistema **VIVAit**, debe tenerse en cuenta que las extensiones webfon consumen licencia, por lo que este concepto puede ser determinante a la hora de elegir el modo de trabajo:
 - Usuario con "Extensión Webfon".
 - Usuarios con "Pool de extensiones Webfon".

En lo referente a la llamada básica no tiene ninguna implicación un modo u otro, pero sí la tiene en cuanto a algunas facilidades del sistema.

- **VIVAit Call Web:** es el portal de acceso al servicio, la URL será pública o privada según el caso. Cuando se accede al servicio vía navegador será necesario introducir las credenciales para validarse en el servicio.

→ ↻ ⚠ No es seguro | labmeet.mdnova.local/webfon2/

Aplicaciones Roundcube webphone

VIVAit Call Web

Usuario

Contraseña

Entrar

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Doble Factor: es un código único generado para probar la identidad de un usuario cuando intenta acceder a su cuenta. Si se elige esta forma de funcionamiento el código se enviaría por email o teléfono asociado con el usuario. Al ingresar el código de autenticación de dos factores, el usuario obtiene acceso a su cuenta.

Verifique el PIN

Introduzca la clave generada por el dispositivo de autenticación

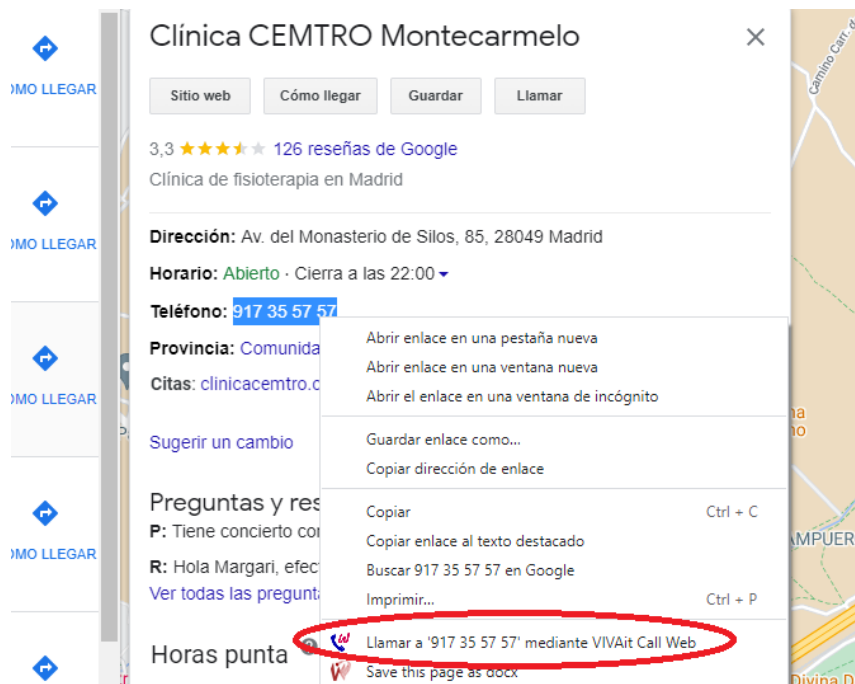
Introduzca la clave de un solo uso



Entrar

Volver

Click2Call: se trata de una extensión del navegador que permite marcar directamente números telefónicos que aparezcan en páginas web.



Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7. Pasos para la configuración de Web Call

En el momento que un usuario vaya a logarse en Webfon, debe:

- Tener asignada una extensión física (bien sea hardphone o softphone).
- Tener su usuario configurado en General → Usuarios → Administrar usuarios.
- Disponer de una extensión de movilidad o webfon, ya sea:
 - o Fija: La tendrá siempre asignada por lo que siempre podrá hacer login (a no ser que se configure un límite de sesiones simultáneas).
 - o Pool de extensiones: Se crea un número de extensiones de movilidad limitado y las asignará aleatoriamente a los usuarios (por lo que es posible que esté todo el grupo de extensiones ocupado y no se pueda disponer de extensión de movilidad y, por tanto, no hacer login). Se asemejan a canales.

Siempre que no se cumpla uno de estos dos requisitos, el usuario no tendrá permitido logarse en el sistema webfon.

Opcionalmente el usuario tendrá asignada una extensión física (bien sea hardphone o softphone) si se utiliza el concepto de multidispositivo.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7.1 Nodos

1. **Nodo/s corporativo/s:** la configuración de este nodo se realizará según el punto 3.3 del documento "Instalación de VM en VIVAit" del Dpto. de Desarrollo de MDTel.



