

## PROYECTOS DE ENAIRE (SACTA, COMETA e ICARO)

# RESUMEN PROCEDIMIENTOS XR12 (IT) (Procedimientos XR12)

Nº Exp.: xxx

Doc. Nº: xxx

Ed./Rv.: 1

Fecha: 23/07/24

Nº LDEC: xxx

Preparado para:

**Enaire - División de Automatización**

Parque Empresarial Las Mercedes  
Edificio 7 – Planta baja  
Avda. de Aragón, 330  
28022 - Madrid

Preparado por:

**Indra Sistemas**

Parque Empresarial San Fernando (Edificio Kenia)  
Avenida de Castilla, 2  
28830 San Fernando de Henares  
Madrid

Este documento ha sido realizado por Indra Sistemas para Enaire en el marco del expediente cuyo número se indica arriba, no pudiendo ser usado con fines distintos de los que ha sido entregado, ni reproducido total o parcialmente, ni transmitido o comunicado a ninguna persona sin autorización expresa de Enaire y de Indra Sistemas.

## ÍNDICE

<b>1.</b>	<b>INTRODUCCIÓN .....</b>	<b>1-1</b>
1.1.	OBJETIVO: DESPLIEGUE DE MUEBLES IFOCUS .....	1-1
1.2.	TAREAS DE MANTENIMIENTO DE MUEBLES IFOCUS .....	1-2
<b>2.</b>	<b>RESUMEN DE PROCEDIMIENTOS DE USO DEL XR12 .....</b>	<b>2-3</b>
2.1.	ÍNDICE GRÁFICO. ....	2-3
2.1.1.	Administración XR12: Anfitrión .....	2-3
2.1.2.	Uso XR12: Virtualización .....	2-4
2.2.	ADMINISTRACIÓN XR12: ANFITRIÓN. ....	2-5
2.2.1.	Administración: Consola .....	2-5
2.2.2.	Demonio: Servidor y Recursos I/O.....	2-7
2.2.3.	SystemD: Disparador de Demonios .....	2-9
2.2.4.	NetworkManager: Gestión de Red.....	2-17
2.2.5.	Chrony: Sincronismo .....	2-20
2.3.	USO XR12: VIRTUALIZACIÓN. ....	2-22
2.3.1.	Hipervisor: Entorno de Ejecución de Máquinas Virtuales .....	2-22
2.3.2.	Máquina Virtual: Gestión de Instancias .....	2-25
2.3.2.1.	Gestión Instancias Máquina Virtual.....	2-25
2.3.2.2.	Instancias de Cometa.....	2-30
2.3.2.3.	Instancias de OUCS .....	2-34
2.3.2.4.	Instancias de POS.....	2-37

## 1. INTRODUCCIÓN

### 1.1. OBJETIVO: DESPLIEGUE DE MUEBLES IFOCUS

Tal como muestra la figura 1.1-1, los muebles iFocus son controlados desde ordenadores XR12 (dos para planificador, dos para ejecutivo) que contienen las aplicaciones de tráfico aéreo encapsuladas dentro de máquinas virtuales. En otras palabras, el ordenador iFocus son varias máquinas anfitrionas XR12 especializadas en gestionar instancias de máquinas virtuales que encapsulan cada aplicación de tráfico aéreo, con todas sus dependencias (sistema operativo y librerías).

Como indica la imagen, consola y ordenador están en distintas salas, conectadas a través de cables de fibra óptica. Detrás de cada mueble iFOCUS, hay un armario con agregadores que multiplexan las señales de las distintas conexiones que viene del mueble sobre una misma fibra óptica. En la sala de cómputo, hay un armario por mueble iFOCUS que tiene sus cuatro XR12 en la parte inferior y los disgregadores en la parte superior que demultiplexan esas señales de la fibra óptica para llevarlas a los ordenadores XR12.

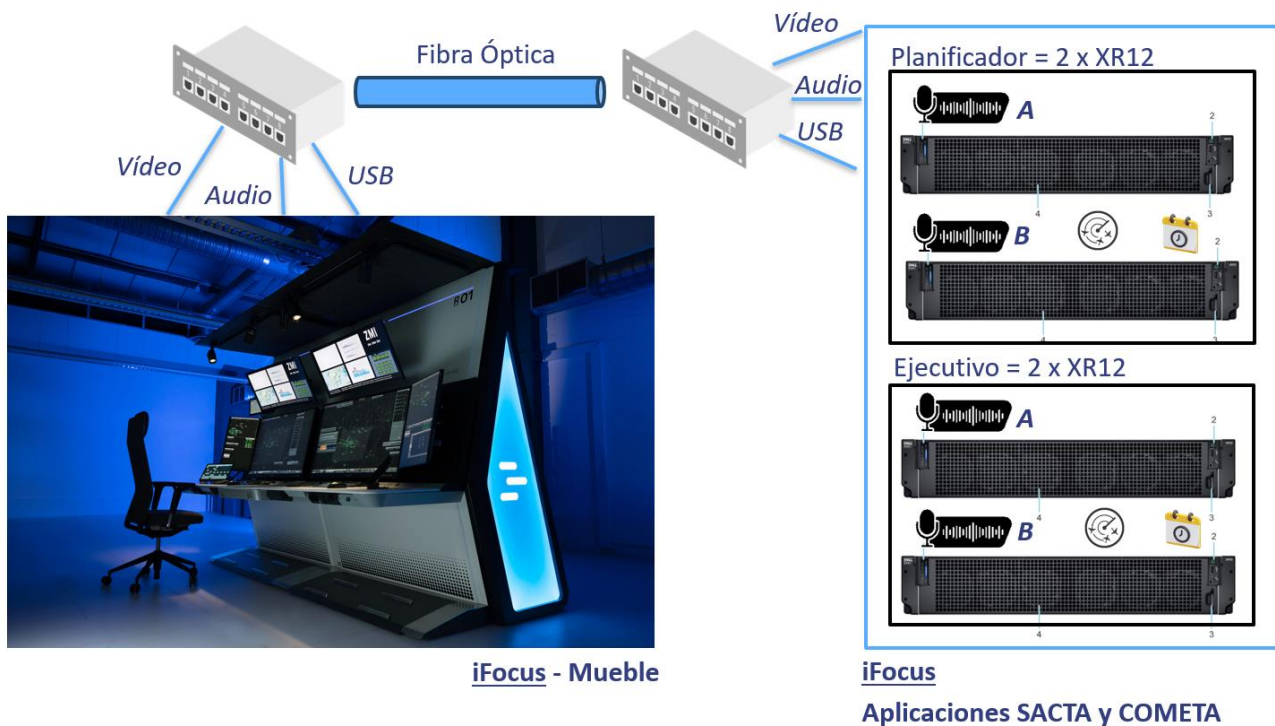


Figura 1.1-1. Esquema Simplificado de los Muebles iFocus

Tal como muestra la figura 1.1-2, Enaire libera versiones tanto de aplicativo como de máquina virtual, **los centros de control son responsables de realizar un proceso de puesta en marcha**, es decir, inyectarles aplicación a las máquinas virtuales, **y a partir de ahí, propagarlas por todos los muebles** que gestione el centro de control. Has tres tipos de chasis virtual, cada uno con su versión de RedHat Enterprise Linux (RHEL): RHEL6 para POS, RHEL7 para OUCS y RHEL8 para CWP. El anfitrión trabaja en RHEL 8.6

Ha de llevarse un control de versiones de las instancias de máquina virtual que están ejecutándose en cada mueble iFocus del centro.

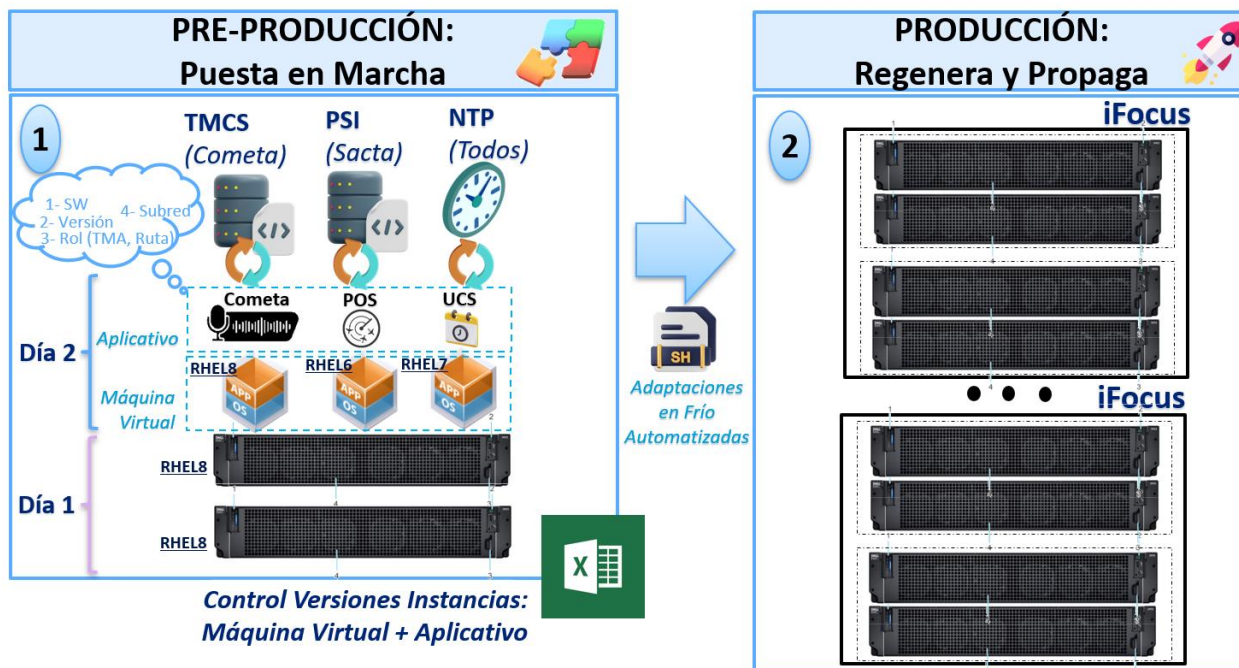


Figura 1.1-2. Proceso de Despliegue de los Muebles iFocus

## 1.2. TAREAS DE MANTENIMIENTO DE MUEBLES IFOCUS

Dos son las tareas de mantenimiento asociadas al mueble iFOCUS:





















- **TAREA 1 – CONTROL DEL MUEBLE IFOCUS:** consiste en automatizar los diagnósticos de los periféricos para hacer verificaciones periódicas de su estado e ir clasificando y guardando los distintos errores que vayan surgiendo, o sea, ir creando una base de datos errores del mueble iFOCUS clasificada de tal forma que sea fácil encontrar cómo resolver cada caso (ejemplo de cómo clasificar errores: por periférico y por puntos de sondeo dentro de su cadena de eventos, identificando planes de acción para cada error). Clave es protocolizar los diagnósticos, tanto para su automatización como para la apertura de casos a través de los cuales ir creando esa base de datos de errores, diagnósticos y resoluciones para el mueble iFOCUS fácil de consultar (RedHat tiene la suya, pero es genérica).
- **TAREA 2 – MANTENIMIENTO DE INSTANCIAS DE MÁQUINA VIRTUAL:** consiste en tener un repositorio de máquinas virtuales con aplicativo instalado (tanto TMA como Ruta), listas para ser propagadas. Además, hay que tener controlados los parámetros de instancia en cada posición de cada mueble iFOCUS. Esos parámetros de instancia son los que permiten la instanciación automatizada de esas máquinas virtuales de referencia en cada mueble del centro de control (las instancias no guardan datos, sino que se sincronizan con servidores externos). Clave es la gestión de versiones, tanto el repositorio de máquinas virtuales de referencia, como parámetros instancia en cada mueble; tal vez un sistema de control de versiones pueda facilitar estas tareas de mantenimiento.

## 2. RESUMEN DE PROCEDIMIENTOS DE USO DEL XR12

### 2.1. ÍNDICE GRÁFICO.




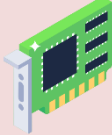







#### 2.1.1. Administración XR12: Anfitrión

Tabla 2.1-1: Administración XR12: Anfitrión.

ANFITRIÓN					
					
<b>ADMIN. Consola</b>	<b>iDRAC Web</b> ✓ Acceso -mini USB -ethernet ✓ Cambio IP ✓ Verificación estado ✓ Acceso concurrente ✓ Configurar NTP	<b>iDRAC Consola</b> ✓ Acceso -mini USB -ethernet ✓ Cambio IP ✓ Verificación estado ✓ Acceso concurrente ✓ Configurar NTP	<b>SSH</b> ✓ MobaXterm ✓ FileZilla	<b>Propiedades</b> ✓ Sesión -region -teclado -hora	<b>Software</b> ✓ Paquetes ✓ Repositorio
					
<b>DEMONIO Servidor y Recursos I/O</b>	<b>Proceso Config</b> ✓ Localizar Estructura	<b>Proceso Actividad</b> ✓ Ejecución (ps) ✓ Logs (/var/log)	<b>Recursos I/O Data (/var)</b> ✓ Tipología Datos	<b>Recursos I/O Sockets (ss)</b> ✓ Conexiones	
					
<b>SYSTEMD Disparador Demonios</b>	<b>Units (/lib/systemd/system)</b> ✓ Estructura Unit -editar Unit -base datos Units -estructura del arranque ✓ Service ✓ Object ✓ Target	<b>Systemctl</b> ✓ Configurar Demonio ✓ Ciclo Vida Units -start/stop -enable/disable -mask/unmask -status ✓ Niveles Ejecución -target -dependencias -default target ✓ Análisis Arranque -systemd-analyze	<b>Journalctl (/run/journal)</b> ✓ Configurar -efimero -persistente ✓ Filtrar -unit -prioridad -tiempo real -boot -kernel -intervalo -extendido ✓ Espacio Disco	<b>Audit (/var/log/audit)</b> ✓ Aplicar Reglas -augenrules -auditctl ✓ Resultados -ausearch -aureport -aullast ✓ Sintaxis 1: Watch ✓ Sintaxis 2: Syscall	<b>Rsyslog (/var/log)</b> ✓ Filtrar - messages - anaconda/syslog - secure - boot.log
					
<b>NETWORK MANAGER Configuración Red</b>	<b>Dispositivo (/sys/class/net)</b> ✓ Tráfico Red ✓ UDev Persistent Rules	<b>Conexión (/etc/sysconfig/network-scripts)</b> ✓ Simple: Ethernet ✓ Simple: VLAN ✓ Master-Slave: Bond ✓ Master-Slave: Bridge ✓ Master-Slave: VLAN Bridge			
					
<b>CHRONY Sincronismo</b>	<b>Cliente</b> ✓ Configurar ✓ Sincronizar	<b>Servidor</b> ✓ Configurar ✓ Sincronizar			

## 2.1.2. Uso XR12: Virtualización

Tabla 2.1-2: Uso XR12: Virtualización.




