



POLLUX
We keep it simple.

Solicitante:

I.T.S. – Instituto Tecnológico Superior Arias - Balparda

Nombre de Fantasía del Proyecto: Pollux

Grupo de Clase: 3°IC

Turno: Matutino

Materia: Sistemas Operativos III

Nombre de los Integrantes del Grupo: Mathias Huque

Wilson Antognazza

Santiago Maciel

Bruno Obispo

Fecha de entrega: 24/09/2021

Instituto Tecnológico Superior F. Arias – L. Balparda

Gral. Flores 3591 esq. Bvar. José Batlle y Ordoñez - Montevideo

Objetivo

El objetivo de esta materia es capacitarnos en la administración de Sistemas Operativos, ya sea en gestión de usuarios como en administración de respaldos de un sistema.

Alcance

A través de esta carpeta se nos permitirá ampliar nuestros conocimientos para argumentar la elección de distintos sistemas operativos ya sea para terminales como para servidores, también nos otorgará noción de los roles empleados para la gestión de una empresa.

Índice

Contenido

| | |
|---|-----------|
| ÍNDICE | 3 |
| 1. Configuraciones de las cuentas y su perfilamiento..... | 4 |
| 1.1. A NIVEL LOCAL EN LA TERMINAL DONDE INSTALARÁN LA APLICACIÓN. | 4 |
| 1.2. CONEXIONES DE LA APLICACIÓN | 4 |
| 1.3. CUENTAS DE LA APLICACIÓN | 5 |
| 1.4. CONFIGURACIÓN DEL SERVICIO SSH | 6 |
| 2. OTROS PROGRAMAS DE LA PLATAFORMA..... | 8 |
| 2.1 SERVIDOR DE BASE DE DATOS..... | 8 |
| 2.2 SOFTWARE DE MONITOREO | 10 |
| 2.3 ANTIVIRUS..... | 21 |
| 3. Respaldo, recuperación de datos, aplicaciones de base y de usuarios.... | 22 |
| 3.1 BACKUP DE DATOS | 23 |
| 3.2 BACKUP DE SISTEMAS..... | 24 |
| 3.3 RESTORE DE DATOS | 25 |
| 3.4 RESTORE DE APLICACIONES | 25 |
| 4. GLOSARIO | 26 |
| HOJA TESTIGO..... | 27 |

1. Configuraciones de las cuentas y su perfilamiento.

1.1. A nivel local en la terminal donde instalarán la aplicación.

La aplicación se instalará en los equipos correspondientes a los gerentes y administrativos del spa.

Se recomendarán 2 modelos de redes distintas, una que no requiere modificaciones a nivel arquitectónico (a este le llamaremos “red Básica”) y otra que se recomendará por parte de la empresa modificaciones a la estructura física del spa (llamaremos a este modelo de red “red Recomendada”, esto contando con la asesoría de un arquitecto.

La red Básica permitirá la instalación de una sola Terminal (PC) para el Gerente lo cual su IP será 172.20.0.42\30 sin embargo se pueden incluir muchos más gerentes si estos se conectaran mediante WIFI las IP de estos dispositivos estaría definida ente la 172.20.6.2 a la 172.20.7.254, y un total de cuatro Terminales (PC) para los Administrativos su rango de IP será de 172.20.2.2 a 172.20.2.5, se pueden incluir más administrativos si estos se conectaran mediante WIFI la IP de estos administrativos estaría comprendida en el rengó de 17.20.8.2 a 172.20.9.254.

La inicialmente por parte de la empresa se instalará en el equipo del gerente y en los cuatro administrativos antes mencionados (estos equipos estarán conectados a la red mediante cableado)

1.2. Conexiones de la aplicación

Se establecerán 3 conexiones correspondientes al servidor de datos lo cuales son las siguientes:

1. Gerente.

Las Personas que se conecten mediante este usuario corresponderán a los Gerentes del spa. Este perfil podrá insertar valores en la base de datos de todo tipo, ya se ingresar nuevas reservas, realizar alta de usuarios de la aplicación (tabla “Usuarios”), realiza alta de clientes

(Tabla “Cliente”), modificaciones de los mismos, modificación y alta de traslados, podrá realizar consultas de cualquier índole a la base de datos, modificar precios y agregar servicios lo cual se almacenan en la tabla “Servicio”.

2. Administrador

Esta conexión corresponde a los usuarios administrativos de la aplicación que serán los mismos del cliente. Esta conexión contará con más restricciones debidas a que se maneja una jerarquía en la empresa del cliente, por lo cual este usuario solo podrá ingresar reservas, realizar consultas de cualquier tipo. Puede modificar la información del cliente y sus reservas. También puede dar de baja reservas y clientes. Mediante esta conexión no se podrá realizar modificaciones en la tabla correspondiente a los servicios, realizar altas de usuarios.