Documentación
vivait-call VM.pdf

Nodos

Datos GENERALES

Id: 2 Nombre *: vivaitsuite-CORP Descripción: vivaitsuite-CORP

Tipo nodo *: Corporativo

Dirección de red *: 172.25.128.91

Sede *: Sede Madrid / c/Manuel Tovar, 38

Datos CTI

Cti_Host: Cti_Puerto: Usuario CTI: Clave CTI:

- **Nodo Webrtc:** antes de dar de alta el nodo en el portal de gestión es necesario su instalación/configuración según el "Manual de instalación y operación del nodo WebRTC", disponible en la wiki: [DOCUMENTO_WEBRTC](#).

Nodos

Datos GENERALES

Id: 4 Nombre *: vivait-webfon Descripción: NODO-WEBFON

Tipo nodo *: Webrtc

Dirección de red *: 172.25.128.253

Sede *: Sede Madrid / c/Manuel Tovar, 38

Datos CTI

Cti_Host: Cti_Puerto: Usuario CTI: Clave CTI:

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7.2 Plantilla

- **Plantilla webfone:** dentro del menú “General > Plantillas” debe existir “Plantilla para Extensiones webfon”.

Tiene que aparecer de la siguiente forma (texto del campo “Valor” copiado abajo):

Siendo el campo “Valor” el siguiente:

```
username=${SIP.EXTEN}
mailbox=${SIP.EXTEN}@default
cc_agent_policy=generic
cc_monitor_policy=generic
sendrpid=pai
trust_id_outbound=yes
trustrpid=yes
rpid_update=yes
insecure=no
setvar=DatosCTI=SI
qualify=yes
```



Esta plantilla debe existir en la carga inicial del sistema y no se debe modificar. Si es necesario realizar alguna modificación se duplicará esta plantilla, debiendo conservar el string “Extensiones webfon” en la nombre de la nueva. Esto permitirá realizar una búsqueda adecuada entre las extensiones de Vivait-

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7.3 Licencias

Además de las licencias de extensiones hard habrá que pedir licencias para las extensiones webfon necesarias en el cliente.

General - Licencias

Licencias

COMUN CENTRALITA CALL_CENTER GRABADOR

Aplicación: CENTRALITA

Versión: 4

Firma: Valida

Empresa: cione

CIF: B82396235

Fecha: 31-12-2099

Hash: a5cd80ddd38e1d39901a34c99b2873bb57acd3d8d1f807d86ef9083ff702adfd

Licencia extensiones 73 Ampliación: 0 Válido hasta:

Ejemplo 1, instalación con todas las extensiones físicas con extensión webfon fija:

100 extensiones de las cuales los 100 usuarios se van a poder logar simultáneamente en Webfon (extensiones fijas o Pool):

- Pediremos 200 licencias de extensiones
[100 EXTENSIONES HARD + 100 EXTENSIONES WEBFON]

Ejemplo 2, instalación con extensiones físicas sin asignación fija de extensiones webfon compartiendo recursos:

100 extensiones de las cuales solo 25 usuarios se van a poder logar simultáneamente en Webfon (25 extensiones fijas asignadas a 25 usuarios concretos o un Pool con 25 extensiones (canales) de este modo las 100 extensiones podrán logarse pero solo 25 simultáneas):

- Pediremos 125 licencias de extensiones
[100 EXTENSIONES HARD + 25 EXTENSIONES WEBFON]

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Ejemplo 3 instalación mixta, extensiones físicas con extensión webfon fija y extensiones físicas con extensión webfon mediante pool:

100 extensiones de las cuales 5 usuarios siempre se podrán login (5 extensiones fijas para 5 usuarios concretos) y los otros 95 variarán entre 25 restantes mediante un Pool de 25 extensiones webfon (canales):

- Pediremos 130 licencias de extensiones
[100 EXTENSIONES HARD + 30 EXTENSIONES WEBFON]

Ejemplo 4, instalación sin extensiones fijas:

100 usuarios se van a poder login simultáneamente en Webfon (extensiones webfon fijas o Pool):

- Pediremos 100 licencias de extensiones
[0 EXTENSIONES HARD + 100 EXTENSIONES WEBFON]

7.4 Extensión física (opcional)

- **Extensión física:** se debe de dar de alta en un nodo de tipo "Corporativo" y provisionar en el terminal correspondiente según el método habitual de provisión de extensiones SIP corporativas.

Las extensiones SIP serán creadas con normalidad en

Dispositivos → Extensiones

No rellenar nunca el campo "Pool de extensiones" que aparece al configurar una extensión (a pesar de que tengamos un Pool configurado).