VIRTUALIZACIÓN					
					
<b>HIPERVISOR</b> Recursos Anfitrión	<b>Libvirt</b> ✓ Instalación ✓ Monitorización CPU y RAM Anfitrión -free, ps, top	<b>QEMU</b> ✓ Carpetas ✓ Trazas	<b>HW Devices</b> ✓ Enchufar USB -Tabla - UDEV Rules -Detección	<b>Storage (Pool)</b> ✓ Creación ✓ Monitorización -df, lsblk, blkid, du -virsh pool-list -virsh vol-list	<b>Network</b> ✓ Regeneración Overlay
					
<b>MÁQUINA VIRTUAL</b> Gestión Instancias	<b>INSTANCIA</b> ✓ Transiciones -define -start -shutdown/destroy -undefine  ✓ Interacciones en frío -virt-copy-in -virt-copy-out  ✓ Interacciones en caliente -virsh console -ssh -virsh dump-xml  ✓ Modificaciones en frío -virt-customize -guestfish -virt-edit  ✓ Modificaciones en caliente -virt-manager  ✓ Automatización Modificaciones en Frío -Script dentro Instancia -Script fuera de Instancia  ✓ Monitorización CPU y RAM (en caliente) -virt-top  ✓ Monitorización Almacenamiento (en frío) -qemu-info -virt-alignment-scan -virt-df	<b>POS</b> ✓ Parar  ✓ Puesta en Marcha -PASO 1: Entorno -PASO 2: Aplicativo -PASO 3: Arranque  ✓ Propagación -PASO 1: Entorno -PASO 2: Verifica -PASO 3: Arranque  ✓ Desfragmenta Disco Virtual	<b>CWP</b> ✓ Parar  ✓ Puesta en Marcha -PASO 1: Entorno -PASO 2: Aplicativo -PASO 3: Arranque  ✓ Propagación -PASO 1: Entorno -PASO 2: Verifica -PASO 3: Arranque  ✓ Desfragmenta Disco Virtual	<b>OUCS</b> ✓ Parar  ✓ Puesta en Marcha -PASO 1: Entorno -PASO 2: Aplicativo -PASO 3: Arranque  ✓ Propagación -PASO 1: Entorno -PASO 2: Verifica -PASO 3: Arranque  ✓ Desfragmenta Disco Virtual	



## 2.2. ADMINISTRACIÓN XR12: ANFITRIÓN.

### 2.2.1. Administración: Consola

Tabla 2.2-1: Procedimientos sobre Consola SSH.




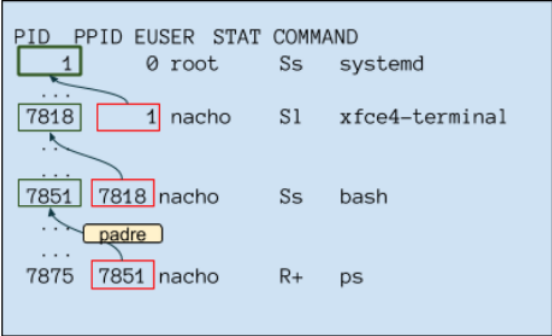

ANFITRIÓN: Consola - Acceso	
	miniUSB: https:// 169.254.0.3 root / calvin
	Ethernet: https:// 192.168.0.120 o https://<dir. Enaire> root / calvin
	ssh root@192.168.0.120 o root@<dir. Enaire> [root]# nohup idrac.sh > idrac_status.txt [root]# cat idrac.sh #!/bin/bash #----- # ENTORNO #----- # 1.- Lista de Maquinas de Las que ver el estado de su hardware IDRAC=( "172.30.237.255" \ "192.168.1.2" )  # 2.- Guardar Snippets en Arrays (para reusar sintaxis ya validada) STATUS=( 'getsvctag' 'getsysinfo' 'getsensorinfo' ' nicstatistics' ) STORAGE=( 'storage get pdisks' \ 'storage get pdisks --refkey Enclosure.Internal.0-0:RAID.Slot.3-1 -o -p status,state,name' \ 'storage get controllers -o' \ 'storage get vdisks -o -p layout,status' \ 'storage get enclosures -o' \ ) INVENTORY=( 'getsel' 'swinventory' 'hwinventory' \ 'getremoteservicesstatus' )  # 3.- Combinar Snippets para ejecutar lo que necesite INSTRUCCIONES=( "\${STATUS[@]}" "\${STORAGE[@]}" \ "\${INVENTORY[@]}" ) #----- # ENVIO INSTRUCCIONES #----- for IP in "\${IDRAC[@]}" ; do echo "#####" echo "# MAQUINA IP = \$IP" echo "#####" for CMD in "\${INSTRUCCIONES[@]}" ; do echo "Instruccion iDRAC: \$CMD" sshpass -p "calvin" ssh -o 'StrictHostKeyChecking no' root@\$IP "\$CMD" echo "==== ..... ===" done done done
	ssh root@<dir. Gestión> Houston
	


	<h3>Entorno de Sesión</h3> <p>___ 1.- Status</p> <pre>localectl status</pre> <p>___ 2.- Configurando fecha y hora</p> <pre>date timedatectl</pre> <p>___ 3.- Configurando region (efimera, solo esta sesión)</p> <pre>localectl list-locales   grep -i es_ localectl status localectl set-locale LANG=es_ES.UTF-8</pre> <p>___ 4.- Configurando teclado (efimera, solo esta sesión)</p> <pre>localectl list-keymaps   grep -i es localectl set-keymap es</pre> <p>___ 5.- Configurando teclado en Desktop(efimera, solo esta sesión)</p> <pre>localectl list-keymaps   grep -i es localectl --no-convert set-x11-keymap es</pre>
	<h3>Paquetes</h3> <p>___ 1.- Instalación Paquetes</p> <pre>dnf search virt-manager dnf install virt-manager dnf remove virt-manager</pre> <p>___ 2.- Estructura de un demonio</p> <pre>dnf provides chronyd dnf repoquery --list chrony-4.1-1.el8.x86_64</pre> <h3>Repositorio</h3> <p>___ 1.- Creacion</p> <pre>vi /etc/yum.repos.d/iso.repo [InstallMedia] name=Red Hat Enterprise Linux 8.6.0 - BaseOS mediaid=None metadata_expire=-1 gpgcheck=0 cost=500 baseurl=file:///run/media/pangea/RHEL-8-6-0-BaseOS-x86_64/BaseOS/ gpgkey=file:///run/media/pangea/RHEL-8-6-0-BaseOS-x86_64/RPM-GPG-KEY-redhat-release</pre> <p>[InstallMedia-AppStream]</p> <pre>name=Red Hat Enterprise Linux 8.6.0 - AppStream mediaid=None metadata_expire=-1 gpgcheck=0 cost=500 baseurl=file:///run/media/pangea/RHEL-8-6-0-BaseOS-x86_64/AppStream/ gpgkey=file:///mnt/pki/rpm-gpg/RPM-GPG-KEY-redhat-release</pre> <p>___ 2.- Verificacion</p> <pre>dnf repolist</pre>



## 2.2.2. Demonio: Servidor y Recursos I/O

Tabla 2.2-2: Ejercicios sobre Estructura de Demonios.

ANFITRIÓN: Demonio – Unidad de Ejecución	
	<b>Proceso: Etiquetado Proceso</b>  ____ 1.- Fichero PID, LOCK <pre>ls /run   grep -E 'pid lock'</pre> <pre>cat /run/libvirtd.pid</pre> ____ 3.- Localizar proceso al que pertenece <pre>ps -aux   grep 221286</pre> ____ 4.- Localizar fichero configuración <pre>ls   grep libvirtd</pre> ____ 5.- Localizar configuraciones, ejecutable y librerías <pre>ls /etc/libvirtd</pre> <pre>ls /usr/sbin/libvirtd</pre> ____ 6.- Localizar sockets <pre>ss -tupln   grep libvirtd</pre> <pre>ss -xpln   grep libvirtd</pre>
	<b>Recursos I/O: Módulos</b>  ____ 1.- Dispositivos (/dev) <pre>ls -la /dev/sd*</pre> ____ 2.- Modulo asociado al dispositivo (/dev) <pre>cat /proc/devices   grep 8</pre> <pre>lsmod   grep sd</pre>
	<b>Proceso: Actividad</b>  ____ 1.- Ejecucion <pre>pstree</pre> <pre>ps -faux</pre> <pre>ps -ax -o pid,ppid,%cpu,%mem,psr,sgi_p,state,comm</pre> <div data-bbox="282 1489 836 1823"></div> ____ 2.- Scheduler (afinidad) <pre>cat /proc/cmdline # Nivel Kernel</pre> <pre>cat /etc/systemd/system.conf # Nivel de Systemd</pre> <pre>taskset -pc 493618 # Afinidad de Proceso</pre>
	

	<pre># Mover proceso a la CPU 3 taskset -pc 3 493618  # Estructura de Memoria dmesg   grep -i numa lscpu   grep -i numa  ___ 3.- Traza ls /var/log</pre>
	<b>Recursos I/O: Acceso Almacenamiento</b> <pre>___ 1.- Localizar propiedades del demonio dnf provides libvirt dnf repoquery --list libvirt-daemon-8.0.0 systemctl status libvirt systemctl cat libvirt.service systemctl show libvirt.service  ___ 2.- Listar sus carpeta de datos ls /var/lib/libvirt # Datos Persistentes ls /var/log/libvirt # Datos Persistentes ls /run/libvirt/    # Datos Volátiles ls /usr/lib64/libvirt # Datos Inmutables PID=\$(ps -aux   grep libvirt   grep -v grep   awk '{print \$2}') ls /proc/\$PID # Datos RAM</pre>
	<b>Recursos I/O: Estado Comunicaciones</b> <pre>ss -tupln ss -xpln</pre>

### 2.2.3. SystemD: Disparador de Demonios

Tabla 2.2-3: Ejercicios sobre el Entorno de Ejecución.

ANFITRIÓN:	
SystemD – Control Entorno Ejecución	
 Units	<b>Estructura Unit</b>  ____ 1.- Proceso PID=1 <b>pstree</b>  ____ 2.- Base Datos Unidades Ejecución <b>tree</b> /usr/lib/systemd/user <b>tree</b> /usr/lib/systemd/system  ____ 3.- Unidades Ejecución durante Arranque <b>tree</b> /etc/systemd/user <b>tree</b> /etc/systemd/system  ____ 4.- Ver Contenido Unit <b>systemctl</b> cat chronyd.service <b>systemctl</b> status chronyd.service <b>systemctl</b> show chronyd.service  ____ 5.- Editar/Crear Unit <b>vi</b> /usr/lib/systemd/system/echo@.service <b>systemctl</b> edit --full --force echo@.service <b>vi</b> /usr/lib/systemd/system/foo.service <b>systemctl</b> edit --force foo.service <b>man</b> systemd.service  ____ 6.- Recarga Unidades de Ejecución y Modelo Dependencias <b>systemctl</b> daemon-reload
	<b>Tipo Service</b>  ____ 1.- Crear Unit <b>vi</b> /usr/lib/systemd/system/echo@.service <b>systemctl</b> edit --force echo@.service <b>vi</b> /usr/lib/systemd/system/foo.service <b>systemctl</b> edit --force foo.service <b>man</b> systemd.service  ____ 2.- Sintaxis Unit Echo (genera Unit transitoria por cada evento socket) [Unit] Description=Echo Service Requires=echo.socket  [Service] Type=simple ExecStart=/usr/bin/python3 /root/systemd/echo.py %i StandardInput=socket StandardError=journal TimeoutStopSec=5 #RuntimeMaxSec=10  [Install] WantedBy=multi-user.target

<pre>_____ 3.- Sintaxis Unit Foo (primer evento socket lanza un único demonio) [Unit] Description=Foo Service After=network.target foo.socket Requires=foo.socket  [Service] Type=simple ExecStart=/usr/bin/python3 %h/root/systemd/foo.py TimeoutStopSec=5  [Install] WantedBy=default.target</pre>
<b>Tipo Object</b>
<pre>_____ 1.- Crear Unit vi /usr/lib/systemd/system/echo.socket systemctl edit --force echo.socket vi /usr/lib/systemd/system/foo.socket systemctl edit --force foo.socket man systemd.socket  _____ 2.- Sintaxis Unit echo (Lanza procesos tras cada petición) [Unit] Description=Echo Socket  [Socket] ListenStream=127.0.0.1:9999 Accept=yes  [Install] WantedBy=sockets.target  _____ 3.- Sintaxis Unit Foo (Lanza un solo demonio) [Unit] Description=Foo Socket PartOf=foo.service  [Socket] ListenStream=127.0.0.1:9000  [Install] WantedBy=sockets.target</pre>
<b>Tipo Target</b>
<pre>_____ 1.- Editar Unit ls -la /etc/systemd/system/multi-user.target.wants vi /usr/lib/systemd/system/multi-user.target systemctl edit multi-user.target man systemd.target  _____ 2.- Sintaxis Unit cat /usr/lib/systemd/system/multi-user.target [Unit] Description=Multi-User System Documentation=man:systemd.special(7) Requires=basic.target Conflicts=rescue.service rescue.target After=basic.target rescue.service rescue.target AllowIsolate=yes</pre>



## Configurar

```
cat /etc/systemd/system.conf
```

## Ciclo de Vida

```
# 1.- Arrancar / Parar
systemctl stop foo.socket
systemctl status foo.socket
systemctl start foo.socket
systemctl status foo.socket

systemctl start echo.socket
systemctl status echo.socket
systemctl stop echo.socket
systemctl status echo.socket

# 2.- Lanzar Evento
systemctl status foo.service
echo "Hello Foo" | nc localhost 9000
systemctl status foo.service
echo "Hello Echo" | nc localhost 9999
systemctl status echo@
# 3.- Activar/Desactivar
systemctl enable chronyd
systemctl disable chronyd

# 4.- Enmascarar/Liberar
systemctl stop chronyd
systemctl mask chronyd
systemctl unmask chronyd
systemctl start chronyd

# 5.- Status
systemctl --state=help
systemctl list-units
systemctl list-unit-files
systemctl status echo.socket
systemctl is-active chronyd
systemctl show chronyd
```

## Nivel de Ejecucion

```
___ 1.- Cambio de nivel y comprobacion
systemctl isolate multi-user.target
systemctl status graphical.target
systemctl list-dependencies graphical.target
who -r
systemctl -t target | grep graphical.target
___ 2.- Dependencias
systemctl list-dependencies

___ 3.- Default Target
systemctl get-default
systemctl set-default graphical.target
```

## Análisis del Arranque

```
systemd-analyze
systemd-analyze blame
systemd-analyze critical-chain
systemd-analyze plot > image.svg
```



## Configurar

### 1.- Configuraciones Efimeras

// Nivel trazado Efimera

```
systemd-analyze get-log-level
```

```
systemd-analyze set-log-level 7
```

// Limpieza de Logs Efimera

```
journalctl --disk-usage
```

```
journalctl --rotate
```

```
journalctl --flush
```

```
journalctl --vacuum-size=100M
```

```
journalctl --vacuum-time=2weeks
```

// Decodificación

```
ls /run/log/journal/
```

```
strings /run/log/journal/accbe4ce8e0543e59dfdaa635b1a4111/system.journal
```

### 2.- Configuración Persistente

```
cat /etc/systemd/journald.conf
```

```
man journald.conf
```

- **SystemMaxUse=:** Especifica el espacio máximo en disco que puede utilizar el diario en el almacenamiento persistente.
- **SystemKeepFree=:** Especifica la cantidad de espacio que el diario debe dejar libre al agregar entradas de diario al almacenamiento persistente.
- **SystemMaxFileSize=:** Controla el tamaño que pueden alcanzar los archivos de diario individuales en el almacenamiento persistente antes de rotarlos.
- **RuntimeMaxUse=:** Especifica el espacio máximo en disco que se puede utilizar en almacenamiento volátil (dentro del sistema de archivos /run).
- **RuntimeKeepFree=:** Especifica la cantidad de espacio que se reservará para otros usos al escribir datos en un almacenamiento volátil (dentro del sistema de archivos /run).
- **RuntimeMaxFileSize=:** Especifica la cantidad de espacio que un archivo de diario individual puede ocupar en almacenamiento volátil (dentro del sistema de archivos /run) antes de rotarse.

## Filtrar

### 1.- Sin Filtro

```
journalctl
```

```
journalctl /usr/bin/bash
```

```
journalctl --catalog
```

### 2.- Intervalo

```
journalctl --since "2015-01-10 17:15:00"
```

```
journalctl --since "2015-01-10" --until "2015-01-11 03:00"
```

```
journalctl --since yesterday
```

```
journalctl --since 09:00 --until "1 hour ago"
```

### 3.- Unit

```
journalctl -u nginx.service --since today
```

```
journalctl -u nginx.service -u php-fpm.service --since today
```

### 4.- Tiempo Real

```
journalctl -f
```

### 5.- Boot

```
journalctl -b
```

	<pre> 6.- Kernel journalctl -k journalctl -k -b  7.- Prioridad journalctl -p err -b • 0: emerg • 1: alert • 2: crit • 3: err • 4: warning • 5: notice • 6: info • 7: debug  8.- Filtro: Usuario, grupo, proceso man systemd.journal-fields journalctl _PID=8088 journalctl _UID=\$(id -u pangea) --since today journalctl _GID=\$(id -g pangea) --since today journalctl -F _GID // Grupos para los que hay entrada en journal journalctl _SYSTEMD_UNIT=chronyd.service -n +1   grep ch </pre>
 <p>Rsyslog</p>	<b>Formato Salida</b>
	<pre> 1.- Truncado o expansion journalctl --no-full journalctl -a journalctl -x journalctl --no-pager  2.- Formato Salida journalctl -b -u chronyd -o json journalctl -b -u chronyd -o json-pretty journalctl -b -u chronyd -o cat journalctl -b -u chronyd -o cat ERRNO=0 journalctl -b -u chronyd -o export &gt; traza.txt journalctl -b -u chronyd -o short journalctl -b -u chronyd -o short-iso journalctl -b -u chronyd -o short-monotonic journalctl -b -u chronyd -o short-precise journalctl -b -u chronyd -o verbose  3.- Recientes journalctl -n journalctl -n 20 </pre>
	<b>Configurar Dispatcher</b> <pre> 1.- Configuraciones Redireccionamiento Trazas (local/remoto) man rsyslog.conf vi /etc/rsyslog.conf </pre> <b>Logrotate - Configurar Rotación Logs y espacio en disco</b> <pre> 1.- Configuraciones del Demonio (no aplica) man logrotate.conf vi /etc/logrotate.conf  2.- Reglas de Rotación de cada log ll /etc/logrotate.d/ </pre>



	<div><b>Trazas</b></div> <div><p>___ 1.- Buffer en anillo del kernel <b>man</b> dmesg</p><p>___ 2.- Trazas generales del sistema <b>cat</b> /var/log/messages</p><p>___ 3.- Trazas del plataformado del anfitrión <b>cat</b> /var/log/anaconda/syslog</p><p>___ 4.- Trazas de accesos al sistema <b>cat</b> /var/log/secure</p><p>___ 5.- Trazas del arranque al sistema <b>cat</b> /var/log/boot*.log</p><p>___ 6.- Registro en binario del Journal <b>ls</b> /run/log/journal <b>strings</b> /run/log/journal/accbe4ce8e0543e59dfdaa635b1a4111/system.journal</p><p>___ 7.- Trazas de paquetes instalados en el sistema <b>cat</b> /var/log/yum.log</p><p>___ 8.- Trazas personalizadas <b>cat</b> /var/log/audit/audit.log</p><p>___ 9.- Trazas de demonios específico (libvirt como ejemplo) <b>ls</b> /var/log/libvirt/qemu/ <b>cat</b> /var/log/libvirt/qemu/*log</p></div>
<div> audit</div>	<div><b>Configurar</b></div> <div><b>dnf</b> install audit <b>cat</b> /etc/audit/auditd.conf <b>ls</b> /etc/audit/plugins.d/</div> <div><b>Aplicar Reglas</b></div> <div><p>___ 1.- Efimeras</p><p><b>auditctl</b> -w /etc/passwd -p wa -k passwd_changes <b>auditctl</b></p><p>-b 8192 Máximo tamaño para el buffer del kernel. -f 1 Como informar al nucleo cuando un problema con auditd ha sido detectado. -e 2 Habilita (0) o deshabilita (1) auditd, con (2) se bloquea. -r 0 Límite de logs registrados por segundo 0 establece sin límite. -s Estado de systemd. -l Lista las reglas actuales en uso. -D Borra todas las reglas en uso. -W Borrar una regla relativa al sistema de ficheros. Debe especificarse siempre todos los parámetros, ej. "<b>auditctl</b> -W /usr/bin -p x -k sbin_monitor" -d Borra las reglas definidas mediante llamadas al sistema: <b>auditctl</b> -d always,exit -F arch=b64 -S unlink -S unlinkat -S rename -S renameat -F uid=0 -k sbin_monitor -R /usr/share/doc/audit-version/stig.rules Permite definir una ruta a un fichero con reglas. -p</p><p>wa: Indica que se auditarán las acciones de "escritura" ("write" en inglés) y "atributo" ("attribute" en inglés). x: Solo audita los eventos de "salida" ("exit" en inglés), que marcan el final de una syscall (llamada al sistema). a: Todas las acciones (lectura, escritura, ejecución y salida).</p></div>

## 2.- Persistentes

```
cat /usr/share/audit/sample-rules/README-rules
```

This group of rules are meant to be used with the augenrules program. The augenrules program expects rules to be located in /etc/audit/rules.d/ The rules will get processed in a specific order based on their natural sort order. To make things easier to use, the files in this directory are organized into groups with the following meanings:

```
10 - Kernel and auditctl configuration
20 - Rules that could match general rules but we want a different match
30 - Main rules
40 - Optional rules
50 - Server Specific rules
70 - System local rules
90 - Finalize (immutable)
```

There is one set of rules, 31-privileged.rules, that should be regenerated. There is a script in the comments of that file. You can uncomment the commands and run the script and then rename the resulting file.

The rules are not meant to be used all at once. They are pieces of a policy that should be thought out and individual files copied to /etc/audit/rules.d/ For example, if you wanted to set a system up in the STIG configuration, copy rules 10-base-config, 30-stig, 31-privileged, and 99-finalize. You can add more if you like. Also, not all arches have the same syscalls. It is expected that the rules be fine tuned for the arch they are deployed on. For example, aarch64 does not have the open syscall. It should just be deleted from the rules.

Once you have the rules in the rules.d directory, you can load them by running  
augenrules --load

```
ls /usr/share/audit/sample-rules
cp /usr/share/audit/sample-rules/30-stig.rules /etc/audit/audit.rules/.
augenrules --load
cat /etc/audit/audit.rules
```

### Filtrar Resultados

#### 1.- Búsqueda

```
ausearch -i --raw -a <EVENT-ID> --file <FILENAME> -k <KEY>
--start <START-TIME> --end <END-TIME>

ausearch -k key # Clave (nombre de la regla / grupo de reglas).
ausearch -p 12345 # Identificador de proceso.
ausearch -c touch # Eventos que tengan que ver con el comando touch (comm="touch").
ausearch -ua 100 # Id de usuario, tanto el efectivo como el login user id (auid).
ausearch -gi 100 # GID.
ausearch -x XXXX # Path del binario ejecutado (exe="/usr/bin/touch").
ausearch -sc XXX # Llamada del sistema.
ausearch -sv yes # Estado de finalización, exitoso "yes", en caso contrario "no".
ausearch -f file # Buscar por fichero / directorio.
ausearch -w word comando # Coincidencias basadas en cadenas de la palabra completa.
# Esta categoría de coincidencias incluye: nombre de archivo,
# nombre de host, terminal y contexto de SELinux.
# Ejemplo: ausearch -w -f /home/vagrant/

ausearch -k passwd_change
```

#### 2.- Informes

```
aureport --summary # Estadísticas generales (eventos, accesos, procesos, etc).
# Se puede combinar con otras opciones para realizar sumatorios.
# Ejemplo: aureport -x --summary
aureport --success # Estadísticas de eventos con resultado exitoso.
aureport --failed # Estadísticas de eventos con resultado fallido.
```

```
# Obtención de reportes (se puede combinar con --summary).
aureport -c # Cambios en la configuración de auditd.
aureport -l # Logins en el sistema.
aureport -p # Procesos: Fecha, tiempo, id,nombre, syscalls, auid y número de evento.
aureport -f # Ficheros: Fecha, tiempo, id,nombre, syscalls, auid y número de evento.
aureport -u # Usuarios: Fecha, tiempo, id,nombre, syscalls, auid y número de evento.
aureport -s # Syscalls: Fecha, tiempo, número de llamada,
# nombre del comando que uso la syscall, auid y número de evento.
aureport -ts yesterday
aureport --summary
```

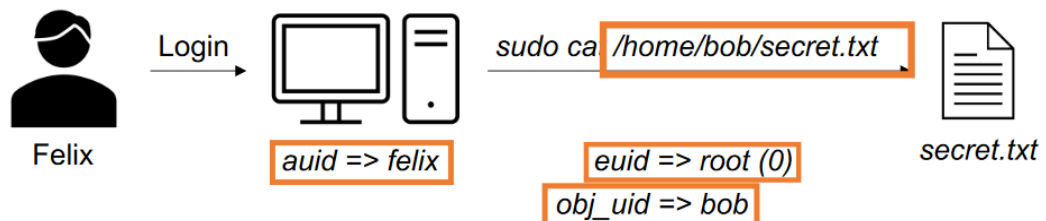
### 3.- Estadísticas Accesos

```
aulast --extract
cat aulast.log | tail -20
```

#### Sintaxis 1: Watch - Auditoría Ficheros

```
man audit.rules # manpages
auditctl -w /etc/passwd -p wa -k <KEY> # watch: write, attribute changes with key
auditctl -w /etc/sysconfig -p rwa -k <KEY> # recursive watch: all files and dirs
auditctl -w /bin -p x # all executions in bin
auditctl -W <PATH> # remove watch rule(s) at path
auditctl -d <RULE> # remove previous -a or -A rule(s)
auditctl -D # remove all rules (or they get removed by reboot)
vim /etc/audit/rules.d/audit.rules # persistent rules, without auditctl at the beginning
-w /etc -p w -k etc_content # w=path, p=permission((r)ead,(w)rite,e(x)ecute,(a)tttribute)
-w /etc -p a -k etc_attribute
```

#### Sintaxis 2: Syscall - Auditoría de Llamadas al Sistema



#### # PROCESOS

```
# 1. execve: Todas las nuevas aplicaciones que son ejecutadas
-a exit,always -S execve -k execve
```

```
# 2. ptrace/process_vm_readv/process_vm_writev: Introspeccion de Procesos
```

```
-a exit,always -S ptrace -F a0=16 -k ptrace_attach
-a exit,always -S ptrace -F a0=16902 -k ptrace_seize
-a exit,always -S process_vm_readv -k process_vm
-a exit,always -S process_vm_writev -k process_vm
```

#### # I/O: Ficheros

```
# 1. connect/listen: network io
```

```
-a exit,always -S connect -k netconns_out
-a exit,always -S listen -k netconns_in
```

```
# 2. Fallos en apertura de Ficheros
```


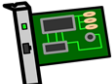

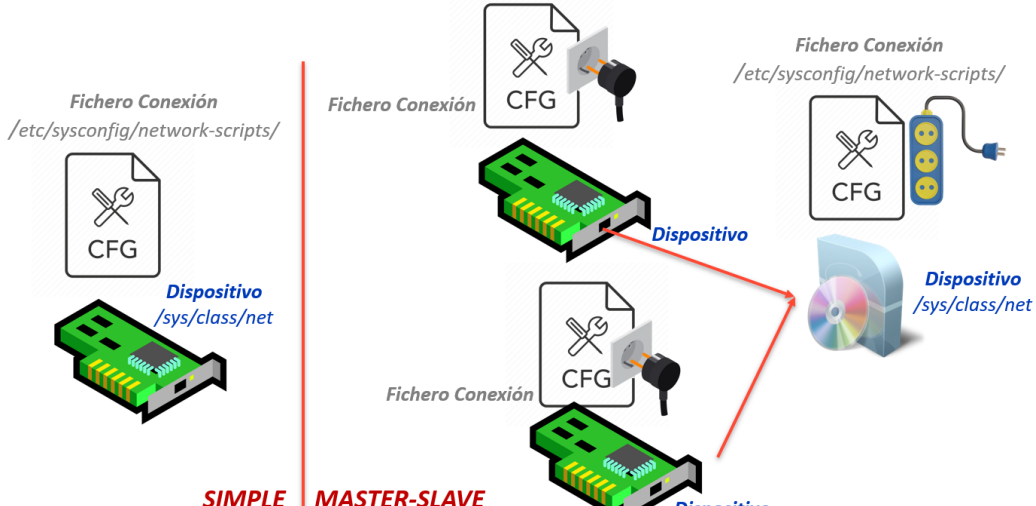
```
-a always,exit -S open -S openat -F exit=-EPERM -k open_fail
-a always,exit -S open -S openat -F exit=-EACCES -k open_fail
-a always,exit -F arch=b64 -S open -F success=0
```

```
# 3. Reglas Watch, pero a través de Auditoría de llamadas al Sistema
```

```
-a exit,always -F dir=/home -F euid=0 -C auid!=obj_uid -k sudoAbuse
```

## 2.2.4. NetworkManager: Gestión de Red

Tabla 2.2-4: Ejercicios sobre el NetworkManager.