Dpto *: Nivel departamental

Grabar por enrutamiento: Sin grabación

Puesto: PORTVIVJRUIZ

Categoría *: Nacional / Permite llamar a internas y nacionales

Dispositivo *: Extensiones CORP

Pool de extensiones: Dejar vacío

Añadir Campos

Los campos con (*) son obligatorios

Cancelar Guardar

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7.5 Extensión Webfon

Las extensiones webfon son necesarias para que los usuarios se puedan logar en el webfon. A pesar de que también permiten recibir llamadas cuando son marcadas, la extensión principal de cada usuario para realizar llamadas será la extensión web que tenga asignada.

Las extensiones webfon se pueden asignar de forma fija o mediante un Pool de extensiones.

De cualquiera de las dos formas, hay que tener en cuenta que la numeración de las extensiones webfon NUNCA PUEDEN COINCIDIR con las extensiones físicas. Por lo que habrá que dejar esa numeración libre.

- **Extensión webfon:** se da de alta en un nodo de tipo "Corporativo", con los parámetros habituales dentro del menú "Vivait Call > Dispositivos > Extensiones" y teniendo en cuenta los siguientes consideraciones:
 - Su plan de marcación es específico y no determinante, ya que estas extensiones nunca se marcaran en el sistema,
 - Se le asigna la plantilla "Plantilla para Extensiones webfone",
 - No se provisionan, y
 - Consumen licencia.

Las extensiones webfon se crean en el apartado:

Vivait Call → Dispositivos → Extensiones

Como su función es decir quién puede logarse en webfon y quién no, no es necesario completar todos los datos del usuario, pero sí es importante **asignarle la Plantilla para Extensiones webfon**, en el campo "Plantilla".



Siempre hay que crear al menos una extensión webfone siguiendo estos pasos.

Terminar de completar los campos obligatorios de **Clave de registro, Sede, Dpto., etc..**

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Extensiones

Id:	52
Numero *:	49600
Nombre mostrar:	
Prioridad:	0
Plantilla *:	Plantilla para Extensiones webfon
Usuario de movilidad:	
Usuario propietario:	/
Hay movilidad:	No ▾
Telefono	/
Hay desvío incondicional por calendario:	No ▾
Desvío incondicional por calendario:	
Desvío incondicional:	
Desvío incondicional a mensa:	No ▾
Desvío no contesta:	
Desvío por no contesta a mensa:	No ▾
Desvío ocupado:	
Desvío por ocupado a mensa:	No ▾
Desvío por fuera de servicio:	
Desvío por fuera de servicio a mensa:	No ▾
Comunicaciones personales:	No ▾
Num. mostrar exterior:	
Música en espera	default / Clase por defecto
Música en retenciones	default / Clase por defecto
Clave de registro:	*****
Buzón	
Categoría de desvío *	Nacional / Permite llamar a internas y nacionales
Sede *	Sede 2 / Sede 2
Nodo	PrePro-CORP1 / Corporativo preproduccion 1
Nodo secundario	/
Dpto *:	Nivel departamental
Grabar por enrutamiento:	Sin grabación ▾

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Pool de extensiones: una forma alternativa, si se va a trabajar con “Pool de Extensiones Webfon” de crear extensiones webfon es hacerlo a la hora de incluirlas en un grupo de extensiones para webfon. Para ello dentro del menú “Vivait Call > Dispositivos > Pool de extensiones” se crea un pool con las siguientes consideraciones:

- Se debe utilizar una extensión webfon previamente creada como “Extensión base”.

Pool de extensiones

Id: 2

Nombre *: pool de homologación

Descripción: pool de homologación

Extensión base *: 2700

Extensiones: 2700, 2710

Rango de Extensiones

Extension Inicial:

Extension Final:

- Se añaden extensiones webfon previamente creadas utilizando el campo “Extensión”,



La extensión base no formará parte del pool si no se añade expresamente.



Cuando se despliega el campo extensiones aparecen todas las extensiones creadas en el sistema, no se discrimina entre las extensiones física y las extensiones webfon.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Una forma alternativa de crear un pool es:

- Se crean y se añaden extensiones webfon utilizando los campos del área “Rango de Extensiones”, donde se indica la primera y la última extensión del rango con el mismo perfil que la extensión base.

Pool de extensiones

Id:

Nombre *:

Descripción:

Extensión base *:

Extensiones:

Rango de Extensiones

Extension Inicial:

Extension Final:



El rango elegido no debe contener ninguna extensión previamente creada.



La extensión base no formará parte del pool si no se añade expresamente.