	<p style="text-align: center;"><b>ANFITRIÓN:</b></p> <h1 style="text-align: center;">Network Manager - Comunicaciones</h1>
	<p><b>Tráfico de Red</b></p> <pre> ip -s link tcpdump -i ens5f2 -nn -s0 -v tcpdump -i br-aux -nn -s0 -v port 22 tcpdump -c &lt;NUMBER&gt; -w &lt;FILE.pcap&gt; # capture number of packets to the file tcpdump -r &lt;FILE.pcap&gt; # read from a capture file tcpdump 'host &lt;HOSTNAME&gt;' # coming to/from host tcpdump 'src &lt;HOSTNAME&gt;' # from host tcpdump 'port &lt;NUMBER&gt;'  # icmp to/from host tcpdump 'ip host &lt;HOSTNAME1&gt; and not &lt;HOSTNAME2&gt;' tcpdump 'icmp and host &lt;IPv4&gt;' tcpdump -x # display packet header and hexadecimal values tcpdump -X # display data as hexadecimal and ASCII values tcpdump -X -r &lt;FILE.pcap&gt; 'host &lt;HOSTNAME&gt;'   grep -i 'pass' # display plaintext passwords </pre> <p><b>Reglas Persistentes</b></p> <p>___ 1.- Secuencia carga de reglas UDEV</p> <pre> /usr/lib/udev/rules.d/60-net.rules (sysconfig) /usr/lib/udev/rules.d/71-biosdevname.rules (Dell) /usr/lib/udev/rules.d/75-net-description.rules (80-net-setup-link.rules) /usr/lib/udev/rules.d/80-net-setup-link.rules (NamePolicy, 99-default.link) </pre> <p>___ 2.- Descripción de una Regla</p> <pre> # cat /etc/udev/rules.d/10-local.rules  SUBSYSTEM=="usb", ACTION=="add remove", ENV{DEVTYPE}=="usb_device", RUN=="/kvm/bin/kvm_usb_reattach \$env{PRODUCT} \$env{ACTION} \$env{BUSNUM} \$env{DEVNUM}" </pre> <p>-<b>SUBSYSTEM</b>: el evento debe provenir de un concentrador USB para que se aplique esta regla.  -<b>ACTION</b>: el evento es enchufar o desenchufar el dispositivo para que se aplique esta regla.  -<b>ENV{DEVICE_TYPE}</b>: el evento ha de provenir de un dispositivo USB para que se aplique la regla.  -<b>RUN</b>: script con sus argumentos, que son las variables del evento UDEV necesarias para crear la definición de XML</p>
	<p><b>Simple: Ethernet</b></p>  <p>The diagram illustrates the configuration of network interfaces. It shows a 'Fichero Conexión' (CFG) file in the directory /etc/sysconfig/network-scripts/. Below this, it shows a 'Dispositivo' (Device) in the directory /sys/class/net. A red line separates the 'SIMPLE' configuration from the 'MASTER-SLAVE' configuration. The 'MASTER-SLAVE' configuration shows a 'Dispositivo' in /sys/class/net connected to a 'Dispositivo' in /sys/class/net, with a 'Fichero Conexión' (CFG) file in /etc/sysconfig/network-scripts/.</p>

```
___ 1.- Perfil Unico
nmcli dev connect eno8403np1
nmcli con add con-name TEST type ethernet ifname eno8403np1
nmcli con show
nmcli connection modify "TEST" connection.id "Internal-LAN"
nmcli connection modify "Internal-LAN" ipv4.method manual
                                   ipv4.addresses 10.10.10.10/24
nmcli connection up Internal-LAN

___ 2.- Multiples Perfiles (method shared: casa, trabajo)
___ 2.1- Perfil CASA
nmcli dev connect eno8403np1
nmcli con add con-name TEST type ethernet ifname eno8403np1
nmcli con show
nmcli connection modify "TEST" connection.id "CASA"
nmcli connection modify "Internal-LAN" ipv4.method share ipv4.addresses
10.10.10.10/24

___ 2.2- Perfil TRABAJO
nmcli con add con-name TRABAJO type ethernet ifname eno8403np1
                                   ipv4.address 5.5.5.5/24 ipv4.method shared

___ 2.3- Conmutar Perfiles
nmcli connection up CASA
nmcli con show
nmcli con down CASA
nmcli con show
nmcli con up TRABAJO
nmcli con show

___ 2.4- Cambiar configuración desde Scripts
FILE=$(ls -l /etc/sysconfig/network-scripts/ | grep CASA)
vi $FILE
nmcli con up CASA
```

#### Simple: VLAN

```
___ 1.- VLAN por DHCP
nmcli connection add con-name VLAN10 type vlan ifname vlan10
                                   vlan.id 10
                                   vlan.parent enp5f2

___ 2.- VLAN con IP estática
nmcli con add con-name VLAN10 type vlan ifname vlan10
                                   vlan.id 10
                                   vlan.parent enp5f2
                                   ipv4.method manual
                                   ipv4.addresses 10.10.10.10/24
```

#### Master-Slave: Bond

```
___ 1.- Crear Agregador Virtual (Master)
___ 1.1- Conexión
nmcli connection add type bond con-name "UCS-Bond" ifname "Regleta0"
                                   bond.options "mode=active-backup,miimon=80,primary=enp5s0,updelay=16000"
nmcli con show
ls /etc/sysconfig/network-scripts/
FILE=$(ls -l /etc/sysconfig/network-scripts/ | grep Bond)
cat /etc/sysconfig/network-scripts/$FILE

___ 1.2- Dispositivo Virtual
ls /sys/class/net/Regleta0
nmcli connection modify 'UCS-Bond' ipv4.addresses '10.10.61.131/24'
                                   ipv4.gateway ' 10.10.61.254' ipv4.method manual
cat /etc/sysconfig/network-scripts/$FILE
```

```

2.- Enchufar dispositivos físicos (interfaz ethernet)
nmcli connection add con-name UCS-Slave2 type ethernet slave-type bond
                           ifname ens4f2 master Regleta0
nmcli connection add con-name UCS-Slave1 type ethernet slave-type bond
                           ifname ens5f2 master Regleta0

```

3.- Arrancar todas las conexiones

```

nmcli connection up UCS-Slave2
nmcli connection up UCS-Slave1
nmcli connection up UCS-Bond

```

4.- Comprobaciones

```

cat /proc/net/bonding/Regleta0

```

5.- Cambiar Configuración del Agregador Virtual desde Scripts

```

FILE=$(ls -l /etc/sysconfig/network-scripts/ | grep Bond)

```

```

vi $FILE

```

```

nmcli connection modify 'UCS-Bond' ipv4.method manual

```

```

nmcli con up UCS-Bond

```

#### Master-Slave: Bridge

1.- Crear Agregador Virtual (Bridge)

1.1- Conexión

```

nmcli con del Interna
nmcli con del br-int-fisico
nmcli con add con-name Interna ifname br-int type bridge
                           autoconnect yes ethernet.auto-negotiate yes
                           ipv6.method disabled bridge.stp no ipv4.method disable
nmcli con show

```

```

ls /etc/sysconfig/network-scripts/

```

```

FILE=$(ls -l /etc/sysconfig/network-scripts/ | grep -i interna)

```

```

cat /etc/sysconfig/network-scripts/$FILE

```

1.2- Dispositivo Virtual

```

ls /sys/class/net/br-int

```

2.- Enchufar dispositivos Físicos (interfaz ethernet)

```

nmcli con add con-name br-int-fisico ifname ens5f3
                           type bridge-slave master br-int

```

3.- Arrancar todas las Conexiones

```

nmcli con up br-int-fisico

```

```

nmcli con up Interna

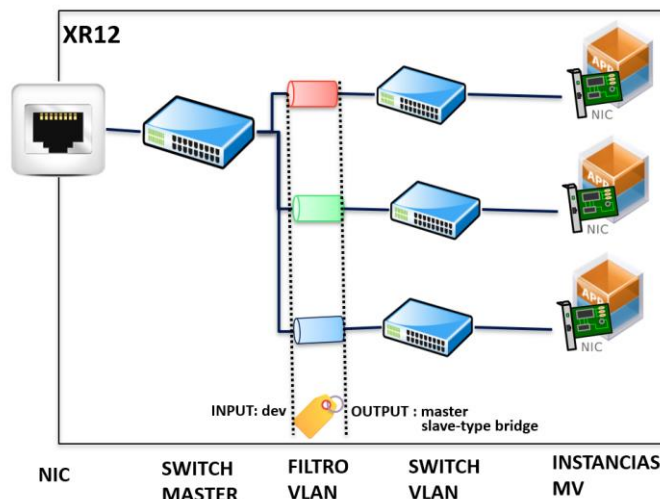
```

```

nmcli con show

```

#### Master-Slave: VLAN Bridge



```
#!/bin/bash
BRIDGE_OPTS="autoconnect yes ethernet.auto-negotiate yes ipv4.method disabled ipv6.method
disabled bridge.stp no"
BRIDGE_OPTS1="ipv4.never-default true ipv6.method ignore ipv6.never-default true 802-3-
ethernet.mtu 1500"

# PASO 1 - Borra las interfaces de red actuales
nmcli con delete $(nmcli con show | cut -d " " -f1 | grep -v NAME)

# PASO 2 - Crear el Switch Principal, asociado a la boca física de red
nmcli con add type bridge          con-name CADENA-AUX      ifname br-aux      ${BRIDGE_OPTS}
${BRIDGE_OPTS1}
nmcli con add type ethernet        con-name ENS4F2          ifname ens4f2      master br-aux


# PASO 3 - Crear el Switch Secundario, al que se engancha filtro VLAN
nmcli con add type bridge  con-name BRGSI
                           ifname br-aux.12      ${BRIDGE_OPTS} ${BRIDGE_OPTS1}

# PASO 4 - Crear el Filtro VLAN, entrada Switch Principal, salida Switch Sec.
nmcli con add type vlan    con-name VRGSI
                           ifname br-vaux.12    # Filtro Virtual
                           dev br-aux id 12      # Entrada Filtro
                           master br-aux.12 slave-type bridge # Salida Filtro


# PASO 5 - Asignar IP al Switch Secundario (acceso al anfitrión)
nmcli con mod BRGSI  ipv4.addresses 10.40.12.229/24 ipv4.method manual
```

### 2.2.5. Chrony: Sincronismo

Tabla 2.2-5: Ejercicios sobre Chrony.

ANFITRIÓN: Chrony – Sincronismo	
	<b>Configurar</b> ____ 1.- /etc/hosts <b>vi</b> /etc/hosts <b>systemctl</b> restart chronyd  ____ 2.- /etc/chrony.conf _____ 2.1- RED AISLADA server master driftfile /var/lib/chrony/drift logdir /var/log/chrony log measurements statistics tracking keyfile /etc/chrony.keys commandkey 24 local stratum 10 initstepslew 20 master allow 192.0.2.123  _____ 2.2- RED CONECTADA NTP EXTERNO # Use public servers from the pool.ntp.org project. # Please consider joining the pool ( <a href="http://www.pool.ntp.org/join.html">http://www.pool.ntp.org/join.html</a> ). server ntpserver1 iburst server ntpserver2 iburst  # Record the rate at which the system clock gains/losses time. driftfile /var/lib/chrony/drift # Allow the system clock to be stepped in the first three updates # if its offset is larger than 1 second. makestep 1.0 3



	<pre># Enable kernel synchronization of the real-time clock (RTC). rtcsync  # Enable hardware timestamping on all interfaces that support it. #hwtimestamp * # Increase the minimum number of selectable sources required to adjust # the system clock. #minsources 2  # Allow NTP client access from local network. #allow 192.168.0.0/16 # Serve time even if not synchronized to a time source. #local stratum 10  # Specify file containing keys for NTP authentication. keyfile /etc/chrony.keys  # Get TAI-UTC offset and leap seconds from the system tz database. leapsetz right/UTC # Specify directory for log files. logdir /var/log/chrony # Select which information is logged. log measurements statistics tracking</pre> <b>Sincronizar</b> <pre>chronyc -n sources -v chronyc -n tracking chronyc -n ntpdata</pre>
	<b>Configurar</b> <pre>___ 1.- /etc/hosts vi /etc/hosts systemctl restart chronyd ___ 2.- /etc/chrony.conf ___ 2.1- RED AISLADA (Configuración Reloj) driftfile /var/lib/chrony/drift commandkey 1 keyfile /etc/chrony.keys initstepslew 10 client1 client3 client6 local stratum 8 manual allow 192.0.2.0 ___ 2.2- RED CONECTADA NTP EXTERNO # Utiliza el servidor NTP por defecto server time.server1.com iburst server time.server2.com iburst # Restringe el acceso a la consulta del servidor NTP allow 127.0.0.1 allow ::1 # Reglas de acceso para otras redes # (Ajusta según tus necesidades) # allow 192.168.1.0/24 # Directorio donde Chrony almacena información temporal driftfile /var/lib/chrony/chrony.drift # Registro de eventos log tracking measurements statistics # Camino hacia el archivo de registro logdir /var/log/chrony</pre> <b>Sincronizar</b> <pre>chronyc -n sources -v chronyc -n tracking chronyc -n ntpdata</pre>

## 2.3. USO XR12: VIRTUALIZACIÓN.

### 2.3.1. Hipervisor: Entorno de Ejecución de Máquinas Virtuales

Tabla 2.3-1: Ejercicios sobre Hipervisor KVM.

MAQUINA VIRTUAL: HIPERVISOR	
	<b>Instalación</b>  ___ 1.- Verificar Aceleración Hardware activada en BIOS <code>grep --color -E "vmx svm" /proc/cpuinfo</code> <code>journalctl -p 3 -xb</code>  ___ 2.- Instalación del Sistema Libvirtd/QEMU <code># sudo dnf install qemu-kvm qemu-img libvirt libvirt-client -y</code>  ___ 3.- Instalación de Herramientas de Gestión <code># sudo dnf install virt-install virt-manager -y</code>  ___ 4.- Arrancar demonio Libvirtd <code># sudo systemctl enable --now libvirtd</code> <code># sudo systemctl status libvirtd</code>  ___ 5.- Validación Instalación <code>dnf list installed *kvm*</code> <code>virt-host-validate</code>
	<b>Monitorización CPU y RAM Anfitrión</b>  ___ 1.- Uso de RAM <code>gnome-system-monitor</code> <code>lscpu</code> <code>cat /proc/cpuinfo</code> <code>lshw -short -C memory</code> <code>cat /proc/meminfo</code> <code>dmidecode -t memory</code> <code>free -h</code>  ___ 2.- Árbol de Procesos, localizar subsistema KVM <code>pstree</code> <code>ps -faux</code>  ___ 3.- Análisis uso de Cores de la CPU <code>ps -ax -o pid,ppid,%cpu,%mem,psr,sgi_p,state,comm</code> <code>top</code>
	<b>Carpetas</b>  ___ 1.- Instancias Definidas <code>tree /etc/libvirt/qemu</code> ___ 2.- Instancias en Ejecución <code>tree /run/libvirt/</code>
	<b>Trazas</b> <code>ls /var/lib/libvirt/qemu</code>



## Enchufar USB

### \_\_\_ 1.- Tabla USB

```
cat /kvm/config/kvm_usb_map
```

### \_\_\_ 2.- Udev Persistent Rules

```
cat /etc/udev/rules.d/10-local.rules
```

```
SUBSYSTEM=="usb", ACTION=="add|remove", ENV{DEVTYPE}=="usb_device",  
RUN="/kvm/bin/kvm_usb_reattach $env{PRODUCT} $env{ACTION} $env{BUSNUM}  
$env{DEVNUM}"
```

### \_\_\_ 3.- Deteccion

```
udevadm monitor -property
```

```
ACTION=bind
```

```
BUSNUM=001
```

```
DEVNAME=/dev/bus/usb/001/011
```

```
DEVNUM=011
```

```
DEVPATH=/devices/pci0000:00/0000:00:14.0/usb1/1-13
```

```
DEVTYPE=usb_device
```

```
DRIVER=usb
```

```
ID_BUS=usb
```

```
ID_MODEL=0005
```

```
ID_MODEL_ENC=0005
```

```
ID_MODEL_FROM_DATABASE=Type 6 Keyboard
```

```
ID_MODEL_ID=0005 # COLUMNA 3: Disp.
```

```
ID_REVISION=0200
```

```
ID_SERIAL=0430_0005
```

```
ID_USB_INTERFACES=:030101:
```

```
ID_VENDOR=0430
```

```
ID_VENDOR_ENC=0430
```

```
ID_VENDOR_FROM_DATABASE=Sun Microsystems, Inc. # COLUMNA 5: Descripcion (opcional)
```

```
ID_VENDOR_ID=0430 # COLUMNA 2: Fab.
```

```
MAJOR=189
```

```
MINOR=10
```

```
PRODUCT=430/5/200 # COLUMNA 1 : ProductID
```

```
SEQNUM=8065
```

```
SUBSYSTEM=usb
```

```
TYPE=0/0/0
```

```
USEC_INITIALIZED=1201201924604
```



## Creación

### \_\_\_ 1.- Listado Pooles

```
virsh domstate
```

```
virsh pool-list
```

```
virsh pool-dumpxml <pool>
```

### \_\_\_ 2.- Creacion de Pooles comando

```
virsh pool-define-as <isos> --type dir --target </var/lib/libvirt/isos>
```

### \_\_\_ 3.- Creacion de Pooles XML

```
sudo cp /etc/libvirt/storage/default.xml $HOME/<isos.xml>
```

```
sudo vi $HOME/<isos.xml>
```

```
<pool type="dir">
```

```
  <name>isos</name>
```

```
  <target>
```

```
    <path>/var/lib/libvirt/isos</path>
```

```
  </target>
```

```
</pool>
```


```
sudo virsh pool-define <isos.xml>
```

	<p>____ 4.- Arranque Pool <b>virsh</b> pool-start &lt;isos&gt; <b>virsh</b> pool-autostart &lt;isos&gt;</p> <p><b>Monitorización Almacenamiento</b></p> <p>____ 1.- Ocupación del Disco <b>df</b> -f <b>lsblk</b> <b>blkid</b> <b>du</b> -h &lt;path&gt;</p> <p>____ 2.- Estado pooles <b>virsh</b> vol-list default --details</p>
	<p><b>Regeneración Overlay</b></p> <pre>#!/bin/bash  # 1.- Generar el Fichero de Configuracion echo "1.- Generar fichero configuracion" echo '# Una vez modificada, se debera ejecutar systemctl restart kvm-net-reconfig IP:172.30.239.41/22 GW:172.30.239.254' &gt; /kvm/config/ip_gestion  cat /kvm/config/ip_gestion  # 2.- Eliminar el lock echo '' echo "rm -f /kvm/ip-actual" rm -f /kvm/ip-actual  # 3.- Reiniciar servicio echo "systemctl restart kvm-net-reconfig.service" systemctl restart kvm-net-reconfig.service  # 4.- Activar Conexiones echo "nmcli con up Auxiliar" nmcli con up Auxiliar  echo "nmcli con up Interna" nmcli con up Interna</pre>

## 2.3.2. Máquina Virtual: Gestión de Instancias

### 2.3.2.1. Gestión Instancias Máquina Virtual

Tabla 2.3-2: Procedimientos sobre Instancias de Máquinas Virtuales.

	MAQUINA VIRTUAL: GESTIÓN DE INSTANCIAS
	<b>Transiciones</b>  _____ 1.- Definir Instancia _____ 1.1.- Copiar XML // Desde capeta local <b>cp</b> XR12_CWP.rhel8.4.xml /kvm/vhd/. <b>cp</b> XR12_OUCS.rhel7.7.xml /kvm/vhd/. <b>cp</b> XR12_POS.rhel6.8.xml /kvm/vhd/.  // Desde capeta remota a la local <b>scp</b> root@<IP gestion>:/kvm/vhd/XR12_CWP.rhel8.4.xml /kvm/vhd/. <b>scp</b> root@<IP gestion>:/kvm/vhd/XR12_OUCS.rhel7.7.xml /kvm/vhd/. <b>scp</b> root@<IP gestion>:/kvm/vhd/XR12_POS.rhel6.8.xml /kvm/vhd/.  _____ 1.2.- Copiar QCOW desde capeta local // Desde capeta local <b>cp</b> CWP.rhel8.4.home.qcow2 /kvm/vhd/. <b>cp</b> CWP.rhel8.4.system.qcow2 /kvm/vhd/. <b>cp</b> OUCS.rhel7.7.qcow2 /kvm/vhd/. <b>cp</b> POS.rhel6.8.qcow2 /kvm/vhd/. // Desde capeta remota a la local <b>scp</b> root@<IP gestion>:/kvm/vhd/CWP.rhel8.4.home.qcow2 /kvm/vhd/. <b>scp</b> root@<IP gestion>:/kvm/vhd/CWP.rhel8.4.system.qcow2 /kvm/vhd/. <b>scp</b> root@<IP gestion>:/kvm/vhd/OUCS.rhel7.7.qcow2 /kvm/vhd/. <b>scp</b> root@<IP gestion>:/kvm/vhd/POS.rhel6.8.qcow2 /kvm/vhd/.  _____ 1.3.- Copiar NVRAM desde capeta local // Desde capeta local <b>cp</b> CWP.rhel8.4_VARS.fd /kvm/uefi <b>cp</b> OUCS.rhel7.7_VARS.fd /kvm/uefi <b>cp</b> POS.rhel6.8_VARS.fd /kvm/uefi // Desde capeta remota a la local <b>scp</b> root@<IP gestion>:/kvm/uefi/CWP.rhel8.4_VARS.fd /kvm/uefi/. <b>scp</b> root@<IP gestion>:/kvm/uefi/OUCS.rhel7.7_VARS.fd /kvm/uefi/. <b>scp</b> root@<IP gestion>:/kvm/uefi/POS.rhel6.8_VARS.fd /kvm/uefi/.  _____ 1.4.- Permisos y comprobaciones <b>chown</b> qemu:qemu /kvm/vhd/*xml <b>chown</b> qemu:qemu /kvm/vhd/*qcow2 <b>chown</b> qemu:qemu /kvm/uefi/*fd <b>chmod</b> 600 /kvm/vhd/*xml <b>chmod</b> 644 /kvm/vhd/*qcow2 <b>chmod</b> 644 /kvm/uefi/*fd <b>ll</b> -h /var/lib/libvirt/images/ <b>ll</b> /var/lib/libvirt/qemu/nvram/

```
_____ 1.5.- Definicion de instancias y comprobaciones
virsh define /kvm/vhd/XR12_CWP.rhel8.4.xml
virsh define /kvm/vhd/XR12_OUCS.rhel7.7.xml
virsh define /kvm/vhd/XR12_POS.rhel6.8.xml
tree /etc/libvirt/qemu
diff /kvm/vhd/XR12_CWP.rhel8.4.xml /etc/libvirt/qemu/CWP.rhel8.4.xml
```

```
_____ 1.6.- Configurar arranque instancias durante Arranque anfitrión
virsh autostart CWP.rhel8.4
virsh autostart OUCS.rhel7.7
virsh autostart POS.rhel6.8
```

```
_____ 2.- Arranca Instancia
virsh list --all
virsh start CWP.rhel8.4
virsh start OUCS.rhel7.7
virsh start POS.rhel6.8
```

```
_____ 3.- Parar Instancia
virsh list
virsh shutdown CWP.rhel8.4
virsh shutdown OUCS.rhel7.7
virsh shutdown POS.rhel6.8
virsh destroy CWP.rhel8.4
virsh destroy OUCS.rhel7.7
virsh destroy POS.rhel6.8
```

```
_____ 4.- Reiniciar Instancia
virsh list
virsh reboot CWP.rhel8.4
virsh reboot OUCS.rhel7.7
virsh reboot POS.rhel6.8
```

```
_____ 5.- Borrar Instancia
virsh list --inactive
virsh undefine CWP.rhel8.4
virsh undefine OUCS.rhel7.7
virsh undefine POS.rhel6.8
```

## Interacciones en Frío

```
_____ 1.- Transferencia ficheros
virsh list --inactive
// 1.1.- Anfitrión -> Instancia
// POS
virt-copy-in -d POS.rhel6.8 /root/TAR_SACTA.tgz /tmp/.
virt-copy-in -a /kvm/vhd/POS.rhel6.8.qcow2 /root/TAR_SACTA.tgz /tmp/.
// OUCS
virt-copy-in -d OUCS.rhel7.7 /root/TAR_SACTA.tgz /tmp/.
virt-copy-in -a /kvm/vhd/OUCS.rhel7.7.qcow2 /root/TAR_SACTA.tgz /tmp/.
// CWP
virt-copy-in -d CWP.rhel8.4 H-CWP_CWPA_CWPA_03.02.63_01.00.tar.gz /var/tmp/.
virt-copy-in -d CWP.rhel8.4 H-CWP_CWPB_CWPB_03.02.63_01.01.tgz /var/tmp/.
virt-copy-in -a /kvm/vhd/CWP.rhel8.4.system.qcow2
                  /root/H-CWP_CWPA_CWPA_03.02.63_01.00.tar.gz /tmp/.
virt-copy-in -a /kvm/vhd/CWP.rhel8.4.system.qcow2
                  /root/H-CWP_CWPB_CWPB_03.02.63_01.01.tgz /tmp/.
```

```
// Instancia -> Anfitrión
// POS
virt-copy-out -d POS.rhel6.8 /tmp/TAR_SACTA.tgz /root/.
virt-copy-out -a /kvm/vhd/POS.rhel6.8.qcow2 /tmp/TAR_SACTA.tgz /tmp/.
// OUCS
virt-copy-out -d OUCS.rhel7.7 /tmp/TAR_SACTA.tgz /root/.
virt-copy-out -a /kvm/vhd/OUCS.rhel7.7.qcow2 /tmp/TAR_SACTA.tgz /tmp/.
// CWP
virt-copy-out -d CWP.rhel8.4
                /var/tmp/H-CWP_CWPA_CWPA_03.02.63_01.00.tar.gz /root/.
virt-copy-out -a /kvm/vhd/OUCS.rhel7.7.qcow2
                /var/tmp/H-CWP_CWPA_CWPA_03.02.63_01.00.tar.gz /root/.
virt-copy-out -d CWP.rhel8.4
                /var/tmp/H-CWP_CWPB_CWPB_03.02.63_01.01.tgz /root/.
virt-copy-out -a /kvm/vhd/OUCS.rhel7.7.qcow2
                /var/tmp/H-CWP_CWPB_CWPB_03.02.63_01.01.tgz /root/.
```

## Interacciones en Caliente

```
_____ 1.- Acceso a la consola
virsh list
virsh start CWP.rhel8.4 --console
virsh console CWP.rhel8.4
// ;;; Cuidado con la salida de la consola !!!  exit + ( CTRL+] ó CTRL+5 )
_____ 2.- Acceso SSH
ssh root@<IP Gestion> (password: focus)
_____ 3.- Descripción de la Instancia
virsh dumpxml CWP.rhel8.4 > test.xml
```

## Modificaciones en Frío

```
_____ 1.- virt-customize
// Password
virt-customize -d CWP.rhel8.4 --password sacta:password:miclave
virt-customize -a /var/lib/libvirt/images/CWP.rhel8.4.system.qcow2 \
                --password sacta:password:miclave

// Password usuario root
virt-customize -d CWP.rhel8.4 --root-password password:miclave
virt-customize -a /var/lib/libvirt/images/CWP.rhel8.4.system.qcow2 \
                --root-password password:miclave

// Hostname
virt-customize -d CWP.rhel8.4 -hostname CWP
virt-customize -a /var/lib/libvirt/images/CWP.rhel8.4.system.qcow2 \
                -hostname CWP2

// Comando
virt-customize -d CWP.rhel8.4 --run-command 'echo "¡Hola, mundo!" >
/etc/motd'
virt-customize -a /var/lib/libvirt/images/CWP.rhel8.4.system.qcow2 \
                --run-command 'echo "¡Hola, mundo2!" > /etc/motd'

// Script
virt-copy-in -d CWP.rhel8.4 /root/kvm/test.sh /tmp/.
virt-customize -d CWP.rhel8.4 -run '/tmp/test.sh'
virt-customize -a /var/lib/libvirt/images/CWP.rhel8.4.system.qcow2 \
                -run '/tmp/test.sh'

_____ 2.- guestfish
guestfish -d CWP.rhel8.4 -i --rw
guestfish -a /var/lib/libvirt/images/CWP.rhel8.4.system.qcow2 -i --rw
```