Una vez creadas estas aparecen incorporadas al pool de extensiones, campo
Vivait Call > Dispositivos > Extensiones > Pool de extensiones



Si eliminamos el valor del campo “Pool de extensiones” de la extensión webfon esta desaparece del pool previamente asignado.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Se puede crear un pool de extensiones simultaneando los dos procedimientos:

- Incorporación de extensiones previamente creadas, y
- Creación e incorporación de extensiones webfon al pool de extensiones.

Pool de extensiones

Id:

Nombre *:

Descripción:

Extensión base *: 

Extensiones: 

Rango de Extensiones

Extension Inicial:

Extension Final:

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7.6 Usuarios

7.6.1 Validación de usuarios

Para que los usuarios hagan login en el webfon, deberán verificarse sus credenciales. La elección se realiza en:

General → Usuarios → Administrar Usuarios
→ Usuario → campo Validación

Se podrán autenticar mediante los datos introducidos en el portal (Opción Vivait-Suite), mediante el servidor LDAP del cliente o ambas.

Usuarios

Usuario ACD Centralita Aprovisionamiento Mensajería

Id: 20

Nombre *: anita

Primer apellido: dinamita

Segundo apellido: canosa

DNI/CIF:

Dirección:

Código postal:

Localidad:

Correo electrónico:

Móvil:

Idioma:

Usuario *: ana

Clave *: ●●●●●●

Validación : Autenticación VIVait-Suite ▼

Depurar: Deshabilitado

Doble factor: Autenticación VIVait-Suite

Nominal: Autenticación LDAP

Gestor de identidad: Ambas

Id gestor de identidad:

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



La explicación de las diferentes opciones es la siguiente:

Autenticación Vivait-Suite

El sistema verificará la autenticidad de las credenciales a partir de los datos que se hayan introducido en el portal a la hora de crear el usuario.

Correo electrónico:

Móvil:

Idioma:

Usuario *:

Clave *:

Validación :

Depurar:

Autenticación LDAP

Para configurar el servidor LDAP del cliente, lo podremos hacer mediante el portal en:
General → Configurar LDAP:

General - Configurar LDAP

Configurar LDAP

Editar

Activar servidor:

Servidor: (Dirección IP del servidor LDAP del cliente)

Puerto: (Valor fijo)

Servidor Secundario: (Dirección IP del segundo servidor LDAP del cliente)

Puerto Secundario: (Valor fijo)

Dominio: (1)

Base DN real: (Datos proporcionados por el cliente) (1)

FMT: (1)

Filtro: (Valor fijo) (1)

Segundos: (Valor fijo)

Activar ssl:

Tipo:

Usuario consulta LDAP

Cuenta: (Datos proporcionados por el cliente)

Contraseña: (Datos proporcionados por el cliente)