## Modificaciones en Caliente

```
virt-manager
virsh list --all
virsh edit CWP.rhel8.4
```

## Automatización Modificaciones en Frío: Script dentro Instancia

```
1.- Parámetros Instancia
#!/bin/bash
SEPARADOR_REEMPLAZO="@@@@"

# 1.- Lista de ficheros sobre los que actuar
FICHERO=( "/etc/sysconfig/network-scripts/ifcfg-config1" \
          "/etc/sysconfig/network-scripts/ifcfg-config1" \
          "/etc/sysconfig/network-scripts/ifcfg-configN" \
          )

# 2.- Lista de reemplazos que realizar
REEMPLAZOS=( "172.18.14.245${SEPARADOR_REEMPLAZO}192.168.1.7" \
             "172.22.12.125${SEPARADOR_REEMPLAZO}192.168.1.15" \
             "172.28.16.82${SEPARADOR_REEMPLAZO}192.168.1.17" )

2.- Lógica Bash
#!/bin/bash
# 1.- Carga Valores Instancia
source "./instancia.txt"

# 2.- Configura Instancia
for ((idx=0; idx<${#FICHERO[@]}; ++idx)); do

    # 2.1.- Extra Parametros de Los Arrays
    DEST_FILE="${FICHERO[$idx]}"
    LISTA_PATRONES=( $(echo ${REEMPLAZOS[$idx]} | sed "s/${SEPARADOR_REEMPLAZO}/ /g" ) )
    DETECTION_PATTERN="${LISTA_PATRONES[0]}"
    REPLACE_PATTERN="${LISTA_PATRONES[1]}"

    # 2.2.- Reemplaza Patrones
    sed -i "s/${DETECTION_PATTERN}/${REPLACE_PATTERN}/g" "${DEST_FILE}"

done

# 3.- Genera Fichero de Validacion
FICHERO_RESULTADOS="/tmp/configInstancia.txt"
LISTA=( $(echo "${FICHERO[@]}" | tr ' ' '\n' | sort -u | tr '\n' ' ' ) )
echo '' > "${FICHERO_RESULTADOS}"
for CONFIG in "${LISTA[@]}; do

    # 3.1.- Encabezado
    echo '-----' >> "${FICHERO_RESULTADOS}"
    echo "    FICHERO CONFIGURACION = ${CONFIG} " >> "${FICHERO_RESULTADOS}"
    echo '-----' >> "${FICHERO_RESULTADOS}"

    # 3.2.- Fichero Configuración modificado
    cat "${CONFIG}" >> "${FICHERO_RESULTADOS}"

done
```

## Automatización Modificaciones en Frío: Script desde Anfitrión

```
#!/bin/bash
# PRECONDICIONES:
# 1.- Instancia Definida
# 2.- Instancia Apagada
WORKING_PATH="/root/exercises"
NOMBRE_INSTANCIA="CWP.rhel8.4"
LOGICA_ADAPTACION="${WORKING_PATH}/configura.sh"
DATOS_ADAPTACION="${WORKING_PATH}/${NOMBRE_INSTANCIA}.txt"
RESULTADO_ADAPTACION="${WORKING_PATH}/resultado_${NOMBRE_INSTANCIA}.txt"
VALIDA_ADAPTACION="${WORKING_PATH}/valida_${NOMBRE_INSTANCIA}.txt"

# 1.- Carga Logica y Datos de Las Modificaciones en Frio
virt-copy-in -d "${NOMBRE_INSTANCIA}" "${LOGICA_ADAPTACION}" "/tmp/."
virt-copy-in -d "${NOMBRE_INSTANCIA}" "${DATOS_ADAPTACION}" "/tmp/."

# 2.- Aplica modificaciones
virt-customize -d "${NOMBRE_INSTANCIA}" --run-command "mv /tmp/${NOMBRE_INSTANCIA}.txt /tmp/instancia.txt"
virt-customize -d "${NOMBRE_INSTANCIA}" --run-command "chmod 777 /tmp/configura.sh"
virt-customize -d "${NOMBRE_INSTANCIA}" --run-command "sh /tmp/configura.sh"

# 3.- Extrae fichero configuraciones
virt-copy-out -d "${NOMBRE_INSTANCIA}" "/tmp/configInstancia.txt" "${WORKING_PATH}"
mv "${WORKING_PATH}/configInstancia.txt" "${RESULTADO_ADAPTACION}"
diff "${RESULTADO_ADAPTACION}" "${VALIDA_ADAPTACION}"
```

## Monitorización CPU y RAM (en caliente)

**virt-top**

## Monitorización de Almacenamiento (en frío)

```
____ 1.- Particiones y alineamiento
virsh list --all
virsh shutdown <DOMAIN>
virt-df # Si se saturan particiones... existe procedimiento
virt-alignment-scan # Si surgen errores... hay que regenerar instancia

____ 2.- Localizar Disco VM
virsh list --all
virsh domblklist <CWP.rhel8.4> --inactive --details

____ 3.- Detectar Estado del Disco
qemu-img info <CWP.rhel8.4.qcow2>
```

### 2.3.2.2. Instancias de Cometa



**MAQUINA VIRTUAL:  
COMETA**

  
**CWP**

**Parar Instancia**

```
poweroff
shutdown now
halt # ¡¡ OJO !! RHEL 5,6... RHEL 7,8 necesita -p para que corte electricidad
```

**Puesta en Marcha (Adaptación en Caliente): PASO 1 –Entorno**

**PRENCONDICIONES**

1) Instancia Definida  
2) Instancia Validada: nunca arranca, o checksum verificados

**1.- Verificación Checksum**

**1.- Extraer Checksum de Lo Máquina Virtual Validada por Enaire**

```
# sha256sum CWP.rhel8.4.home.qcow2 > checksums.log
# sha256sum CWP.rhel8.4.system.qcow2 >> checksums.log
# sha256sum CWP.rhel8.4_VARS.fd >> checksums.log
# sha256sum XR12_CWP.rhel8.4.xml >> checksums.log
```

**2.- Comparación Automatica Checksum con el fichero de Checksums suministrado**

```
# diff CWP.rhel8.4.sha256 checksums.log
4d3
< 5f907881140ca1b51cfe96219b6b4d51c5d00f8d3ccdc00ac56ec836d39e786e XE2420_CWP.rhel8.4.xml
```

**3.- Comparación Manual Checksum con el fichero de Checksums suministrado**

```
# cat CWP.rhel8.4.sha256
783e0c1388691c50f6a2fb87bade08bfffce3a8140a90d841a0cf63bfb84bcef6 CWP.rhel8.4.home.qcow2
16d6875b0b145a94a8fb705d3bf9ef803349f4f81b54f328658b2235eaa7faf6 CWP.rhel8.4.system.qcow2
ff75f2a9b5eb65aa92c5d8c05f5d736ab054f5509e6907918727ab9165f43f18 CWP.rhel8.4_VARS.fd
5f907881140ca1b51cfe96219b6b4d51c5d00f8d3ccdc00ac56ec836d39e786e XE2420_CWP.rhel8.4.xml
995260e3cbf29c8bd7fea10eefddb6aebc663ff7e5d66a08ea688b09ac94808 XR12_CWP.rhel8.4.xml
```

```
# cat checksums.log
783e0c1388691c50f6a2fb87bade08bfffce3a8140a90d841a0cf63bfb84bcef6 CWP.rhel8.4.home.qcow2
16d6875b0b145a94a8fb705d3bf9ef803349f4f81b54f328658b2235eaa7faf6 CWP.rhel8.4.system.qcow2
ff75f2a9b5eb65aa92c5d8c05f5d736ab054f5509e6907918727ab9165f43f18 CWP.rhel8.4_VARS.fd
995260e3cbf29c8bd7fea10eefddb6aebc663ff7e5d66a08ea688b09ac94808 XR12_CWP.rhel8.4.xml
```

**2.- Transferencia de Ficheros**

```
virt-copy-in -d CWP.rhel8.4 H-CWP_CWPA_CWPA_03.02.63_01.00.tar.gz /var/tmp/.
virt-copy-in -d CWP.rhel8.4 H-CWP_CWPB_CWPB_03.02.63_01.01.tgz /var/tmp/.
```

**3.- Arranque de Instancia**

```
# virsh start CWP.rhel8.4 --console
>>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Ctrl+5
User: root
Password: focus
[root@posra1 ~]# cd /var/tmp
[root@posra1 tmp]# ls -l
Total 43692
-rw-r--r-- 1 root root 31094352 Oct 18 12:00 H-CWP_CWPA_CWPA_03.02.63_01.00.tar.gz
-rw-r--r-- 1 root root 13635364 Oct 18 12:00 H-CWP_CWPB_CWPB_03.02.63_01.01.tgz
drwx----- 3 root root 17 Nov 9 15:53 system-private-bb3...-chronyd.service-NEgvej
drwx----- 3 root root 17 Nov 9 15:53 system-private-bb3...-ModemManager.service-JcC4jf
```

```
_____ 4.- Modificar Ficheros de Configuración
// PANTALLAS ELO
_____ 4.1.- Modificar el enlace simbólico de la pantalla en /etc/xorg.conf.d/
[root@posra1 ~]# cd /etc/xorg.conf.d/
[root@posra1 xorg.conf.d]# ll
[root@posra1 xorg.conf.d]# unlink 50-screen-cometa.conf
[root@posra1 xorg.conf.d]# ln -s 59-screen-cometa-elo.conf 50-screen-cometa.conf
[root@posra1 xorg.conf.d]# ll

_____ 4.2.- Status del Servicio que gestiona este tipo de pantallas
[root@posra1 ~]# systemctl status eloser.service
• eloser.service - Invoke Elo serial service at system startup.
  Loaded: loaded (/etc/systemd/system/eloser.service; disabled; vendor preset: >
  Active: inactive (dead)

_____ 4.3.- Activar Servicio, si estuviese parado o inhabilitado
[root@posra1 ~]# systemctl enable --now eloser.service
Created symlink /etc/systemd/system/default.target.wants/eloser.service →
/etc/systemd/system/eloser.service.

_____ 4.4.- Comprobar el status de activo y habilidad
[root@posra1 ~]# systemctl status eloser.service
• eloser.service - Invoke Elo serial service at system startup.
  Loaded: loaded (/etc/systemd/system/eloser.service; enabled; vendor preset: >
  Active: active (running) since Mon 2024-04-29 13:18:48 UTC; 2min 52s ago
  CGroup: /system.slice/eloser.service
          └─737340 /etc/opt/elo-ser/eloser ttyS1

// Lista Ficheros de Configuración a modificar
1. /etc/sysconfig/network-scripts/ifcfg-bond0
2. /etc/hostname
3. /vcs/runtime/ETIQUETA_HW >>> Revisaremos TIPO_MAQUINA e INSTANCIA
4. /etc/hosts
5. /vcs/runtime/cfg/TMCS_AGTM_cfg_multi_agentConf.cnf
```

Nº LINEA	AJUSTE	EJEMPLO
8	MCAST_AGENTE = 225.centro.vlan_voz.11	MCAST_AGENTE = 225.10.61.11
77	MCAST_GRP = 225.centro.vlan_voz.10	MCAST_GRP = 225.10.61.10
86	RED_VOIP = 10.centro.vlan_voz	RED_VOIP = 10.10.61
98	IP_FTP_VOIP = 10.centro.vlan_voz.tmcs1	IP_FTP_VOIP = 10.10.61.72

## Puesta en Marcha (Adaptación en Caliente): PASO 2 – Aplicativo

```
_____ 1.- Ir a carpeta de COTS
[root@posra1 ~]# cd /var/tmp
[root@posra1 tmp]# ls -l
Total 43692

-rw-r--r-- 1 root root 31094352 Oct 18 12:00 H-CWP_CWPA_CWPA_03.02.63_01.00.tar.gz
-rw-r--r-- 1 root root 13635364 Oct 18 12:00 H-CWP_CWPB_CWPB_03.02.63_01.01.tgz
drwx----- 3 root root 17 Nov 9 15:53 system-private-bb3...-chronyd.service-NEgvej
drwx----- 3 root root 17 Nov 9 15:53 system-private-bb3...-ModemManager.service-JcC4jf

_____ 2.- Instalar COTS

SW-A:
[root@posra1 ~]# tar xvf H-CWP_CWPA_CWPA_03.02_63_01.00.tar.gz
[root@posra1 ~]# ./install.sh
[root@posra1 ~]# chown -R vcs:vcs /vcs
[root@posra1 ~]# rm -f install.sh cots-rhel-5.5.tar.gz cots-rhel-8.4.tar.gz H-CWP*
```

**SW-B:**

```
[root@posra1 ~]# tar xvf H-CWP_CWPB_CWPB_03.02_63_01.01.tgz
[root@posra1 ~]# instaladorCOTS.sh cots_CWPB.tar.gz
[root@posra1 ~]# chown -R vcs:vcs /vcs
[root@posra1 ~]# rm -f instaladorCOTS.sh cots_CWPB.tar.gz H-CWP*
```

*Editaremos el fichero /etc/rc.local, borrando el carácter almohadilla de la línea 15 para habilitar el autoinicio del aplicativo y el entorno gráfico que este necesita.*

### **Puesta en Marcha (Adaptación en Caliente): PASO 3 – Arranque**

\_\_\_\_ 1.- Arranque de aplicativo en reinicios (editar /etc/rc.local... línea 15, borrar almohadilla)

\_\_\_\_ 2.- Reiniciar La Máquina Virtual

```
[root@posra1 ~]# reboot
```

\_\_\_\_ 3.- Acceder como usuario vcs

```
# virsh start CWP.rhel8.4 --console
```

>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Crtl+5

User: root

Password: focus

```
[root@posra1 ~]# su - vcs
```

```
[vcs@CWP2 runtime]$
```

\_\_\_\_ 4.- Esperar actualización de aplicativo (1 minuto, aprox.) y reiniciar

```
[vcs@CWP2 runtime]$ cd /vcs/runtime
```

```
[vcs@CWP2 runtime]$ ./shutdown.sh
```