VALIDADO

Cancelar Guardar

Siempre reiniciar el servicio serCen cuando hagamos cambios en LDAP:

```
/etc/init.d/serCen stop
```

```
/etc/init.d/serCen start
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Ambas

El sistema de validación de credenciales se realizará a partir de los datos insertados en el portal y del servidor LDAP de cliente. La contraseña de un usuario registrada en el portal no tiene por qué coincidir con la de LDAP.

Usuarios

Usuario ACD Centralita Aprovisionamiento Mensajería Extensiones Meet

Id: 20
 Nombre *: anita
 Primer apellido: dinamita
 Segundo apellido: canosa
 DNI/CIF:
 Dirección:
 Código postal:
 Localidad:
 Correo electrónico:
 Móvil:
 Idioma:
 Usuario *: ana
 Clave *:
 Validación : Ambas
 Depurar: Si
 Doble factor: Ninguno
 Nominal: No
 Gestor de identidad: Azure
 Id gestor de identidad:
 Dpto *: Nivel departamental

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7.6.2 Usuario con Extensión Webfon fija

Este usuario cuando se loga en **VIVAit** Call Web siempre tiene asignada la misma extensión webfon. Para ello se seguirá el siguiente procedimiento:

Creamos un usuario por cada extensión física que vaya a utilizar webfon en:

General → Usuarios → Administrar Usuarios

Pestaña Usuario

Usuarios

Usuario ACD Centralita Aprovisionamiento Mensajería Extensiones Meet

Id: 20

Nombre *: anita **-(Nombre de Usuario en webfon)**

Primer apellido: dinamita

Segundo apellido: canosa

DNI/CIF:

Dirección:

Código postal:

Localidad:

Correo electrónico: **(Email del Usuario, aquí se enviara el PIN del doble factor)**

Móvil:

Idioma:

Usuario *: ana **(Nombre para logarse en webfon (será el sAMAccountName del LDAP))**

Clave *: **(Clave para Autenticación Vivait-Suite, si el por LDAP no afecta)**

Validación : Autenticación VIVait-Suite **(Elegir modo de autenticación)**

Depurar: Si

Doble factor: Ninguno **(Elegir "EMAIL" si se va a enviar el PIN del doble factor por correo)**

Nominal: No

Gestor de identidad: Azure

Id gestor de identidad:

Dpto *: Nivel departamental

Los campos con (*) son obligatorios

Cancelar Guardar

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Pestaña Centralita

Usuarios

Usuario ACD Centralita Aprovisionamiento Mensajería Extensiones Meet

Nombre mostrar: **Nombre de usuario a mostrar**

Desvío no login mensa: No ▾

Desvío no login:

Desvío por no contesta a mensa: No ▾

Desvío no contesta:

Desvío por ocupado a mensa: No ▾

Desvío ocupado:

Desvío incondicional a mensa: No ▾

Desvío incondicional:

Hay desvío incondicional por calendario: No ▾

Desvío incondicional por calendario:

Número llamada externa:

Número login: **Extensión web**

Clave login: **No aplica**

Comunicaciones personales: No ▾

Grabar por enrutamiento: Sin grabación ▾

Categoría: Nacional / Permite llamar a internas y nacionales

Desvío categoría: Nacional / Permite llamar a internas y nacionales

Grupo movilidad: /

Pool de extensiones Webphone:

Extensiones Webphone: **Extensión webfone**

Número de líneas Webphone: -1

Salas de conferencia Webphone: -1

Conferencias BEEP: No ▾

Mover Cola: No ▾

- 2.- En el nodo Webrtc editar el fichero /etc/janus/vivait.plugin.webfon.jcfg
[root@vivait-meet-install](#):~#vi /etc/janus/vivait.plugin.webfon.jcf

Asignar a la línea colas_limpiar_siempre el valor false:

```
colas_limpiar_siempre = false
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



3.- A continuación se reinicia el servicio janus:

```
root@vivait-meet-install:~/etc/init.d/janus status
Status of janus: janus is running.
root@vivait-meet-install:~/etc/init.d/janus stop
Stopping janus: janus.
root@vivait-meet-install:~/etc/init.d/janus start
Starting janus: janus.
```



Se recomienda que el número de login del usuario sea el mismo que el de la extensión física sobre la que se va a aplicar la movilidad.

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7.6.3 Usuario con Pool de extensiones Webfon

Este usuario cuando se loga en VIVAit Call Web el sistema le asigna una extensión libre del pool de extensiones al que pertenece. Para ello se debe seguir el siguiente procedimiento:

1. Una vez configurado el usuario de la forma habitual procederemos a asignarle un pool de extensiones webfon, previamente creada, en el campo "Pool de extensiones Webphone" que se encuentra en la pestaña

Usuarios > Centralita

Usuarios

Usuario ACD Centralita Aprovisionamiento Mensajería Extensiones Meet

Nombre mostrar:

Desvío no login mensa:

Desvío no login:

Desvío por no contesta a mensa:

Desvío no contesta:

Desvío por ocupado a mensa:

Desvío ocupado:

Desvío incondicional a mensa:

Desvío incondicional:

Hay desvío incondicional por calendario:

Desvío incondicional por calendario:

Número llamada externa:

Número login:

Clave login:

Comunicaciones personales:

Grabar por enrutamiento:

Categoría:

Desvío categoría:

Grupo movilidad:

Pool de extensiones Webphone:

Extensiones Webphone:

Número de líneas Webphone:

Salas de conferencia Webphone:

Conferencias BEEP:

Mover Cola:

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



- 2.- En el nodo Webrtc editar el fichero `/etc/janus/vivait.plugin.webfon.jcfg`

```
root@vivait-meet-install:~#vi /etc/janus/vivait.plugin.webfon.jcf
```

Asignar a la línea `colas_limpiar_siempre` el valor `true`:

```
colas_limpiar_siempre = true
```

- 3.- A continuación se reinicia el servicio `janus`:

```
root@vivait-meet-install:~/#etc/init.d/janus status
Status of janus: janus is running.
root@vivait-meet-install:~#etc/init.d/janus stop
Stopping janus: janus.
root@vivait-meet-install:~#etc/init.d/janus start
Starting janus: janus.
```