\_\_\_\_ 5.- Desmoldar ficheros y forzar inicio Cometa

**SW-A:**

```
$ cp CWP_CWPA_cfg_multi_paramPuesto.txt molde CWP_CWPA_cfg_multi_paramPuesto.txt
```

```
$ cp ESTA_AGES_cfg_multi_localstagent.ini molde ESTA_AGES_cfg_multi_localstagent.ini
```

```
$ vi CWP_CWPA_cfg_multi_paramPuesto.txt:
```

1. En la línea 37 cambiaremos el tipo de UGA, según proceda:

```
[T_UGA]
```

```
** Tipo de UGA: UGA2 o UGA3
```

```
TIPO_UGA= UGA3,
```

2. A partir de la línea 67, invertiremos el orden de ALTAVOZ\_ED y ALTAVOZ\_TF:

```
ALTAVOZ_RD= 1, >>> cambiar a 3
```

```
ALTAVOZ_TF= 3, >>> cambiar a 1
```

```
CASCOS_RD= 2,
```

```
CASCOS_TF= 4,
```

```
[vcs@posra1 runtime]$ ./startup.sh >>> Mostrará fallo de configuración
```

```
[vcs@posra1 runtime]$ ./shutdown.sh
```

**SW-B:**

```
$ cp CWP_CWPB_cfg_multi_paramsInstall.ini molde CWP_CWPB_cfg_multi_paramsInstall.ini
```

```
$ cp CWP_CWPB_cfg_multi_params.rdlc.ini CWP_CWPB_cfg_multi_params.ini
```

```
$ cp ESTA_AGES_cfg_multi_localstagent.ini molde ESTA_AGES_cfg_multi_localstagent.ini
```

```
$ cp CWP_CWPB_cfg_multi_tonesVolume.ini molde CWP_CWPB_cfg_multi_tonesVolume.ini
```

NOTA: Los valores de CWP\_CWPB\_cfg\_multi\_tonesVolume.ini serán genéricos, pero deben ser definidos por el centro, pudiéndose copiar desde una HP Z840 ya configurada

\_\_\_\_ 6.- Recuperar la configuración activa desde el tmc1, cuando éste se encuentre como principal

**SW-A y SW-B:**

```
[vcs@posra1 runtime]$ cd /vcs/runtime/dat/
```

```
[vcs@posra1 dat]$ scp tmc1:/vcs/runtime/dat/fichconf/config* .
```

**SW-B:**

```
[vcs@posra1 dat]$ cp /vcs/runtime/dat/config* /vcs/runtime/dat/emergency/
```

\_\_\_\_ 7.- Iniciar Aplicación

```
[vcs@posra1 runtime]$ ./startup.sh
```

## Propagación (Adaptación en Frío): PASO 1 – Entorno

### PRECONDICIONES

- Instancia Definida (Transición Define).
- Instancia nunca Arrancada, para evitar colisiones con otras instancias.
- QCOW Desfragmentado: antes de la transferencia, aplicar procedimiento desfragmentación

### 1.- Ficheros sobre los que debe actuar la lógica

- /etc/hostname
- /etc/sysconfig/network-scripts/ifcfg-enp7s0 >>> Auxiliar/Gestión/CAU
- /etc/sysconfig/network-scripts/ifcfg-bond0 >>> V1 + V2
- /vcs/runtime/ETIQUETA\_HW

## Propagación (Adaptación en Frío): PASO 2 – Verificación

### 1.- Comprobaciones de los cambios

```
guestfish -rw -d CWP.rhel8.4 -i
Welcome to guestfish, the guest filesystem shell for
Editing virtual machine filesystems and disk images.
Type: 'help' for help on commands
'man' to read the manual
'quit' to quit the shell
Sistema operativo: Red Hat Enterprise Linux release 8.4 (Ootpa)
/dev/sda4 montado en /
/dev/sda2 montado en /boot
/dev/sda1 montado en /boot/efi
/dev/sda6 montado en /tmp
/dev/sda7 montado en /usr
/dev/sda5 montado en /var
/dev/sda3 montado en /var/crash
/dev/sdb1 montado en /vcs
><fs> _
><fs> vi /etc/hostname
><fs> vi /etc/sysconfig/network-scripts/ifcfg-bond0
><fs> vi /vcs/runtime/ETIQUETA_HW
><fs> exit (!) IMPORTANTE: Si no cerramos bien el intérprete pueden surgir fallos
```

### 2.- Comprobar que esta todo OK

```
virsh start CWP.rhel8.4
```

## Propagación (Adaptación en Frío): PASO 3 – Arranque

```
# virsh start CWP.rhel8.4 --console
>>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Ctrl+5
User: root
Password: focus
[root@posra1 ~]# ip a
```

## Desfragmentación del Disco Virtual

### 1.- Rellenar con '0' el disco duro

```
virsh start CWP.rhel8.4 --console
dd if=/dev/zero of=/vcs/ficherazo bs=20M
rm /vcs/ficherazo
```

### 2- Compactar QCOW

// Compactado

```
virsh shutdown CWP.rhel8.4
mv /kvm/vhd/CWP.rhel8.4.home.qcow2 /kvm/vhd/CWP.rhel8.4.home.qcow2.LARGE
qemu-img convert /kvm/vhd/CWP.rhel8.4.home.qcow2.LARGE \
-O qcow2 /kvm/vhd/CWP.rhel8.4.home.qcow2
```

// Comprobaciones

```
ll /kvm/vhd/
virsh start CWP.rhel8.4
// Eliminar antiguo disco duro
rm -f /kvm/vhd/CWP.rhel8.4.home.qcow2.LARGE
rm -f /kvm/vhd/CWP.rhel8.4.home.qcow2.old
```

### 2.3.2.3. Instancias de OUCS



## MAQUINA VIRTUAL: OUCS



### OUCS

#### Parar Instancia

```
poweroff  
shutdown now
```

#### Puesta en Marcha (Adaptación en Caliente): PASO 1 - Entorno

##### PRECONDICIONES

- 1) Instancia Definida
- 2) Instancia Validada: nunca arranca, o checksum verificados

##### 1.- Verificación Checksum

1.- Extraer Checksum de Lo Máquina Virtual Validada por Enaire

```
# sha256sum OUCS.rhel7.7.qcow2 > checksumsPOS.log  
# sha256sum OUCS.rhel7.7_VARS.fd >> checksumsPOS.log  
# sha256sum XR12_OUCS.rhel7.7.xml >> checksumsPOS.log
```

2.- Comparación Automatica Checksum con el fichero de Checksums suministrado

```
# diff OUCS.rhel7.7.sha256 checksumPOS.log
```

3.- Comparación Manual Checksum con el fichero de Checksums suministrado

```
# cat OUCS.rhel7.7.sha256  
fcea21fd0ebab2cdc005d9a8e814cd94421328ef6ce7ac6d0cf12068c716f806 OUCS.rhel7.7.qcow2  
4b404005a0150e14ee71d30e21cf65753d44529c313123f2e333d0fd9be501f1 OUCS.rhel7.7_VARS.fd  
d1e20e173874db0e12b16ebbd8fd448ad7da5228defbabbef1c2fbfa75083a31 XR12_OUCS.rhel7.7.xml  
# cat checksumsPOS.log  
fcea21fd0ebab2cdc005d9a8e814cd94421328ef6ce7ac6d0cf12068c716f806 OUCS.rhel7.7.qcow2  
4b404005a0150e14ee71d30e21cf65753d44529c313123f2e333d0fd9be501f1 OUCS.rhel7.7_VARS.fd  
d1e20e173874db0e12b16ebbd8fd448ad7da5228defbabbef1c2fbfa75083a31 XR12_OUCS.rhel7.7.xml
```

##### 2.- Transferencia de Ficheros

```
virt-copy-in -d OUCS.rhel7.7 TAR_SACTA.tgz /var/tmp/.
```

##### 3.- Arranque de Instancia

```
# virsh start UCS.rhel7.7--console
```

>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Crtl+5

User: root

Password: focus

```
[root@pos-base /]# cd /var/tmp
```

```
[root@pos-base tmp]# ls -l
```

```
-rw-r--r-- 1 root root xxxxxxx Oct 18 12:00 TAR_SACTA.tgz
```

##### 4.- Modificar Ficheros de Configuración

1.- Nombre de La máquina.

```
[root@ucs-base /]# /bin/hostnamectl set-hostname ruta1
```

```
[root@ucs-base /]# echo "ruta1" > /etc/hostname
```

##### 2.- HOSTS

```
[root@ucs-base /]# vim /etc/hosts
```

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
```

```
:::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
```

```
10.1.50.51 ruta1-int ruta1 ruta1.sacta loghost
```

```
10.1.50.52 posra1-int posa_spv
```

```
10.1.50.54 posrb1-int posb_spv
```

```
10.40.6.131 ruta1-c1
```

```
10.40.7.131 ruta1-c2
```

```
10.40.40.71 ruta1-gest
```



### \_\_\_ 3.- Direcciones IP (no hay Network Manager)

```
[root@ucs-base /]# vi /etc/sysconfig/network-scripts/ifcfg-enp9s0f0 >>> C1
[root@ucs-base /]# vi /etc/sysconfig/network-scripts/ifcfg-enp9s0f1 >>> C2
[root@ucs-base /]# vi /etc/sysconfig/network-scripts/ifcfg-enp9s0f2 >>> Auxiliar/Gestión/CAU
[root@ucs-base /]# vi /etc/sysconfig/network-scripts/ifcfg-enp9s0f3 >>> Interna
[root@ucs-base /]# for CON in $(nmcli -t -f uuid con) ; do nmcli con up $CON ; done
```

### \_\_\_ 4.- SYSCTL.CONF (activar broadcast)

```
[root@ucs-base /]# vim /etc/sysctl.conf
// Insertar lineas
#MODIFICACIONES PARA QUE RESPONDA A LOS ICMP BROADCAST
net.ipv4.icmp_echo_ignore_broadcasts = 0
```

```
[root@ucs-base /]# sysctl -p
```

### \_\_\_ 5.- Demonio DIGI (IP de DIGI)

```
[root@ucs-base /]# vim /etc/dgrp.backing.store
```

```
1 10.1.50.254 8 auto default default default default never default default
root@ucs-base $ /etc/init.d/dgrp_daemon restart
```

### \_\_\_ 6.- Workaround NFS

```
[root@ucs-base /]# su - sacta
root@ucs-base $ vim /sacta/version/ejecución/fs_gen_def_entorno.local
```

# Aunque la version MAJOR de RHEL es 7, se debe definir 6 para que monte NFS del servidor de ficheros.

```
RHEL_RELEASE_MAJ=6
export RHEL_RELEASE_MAJ
```

## Puesta en Marcha (Adaptación en Caliente): PASO 2 - Aplicativo

### \_\_\_ 1.- Desempaquetado de aplicativo

```
[root@usc-base /]# cd /sacta_home
[root@ucs-base /]# tar xf /var/tmp/TAR_SACTA.tgz
[root@ucs-base /]# chown -Rh sacta:general sacta_v04
```

### \_\_\_ 2.- Adecuar Los ficheros ETIQUETA.TXT y DEPENDENCIA.TXT

```
[root@ucs-base /]# su - sacta
root@ucs-base $ cd /sacta/version/ejecucion
root@ucs-base $ echo UCS_RUTA_1 > ETIQUETA.TXT
root@ucs-base $ echo RSIM > DEPENDENCIA.TXT
root@ucs-base $ ms_gen_sactad restart
```

### \_\_\_ 3.- Adecuar el fichero inittab

```
sacta@ucs-base $ exit
[root@ucs-base /]# vi /etc/inittab
// Cambiar de nivel 1 al nivel 5
id:5:initdefault:
```

### \_\_\_ 4.- Reiniciar La maquina

```
[root@ucs-base /]# reboot
```

## Puesta en Marcha (Adaptación en Caliente): PASO 3 - Arranque

### \_\_\_ 1.- Entrar en La instancia

```
# virsh console UCS.rhel7.7
>>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Ctrl+5
User: root
Password: focus
```

### \_\_\_ 2.- Esperar actualización de aplicativo (1 minuto, aprox.) y reiniciar

## Propagación (Adaptación en Frío): PASO 1 - Entorno

### PRECONDICIONES

- Instancia Definida (Transición Define).
- Instancia nunca Arrancada, para evitar colisiones con otras instancias.
- QCOW Desfragmentado: antes de la transferencia, aplicar procedimiento desfragmentación

### 1.- Ficheros sobre los que debe actuar la lógica

- /etc/hostname
- /etc/sysconfig/network-scripts/ifcfg-enp9s0f0 >>> C1
- /etc/sysconfig/network-scripts/ifcfg-enp9s0f1 >>> C2
- /etc/sysconfig/network-scripts/ifcfg-enp9s0f2 >>> Auxiliar/Gestión/CAU
- /etc/sysconfig/network-scripts/ifcfg-enp9s0f3 >>> Interna
- /sacta/version/ejecucion/ETIQUETA.txt
- /sacta/version/ejecucion/DEPENDENCIA.txt
- /sacta/version/ejecucion/fs\_pos\_def\_entorno.local
- /etc/hosts
- /etc/sysctl.conf >>> Verificar su configuración
- /etc/dgrp.backing.store >>> Verificar su configuración

## Propagación (Adaptación en Frío): PASO 2 – Verificación

1.- Desde este intérprete, se comprueban los ficheros de la nueva instancia

```
[root@UCS01 ~]# cd /kvm/vhd
```

```
[root@UCS01 vhd]# guestfish -rw -d OUCS.rhel7.7 -i
```

Welcome to guestfish, the guest filesystem shell for  
Editing virtual machine filesystems and disk images.

Type: 'help' for help on commands

'man' to read the manual

'quit' to quit the shell

Sistema operativo: Red Hat Enterprise Linux release 8.4 (Ootpa)

/dev/sda4 montado en /

/dev/sda2 montado en /boot

/dev/sda1 montado en /boot/efi

/dev/sda3 montado en /sacta\_home

/dev/sdb5 montado en /var/crash

```
><fs> cat /etc/hostname >>> En RHEL7.7, el hostname se almacena aquí
```

```
><fs> cat /etc/sysconfig/network-scripts/ifcfg-enp9s0f0 >>> C1
```

```
><fs> cat /etc/sysconfig/network-scripts/ifcfg-enp9s0f1 >>> C2
```

```
><fs> cat /etc/sysconfig/network-scripts/ifcfg-enp9s0f2 >>> Auxiliar/Gestión/CAU
```

```
><fs> cat /etc/sysconfig/network-scripts/ifcfg-enp9s0f3 >>> Interna
```

```
### Resto de ficheros SACTA a editar (ETIQUETA.TXT, DEPENDENCIA.TXT, /etc/hosts, ...) ###
```

```
><fs> exit >>> IMPORTANTE, si no cerramos bien el intérprete pueden surgir fallos
```

## Propagación (Adaptación en Frío): PASO 3 – Arranque

```
# virsh start UCS.rhel7.7--console
```

```
>>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Ctrl+5
```

```
User: root
```

```
Password: focus
```

```
[root@pos-base /]# ip a
```

## Desfragmentación Disco Virtual

```
___ 1.- Rellenar con '0' el disco duro
virsh start UCS.rhel7.7 --console
dd if=/dev/zero of=/sacta_home/ficherazo bs=20M
rm /sacta_home/ficherazo

___ 2- Compactar QCOW
// Compactado
virsh shutdown UCS.rhel7.7
mv /kvm/vhd/ UCS.rhel7.7.qcow2 /kvm/vhd/ UCS.rhel7.7.qcow2.LARGE
qemu-img convert /kvm/vhd/UCS.rhel7.7.qcow2.LARGE \
                -O qcow2 /kvm/vhd/UCS.rhel7.7.qcow2
// Comprobaciones
ll /kvm/vhd/
virsh start UCS.rhel7.7

// Eliminar antiguo disco duro
rm -f /kvm/vhd/UCS.rhel7.7.qcow2.LARGE
rm -f /kvm/vhd/UCS.rhel7.7.qcow2.old
```

### 2.3.2.4. Instancias de POS



## MAQUINA VIRTUAL: POS



**POS**

### Parar Instancia

```
halt -p # Sin -p, no corta suministro eléctrico
```

### Puesta en Marcha (Adaptación en Caliente): PASO 1 - Entorno

#### \_\_\_ PRENCONDICIONES

- 1) Instancia Definida
- 2) Instancia Validada: nunca arranca, o checksum verificados

#### \_\_\_ 1.- Verificación Checksum

\_\_\_ 1.- Extraer Checksum de Lo Máquina Virtual Validada por Enaire

```
# sha256sum POS.rhel6.8.qcow2 > checksumsPOS.log
```

```
# sha256sum POS.rhel6.8_VARS.fd >> checksumsPOS.log
```

```
# sha256sum XR12_POS.rhel6.8.xml >> checksumsPOS.log
```

\_\_\_ 2.- Comparación Automática Checksum con el fichero de Checksums suministrado

```
# diff POS.rhel6.8.sha256 checksumPOS.log
```

\_\_\_ 3.- Comparación Manual Checksum con el fichero de Checksums suministrado

```
# cat POS.rhel6.8.sha256
```

```
fcea21fd0ebab2cdc005d9a8e814cd94421328ef6ce7ac6d0cf12068c716f806 POS.rhel6.8.qcow2
```

```
4b404005a0150e14ee71d30e21cf65753d44529c313123f2e333d0fd9be501f1 POS.rhel6.8_VARS.fd
```

```
d1e20e173874db0e12b16ebbd8fd448ad7da5228defbabbef1fc2fbfa75083a31 XR12_POS.rhel6.8.xml
```

```
# cat checksumsPOS.log
```

```
fcea21fd0ebab2cdc005d9a8e814cd94421328ef6ce7ac6d0cf12068c716f806 POS.rhel6.8.qcow2
```

```
4b404005a0150e14ee71d30e21cf65753d44529c313123f2e333d0fd9be501f1 POS.rhel6.8_VARS.fd
```

```
d1e20e173874db0e12b16ebbd8fd448ad7da5228defbabbef1fc2fbfa75083a31 XR12_POS.rhel6.8.xml
```

#### \_\_\_ 2.- Transferencia de Ficheros

```
virt-copy-in -d POS.rhel6.8 TAR_SACTA.tgz /var/tmp/.
```

```
____ 3.- Arranque de Instancia
# virsh start POS.rhel6.8 --console
>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Crtl+5
User: root
Password: focus
[root@pos-base /]# cd /var/tmp
[root@pos-base tmp]# ls -l
-rw-r--r-- 1 root root xxxxxxx Oct 18 12:00 TAR_SACTA.tgz
____ 4.- Modificar Ficheros de Configuración
____ 1.- Nombre de la máquina.
[root@pos-base~] vim /etc/sysconfig/network --> Editar la línea: HOSTNAME=posra1

____ 2.- HOSTS
// Ejemplo para una posición de EJECUTIVO:
[root@pos-base /]# vim /etc/hosts
127.0.0.1 localhost.localdomain localhost
::1 localhost6.localdomain6 localhost6
10.1.50.51 ruta1-int ucs_spv
10.1.50.52 posra1-int posa_spv posra1 posra1.sacta loghost
10.1.50.54 posrb1-int posb_spv
10.1.50.254 digi
10.40.8.131 posra1-r1
10.40.9.131 posra1-r2
10.40.40.72 posra1-gest
// Ejemplo para una posición de PLANIFICADOR:
[root@pos-base /]# vim /etc/hosts
127.0.0.1 localhost.localdomain localhost
::1 localhost6.localdomain6 localhost6
10.1.50.51 ruta1-int ucs_spv
10.1.50.52 posra1-int posa_spv
10.1.50.54 posrb1-int posb_spv posrb1 posrb1.sacta loghost
10.1.50.254 digi
10.40.8.132 posrb1-r1
10.40.9.132 posrb1-r2
10.40.40.73 posrb1-gest

____ 3.- Direcciones IP (no hay Network Manager)
[root@pos-base /]# vi /etc/sysconfig/network-scripts/ifcfg-eth0 >>> R1
[root@pos-base /]# vi /etc/sysconfig/network-scripts/ifcfg-eth1 >>> R2
[root@pos-base /]# vi /etc/sysconfig/network-scripts/ifcfg-eth2 >>> Auxiliar/Gestion/CAU
[root@pos-base /]# vi /etc/sysconfig/network-scripts/ifcfg-eth3 >>> Interna
[root@pos-base /]# for DEV in $(ls /etc/sysconfig/network-scripts) ; do ifup ${DEV##*-} ;
done

____ 4.- SYSCTL.CONF (activar broadcast)
[root@pos-base /]# vim /etc/sysctl.conf

// Insertar líneas
#MODIFICACIONES PARA QUE RESPONDA A LOS ICMP BROADCAST
net.ipv4.icmp_echo_ignore_broadcasts = 0

[root@pos-base /]# sysctl -p

____ 5.- XORG.CONF (enlace simbolico ejecutivo/planificador)
[root@pos-base /]# cd /etc/X11
[root@pos-base /]# ls -l xorg.conf
[root@pos-base /]# ln -s xorg.conf_ejecutivo xorg.conf // Caso Ejecutivo
[root@pos-base /]# ln -s xorg.conf_planificador xorg.conf // Caso Planificador
```

## Puesta en Marcha (Adaptación en Caliente): PASO 2 - Aplicativo

```
___ 1.- Desempaquetado de aplicativo
[root@pos-base /]# cd /sacta_home
[root@pos-base /]# tar xf /var/tmp/TAR_SACTA.tgz
[root@pos-base /]# chown -Rh sacta:general sacta_v04

___ 2.- Revisar enlace simbólico
[root@pos-base /]# ls -la # /sacta -> /sacta_home/sacta_v04

___ 3.- Adecuar los ficheros ETIQUETA.TXT y DEPENDENCIA.TXT
[root@pos-base /]# su - sacta
sacta@pos-base $ cd /sacta/version/ejecucion
sacta@pos-base $ echo POS_RUTA_A_1 > ETIQUETA.TXT # Depende Centro
sacta@pos-base $ echo RSIM > DEPENDENCIA.TXT # Depende Centro
sacta@pos-base $ ms_gen_sactad restart

___ 4.- Adecuar el fichero fs_pos_def_entorno.Local => REVISAR
sacta@pos-base $ vi /sacta/versión/ejecución/fs_pos_def_entorno.local
EJECUTIVO:
BORDE_CRUCE="*misc.bordeCruce: DERECHA"
IP_TABLET=10.1.50.10:0
PLANIFICADOR:
BORDE_CRUCE="*misc.bordeCruce: IZQUIERDA"
IP_TABLET=10.1.50.20:0

___ 5.- Adecuar el fichero inittab
sacta@pos-base $ exit
[root@pos-base /]# vi /etc/inittab
// Cambiar de nivel 1 al nivel 5
id:5:initdefault:

___ 6.- Reiniciar La maquina
[root@pos-base /]# reboot
```

## Puesta en Marcha (Adaptación en Caliente): PASO 3 - Arranque

```
___ 1.- Entrar en la instancia
# virsh console POS.rhel6.8
>>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Ctrl+5
User: root
Password: focus

___ 2.- Esperar actualización de aplicativo (1 minuto, aprox.) y reiniciar
___ 3.- Adaptaciones POS para la Tablet
# sum TAR_SSH_ID_RSA_IFOCUCS.gz
06509 2
// Como root
# cp -p TAR_SSH_ID_RSA_IFOCUCS.gz /root/.
# cd /root
# tar xvf TAR_SSH_ID_RSA_IFOCUCS.gz
# su - sacta
// Como sacta
$ cp -p TAR_SSH_ID_RSA_IFOCUCS.gz /sacta/version/ejecucion/.
$ cd /sacta/version/ejecucion
$ tar xvf TAR_SSH_ID_RSA_IFOCUCS.gz

___ 4.- Adaptaciones Tablet
Para TABLET A:
# vi /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
```

```
10.1.50.10 TABLET_A TABLET_A.sacta
10.1.50.52 posra1 ntpserver1 POS_TABLET
10.1.50.54 posrb1 ntpserver2
```

```
ETIQUETA.TXT /sacta_home/sacta_4.0/version/ejecucion/ETIQUETA.TXT (para no olvidar)
# echo "TABLET_A.sacta" > /sacta/version/ejecucion/ETIQUETA.TXT
```

**Para TABLET B:**

```
# vi /etc/hosts
```

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
10.1.50.20 TABLET_B TABLET_B.sacta
10.1.50.52 posra1 ntpserver1
10.1.50.54 posrb1 ntpserver2 POS_TABLET
```

ETIQUETA.TXT

```
# echo "TABLET_B.sacta" > /sacta/version/ejecucion/ETIQUETA.TXT
```

\_\_\_ 5.- Instalar el Software

// Como root

```
# cp -p TAR_SSH_ID_RSA_IFOCUCS.gz /root/.
# cd /root
# tar xvf TAR_SSH_ID_RSA_IFOCUCS.gz
# su - sacta
```

// Como sacta

```
$ cp -p TAR_SSH_ID_RSA_IFOCUCS.gz /sacta_home/sacta/.
$ cd /sacta_home/sacta
$ tar xvf TAR_SSH_ID_RSA_IFOCUCS.gz
```

\_\_\_ 6.- Revisar Notas Técnicas de la versión del sistema y las configuraciones específicas de cada centro.

## Propagación (Adaptación en Frío): PASO 1 - Entorno

### PRECONDICIONES

- Instancia Definida (Transición Define).
- Instancia nunca Arrancada, para evitar colisiones con otras instancias.
- QCOW Desfragmentado: antes de la transferencia, aplicar procedimiento desfragmentación

### 1.- Ficheros sobre los que debe actuar la lógica

- /etc/sysconfig/network >>> En RHEL6.8, el hostname se almacena aquí
- /etc/sysconfig/network-scripts/ifcfg-eth0 >>> R1
- /etc/sysconfig/network-scripts/ifcfg-eth1 >>> R2
- /etc/sysconfig/network-scripts/ifcfg-eth2 >>> Auxiliar/Gestión/CAU
- /etc/sysconfig/network-scripts/ifcfg-eth3 >>> Interna
- /sacta/version/ejecucion/ETIQUETA.txt
- /sacta/version/ejecucion/DEPENDENCIA.txt
- /sacta/version/ejecucion/fs\_pos\_def\_entorno.local
- /etc/hosts
- /etc/X11/xorg.conf >>> Modificar enlace simbólico
- /etc/sysctl.conf >>> Verificar su configuración

## Propagación (Adaptación en Frío): PASO 2 - Verificación

```
1.- Desde este intérprete, editaremos los ficheros de la nueva instancia
# cd /kvm/vhd
# guestfish-rw -d POS.rhel6.8 -i

Welcome to guestfish, the guest filesystem shell for
Editing virtual machine filesystems and disk images.
Type: 'help' for help on commands
'man' to read the manual
'quit' to quit the shell

Sistema operativo: Red Hat Enterprise Linux release 8.4 (Ootpa)
/dev/sda5 montado en /
/dev/sda2 montado en /boot
/dev/sda1 montado en /boot/efi
/dev/sda3 montado en /sacta_home
/dev/sda7 montado en /tmp
/dev/sda8 montado en /usr
/dev/sda6 montado en /var
/dev/sdb4 montado en /var/crash
><fs> cat /etc/sysconfig/network >>> En RHEL6.8, el hostname se almacena aquí
><fs> cat /etc/sysconfig/network-scripts/ifcfg-eth0 >>> R1
><fs> cat /etc/sysconfig/network-scripts/ifcfg-eth1 >>> R2
><fs> cat /etc/sysconfig/network-scripts/ifcfg-eth2 >>> Auxiliar/Gestión/CAU
><fs> cat /etc/sysconfig/network-scripts/ifcfg-eth3 >>> Interna
### Resto de ficheros SACTA a editar (ETIQUETA.TXT, DEPENDENCIA.TXT, /etc/hosts, ...) ###
><fs> exit >>> IMPORTANTE, si no cerramos bien el intérprete pueden surgir
```

## Propagación (Adaptación en Frío): PASO 3 - Arranque

```
# virsh start POS.rhel6.8 --console
>>>> A partir de aquí, veremos la consola de la MV. // Para salir, pulsar Ctrl+5
User: root
Password: focus
[root@pos-base /]# ip a
```

## Desfragmentación del Disco Virtual

```
_____ 1.- Rellenar con '0' el disco duro
virsh start POS.rhel6.8 --console
dd if=/dev/zero of=/sacta_home/ficherazo bs=20M
rm /sacta_home/ficherazo

_____ 2- Compactar QCOW
// Compactado
virsh shutdown POS.rhel6.8
mv /kvm/vhd/POS.rhel6.8.qcow2 /kvm/vhd/POS.rhel6.8.qcow2.LARGE
qemu-img convert /kvm/vhd/POS.rhel6.8.qcow2.LARGE \
-O qcow2 /kvm/vhd/POS.rhel6.8.qcow2

// Comprobaciones
ll /kvm/vhd/
virsh start POS.rhel6.8

// Eliminar antiguo disco duro
rm -f /kvm/vhd/POS.rhel6.8.qcow2.LARGE
rm -f /kvm/vhd/POS.rhel6.8.qcow2.old
```