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



7.6.4 Usuario con Doble Factor

Si además de introducir las credenciales habituales de usuario y password se necesita alguna medida de seguridad adicional se puede implementar el doble factor. Esta medida de seguridad adicional, en el momento actual tiene la opción de enviar un código pin a una cuenta de correo, o mediante la funcionalidad TOTP que genera claves por el dispositivo de autenticación –app en el móvil-. En ambos casos el PIN Usuario con Extensión Webfon fija

Este usuario cuando se loga en **VIVAit** Call Web siempre tiene asignada la misma extensión webfon. Para ello se seguirá el siguiente procedimiento:

Crearemos un usuario por cada extensión física que vaya a utilizar webfon en:

General → Usuarios → Administrar Usuarios

Pestaña Usuario

Usuarios

Usuario ACD Centralita Aprovisionamiento Mensajería Extensiones Meet

Id: 20

Nombre *: anita **{(Nombre de Usuario en webfon)}**

Primer apellido: dinamita

Segundo apellido: canosa

DNI/CIF:

Dirección:

Código postal:

Localidad:

Correo electrónico: **(Email del Usuario, aquí se enviara el PIN del doble factor)**

Móvil:

Idioma:

Usuario *: ana **(Nombre para logarse en webfon (será el sAMAccountName del LDAP))**

Clave *: ***** **(Clave para Autenticación Vivait-Suite, si el por LDAP no afecta)**

Validación : Autenticación VIVait-Suite **(Elegir modo de autenticación)**

Depurar: Si

Doble factor: Ninguno **(Elegir "EMAIL" si se va a enviar el PIN del doble factor por correo)**

Nominal: No

Gestor de identidad: Azure

Id gestor de identidad:

Dpto *: Nivel departamental

Los campos con (*) son obligatorios

Cancelar Guardar

Autor: Bartosz Hubert Grzech	Asunto: Funcionalidad del nodo WebRTC
Revisado:	Fecha: 05 de julio de 2022



Pestaña Centralita

Usuarios

Usuario ACD Centralita Aprovisionamiento Mensajería Extensiones Meet

Nombre mostrar: **Nombre de usuario a mostrar**

Desvío no login mensa: No ▾

Desvío no login:

Desvío por no contesta a mensa: No ▾

Desvío no contesta:

Desvío por ocupado a mensa: No ▾

Desvío ocupado:

Desvío incondicional a mensa: No ▾

Desvío incondicional:

Hay desvío incondicional por calendario: No ▾

Desvío incondicional por calendario:

Número llamada externa:

Número login: **Extensión web**

Clave login: **No aplica**

Comunicaciones personales: No ▾

Grabar por enrutamiento: Sin grabación ▾

Categoría: Nacional / Permite llamar a internas y nacionales

Desvío categoría: Nacional / Permite llamar a internas y nacionales

Grupo movilidad: /

Pool de extensiones Webphone:

Extensiones Webphone: **Extensión webfone**

Número de líneas Webphone: -1

Salas de conferencia Webphone: -1

Conferencias BEEP: No ▾

Mover Cola: No ▾

- 2.- En el nodo Webrtc editar el fichero `/etc/janus/vivait.plugin.webfon.jcfg`
- ```
root@vivait-meet-install:~#vi /etc/janus/vivait.plugin.webfon.jcf
```

Asignar a la línea `colas_limpiar_siempre` el valor `false`:

```
colas_limpiar_siempre = false
```

|                              |                                       |
|------------------------------|---------------------------------------|
| Autor: Bartosz Hubert Grzech | Asunto: Funcionalidad del nodo WebRTC |
| Revisado:                    | Fecha: 05 de julio de 2022            |



### 3.- A continuación se reinicia el servicio janus:

```
root@vivait-meet-install:~/etc/init.d/janus status
Status of janus: janus is running.
root@vivait-meet-install:~/etc/init.d/janus stop
Stopping janus: janus.
root@vivait-meet-install:~/etc/init.d/janus start
Starting janus: janus.
```



Se recomienda que el número de login del usuario sea el mismo que el de la extensión física sobre la que se va a aplicar la movilidad.

|                              |                                       |
|------------------------------|---------------------------------------|
| Autor: Bartosz Hubert Grzech | Asunto: Funcionalidad del nodo WebRTC |
| Revisado:                    | Fecha: 05 de julio de 2022            |



### 7.6.3 Usuario con Pool de extensiones Webfon

Este usuario cuando se loga en VIVAit Call Web el sistema le asigna una extensión libre del pool de extensiones al que pertenece. Para ello se debe seguir el siguiente procedimiento:

2. Una vez configurado el usuario de la forma habitual procederemos a asignarle un pool de extensiones webfon, previamente creada, en el campo "Pool de extensiones Webphone" que se encuentra en la pestaña

Usuarios > Centralita

**Usuarios**

Usuario ACD Centralita Aprovisionamiento Mensajería Extensiones Meet

Nombre mostrar:

Desvío no login mensa:

Desvío no login:

Desvío por no contesta a mensa:

Desvío no contesta:

Desvío por ocupado a mensa:

Desvío ocupado:

Desvío incondicional a mensa:

Desvío incondicional:

Hay desvío incondicional por calendario:

Desvío incondicional por calendario:

Número llamada externa:

Número login:

Clave login:

Comunicaciones personales:

Grabar por enrutamiento:

Categoría:

Desvío categoría:

Grupo movilidad:

Pool de extensiones Webphone:

Extensiones Webphone:

Número de líneas Webphone:

Salas de conferencia Webphone:

Conferencias BEEP:

Mover Cola:

|                              |                                       |
|------------------------------|---------------------------------------|
| Autor: Bartosz Hubert Grzech | Asunto: Funcionalidad del nodo WebRTC |
| Revisado:                    | Fecha: 05 de julio de 2022            |



- 2.- En el nodo Webrtc editar el fichero `/etc/janus/vivait.plugin.webfon.jcfg`

```
root@vivait-meet-install:~#vi /etc/janus/vivait.plugin.webfon.jcf
```

Asignar a la línea `colas_limpiar_siempre` el valor `true`:

```
colas_limpiar_siempre = true
```

- 3.- A continuación se reinicia el servicio `janus`:

```
root@vivait-meet-install:~/etc/init.d/janus status
Status of janus: janus is running.
root@vivait-meet-install:~#etc/init.d/janus stop
Stopping janus: janus.
root@vivait-meet-install:~#etc/init.d/janus start
```

Starting `janus`: `janus`. debe ser introducido en Vivait Call Web después de que el usuario introduzca correctamente sus credenciales Usuario/Password.

Para configurar esta funcionalidad se seguirán los siguientes pasos:

**Paso 1.-** Para el caso de PIN por email: configurar la cuenta de correo a la que se enviará el pin de acceso, este dato se configura en el campo "Correo electrónico" del portal de administración en la pestaña "Usuarios > usuario".

Para el caso de PIN por TOTP: configurar el teléfono móvil del usuario, este dato se configura en el campo "Móvil" del portal de administración en la pestaña "Usuarios > usuario".

**Paso 2.-** En la pestaña "Usuarios > usuario" asignar al campo "Doble factor" el valor:  
 "Email" – para el caso de PIN por email,  
 "TOTP" – para el caso de PIN por TOTP.

**Paso 3.-** En el nodo Webrtc editar el fichero `/etc/MDtel/serCen.conf`

```
root@vivait-meet-install:~#vi /etc/MDtel/serCen.conf
```

En las líneas `smtp` se configuran los datos del servidor de correo:

```
smtps_usa = false
smtp_host = "lab.mdnova.local"
smtp_puerto = 25
smtp_usuario = ""
smtp_clave = ""
```

|                              |                                       |
|------------------------------|---------------------------------------|
| Autor: Bartosz Hubert Grzech | Asunto: Funcionalidad del nodo WebRTC |
| Revisado:                    | Fecha: 05 de julio de 2022            |



Y en las líneas “flfactor\_xx\_ip” y “flfactor\_xx\_msk” las redes desde las que no se pedirá doble factor

```
flfactor_00_ip = "172.25.1.0"
flfactor_00_msk = "255.255.255.0"
flfactor_01_ip = ""
flfactor_01_msk = ""
flfactor_02_ip = ""
flfactor_02_msk = ""
flfactor_03_ip = ""
flfactor_03_msk = ""
```

**Paso 4.-** A continuación se reinicia el servicio serCen:

```
root@vivait-meet-install:~#/etc/init.d/serCen status
serCen is running
root@vivait-meet-install:~#/etc/init.d/serCen stop
Shutting down serCen
root@vivait-meet-install:~#/etc/init.d/serCen start
Starting serCen
```

|                              |                                       |
|------------------------------|---------------------------------------|
| Autor: Bartosz Hubert Grzech | Asunto: Funcionalidad del nodo WebRTC |
| Revisado:                    | Fecha: 05 de julio de 2022            |



### 7.6.5 Usuario con extensión física (multidispositivo)

Opcionalmente si se va a trabajar con extensión física o con webfon, según las circunstancias, cuando tengamos creado el USUARIO que se asignará a esta extensión, tendremos que volver a la extensión física y asignárselo en el campo "Usuario propietario":

**Extensiones**

|                                          |                                        |
|------------------------------------------|----------------------------------------|
| Id:                                      | 7                                      |
| Numero *:                                | 40102                                  |
| Nombre mostrar:                          | pepe perez                             |
| Prioridad:                               | 0                                      |
| Plantilla *                              | Plantilla para GrandStream con qualify |
| Usuario de movilidad:                    |                                        |
| <b>Usuario propietario</b>               | <b>anita dinamita canosa / ana</b>     |
| Hay movilidad:                           | Si ▼                                   |
| Telefono                                 | Grandstream GRP2604 / C074AD34BBE2     |
| Hay desvío incondicional por calendario: | Si ▼                                   |

|                              |                                       |
|------------------------------|---------------------------------------|
| Autor: Bartosz Hubert Grzech | Asunto: Funcionalidad del nodo WebRTC |
| Revisado:                    | Fecha: 05 de julio de 2022            |



**Usuarios**

Usuario ACD Centralita Aprovisionamiento Mensajería Extensiones Meet

Id: 20

Nombre \*: anita

Primer apellido: dinamita

Segundo apellido: canosa

DNI/CIF:

Dirección:

Código postal:

Localidad:

Correo electrónico: demo02-out@lab.mdnova.local

Móvil: 650666444

Idioma: Español

Usuario \*: ana

Clave \*: \*\*\*\*\*

Validación: Autenticación VIVait-Suite

Depurar: Si

Doble factor: TOTP

Nominal: No

Gestor de identidad: Azure

Id gestor de identidad:

Dpto \*: Nivel departamental

Los campos con (\*) son obligatorios

Cancelar Guardar

**usuario.2factor**

Mostrar 25 registros

| Descripción |
|-------------|
| Email       |
| Ninguno     |
| TOTP        |

Mostrando registros del 1 al 3 de un total de 3 registros

**Paso 3.-** En el nodo Webrtc editar el fichero /etc/MDtel/serCen.conf

```
root@vivait-meet-install:~#vi /etc/MDtel/serCen.conf
```

En las líneas smtp se configuran los datos del servidor de correo:

```
smtps_usa = false
smtp_host = "lab.mdnova.local"
smtp_puerto = 25
smtp_usuario = ""
smtp_clave = ""
```

|                              |                                       |
|------------------------------|---------------------------------------|
| Autor: Bartosz Hubert Grzech | Asunto: Funcionalidad del nodo WebRTC |
| Revisado:                    | Fecha: 05 de julio de 2022            |



Y en las líneas "f1factor\_xx\_ip" y "f1factor\_xx\_msk" las redes desde las que no se pedirá doble factor

```
f1factor_00_ip = "172.25.1.0"
f1factor_00_msk = "255.255.255.0"
f1factor_01_ip = ""
f1factor_01_msk = ""
f1factor_02_ip = ""
f1factor_02_msk = ""
f1factor_03_ip = ""
f1factor_03_msk = ""
```

**Paso 4.-** A continuación se reinicia el servicio serCen:

```
root@vivait-meet-install:~#/etc/init.d/serCen status
serCen is running
root@vivait-meet-install:~#/etc/init.d/serCen stop
Shutting down serCen
root@vivait-meet-install:~#/etc/init.d/serCen start
Starting serCen
```

|                              |                                       |
|------------------------------|---------------------------------------|
| Autor: Bartosz Hubert Grzech | Asunto: Funcionalidad del nodo WebRTC |
| Revisado:                    | Fecha: 05 de julio de 2022            |



### 7.6.5 Usuario con extensión física (multidispositivo)

Opcionalmente si se va a trabajar con extensión física o con webfon, según las circunstancias, cuando tengamos creado el USUARIO que se asignará a esta extensión, tendremos que volver a la extensión física y asignárselo en el campo "Usuario propietario":

**Extensiones**

|                                          |                                        |
|------------------------------------------|----------------------------------------|
| Id:                                      | 7                                      |
| Numero *:                                | 40102                                  |
| Nombre mostrar:                          | pepe perez                             |
| Prioridad:                               | 0                                      |
| Plantilla *                              | Plantilla para GrandStream con qualify |
| Usuario de movilidad:                    |                                        |
| <b>Usuario propietario</b>               | <b>anita dinamita canosa / ana</b>     |
| Hay movilidad:                           | Si ▼                                   |
| Telefono                                 | Grandstream GRP2604 / C074AD34BBE2     |
| Hay desvío incondicional por calendario: | Si ▼                                   |