3. Gestor de Servidor.

Este usuario realizara los Backus correspondientes a la base de datos y realizara tareas de gestión de almacenamiento. Este usuario podrá virtualizar la interfaz del servidor a través de un hipervisor.

Las conexiones “Administrador” y “Gerente” se conectarán al servidor de base de datos de manera indirecta mediante la aplicación solicitada por el cliente, El usuario “Gestor de Servidor” se virtualizara en el servidor mediante un hipervisor respaldado por protocolos de seguridad como SSH. Se basará en el uso de NVD para identificar las vulnerabilidades en el servidor virtual y según el sistema de puntuación de vulnerabilidad CVSS se elaborará un informe de dichos riesgos utilizando el estándar CVE para asignar identificadores a los mismos

1.3. Cuentas de la aplicación

Existirán 2 tipos de cuentas de en la aplicación, gerente y administrativo, al inicio del programa estos usuarios deberán ingresar su cédula como nombre de usuario y su contraseña, estos usuarios serán únicos para cada persona, una vez ingresados estos valores se intentará establecer la conexión con la base de datos, si dicha conexión falla es debido a que el usuario es incorrecto o no ha sido creado, posteriormente se le pedirá a un gerente que verifique si dicho usuario fue creado si así fuera y el administrativo no recordase la password el gerente eliminara dicho usuario para que sea creado nuevamente; Si la conexión es exitosa se verificará que el usuario está activo, para ello se realizará una consulta a las base de datos precisamente a la tabla “Usuario” mediante un usuario llamado “VerificacionUsuario” lo cual sólo tendrá permiso para realizar “select” a dicha tabla, está contendrá los nombres de los usuarios, su estado (“Activo” o “Inactivo”) y el tipo (“Administrativo” o “Gerente”), si el estado de este usuario es “Inactivo” entonces se le negara el ingreso al sistema y se le pedirá que se contacte con un Gerente, si el estado es “Activo” la aplicación ingresa a la interfaz correspondientes a su tipo de usuario.

1.4. Configuración del servicio SSH

Que es el servicio SSH?

SSH es un protocolo que facilita las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor y que permite a los usuarios conectarse a un host remotamente. A diferencia de otros protocolos de comunicación remota tales como FTP o Telnet, SSH encripta la sesión de conexión, haciendo imposible que alguien pueda obtener contraseñas no encriptadas.

Instalación del servicio SSH

Para instalar SSH en un equipo es tan sencillo como entrar en la terminal del sistema operativo y ejecutar el siguiente comando “#sudo yum install openssh-server”, si ya tenemos instalada la última versión de SSH el comando antes mencionado no es necesario ejecutarlo, si no sabemos si ya tenemos SSH instalado es mejor ejecutar dicho comando por precaución.

Cambiar puerto de acceso

La primera es cambiar el puerto por el cual escucha el servicio ssh, el puerto estándar es el TCP 22 y por lo general cualquier persona que intente tener acceso no autorizado a tu servidor va a probar este puerto, por lo que es importante que lo configures a que escuche por otro puerto por ejemplo por el TCP 23022 o el que tu prefieras, recordar que al cambiar el puerto se debe especificar en la conexión al servidor.

Edita el archivo /etc/ssh/sshd_config (#Vi /etc/ssh/sshd_config) y cambia lo siguiente:

```
#Port 22 ; Cambiar a #Port 23033
```

Otro punto importante es asegurarte que se utilice únicamente la versión 2 del protocolo con la siguiente línea:

```
#Protocol 2,1 ; Cambiar a #Protocol 2
```

Debes evitar que se pueda ingresar al sistema con el usuario root, esto al igual que el puerto 22 es conocido por quienes intenten acceder al sistema y el nombre de usuario (conocido) y el más importante a probar es root, ya que con este usuario se puede tener total control sobre el servidor. Para evitar eso niega el acceso mediante ssh como el usuario root.

```
#PermitRootLogion yes ; Cambiar a #PermitRootLogion no
```

Para establecer si se permite o no la ejecución remota de aplicaciones gráficas. Si se va a acceder hacia el servidor desde red local, este parámetro puede quedarse

con el valor yes. Si se va a permitir el acceso hacia el servidor desde redes públicas, resultará prudente utilizar este parámetro con el valor no.

```
#X11Forwarding yes
```

Define una interfaz de red

Muchos servidores tienen dos o más interfaces de red y puedes limitar que ssh escuche por una interfaz de red ssh así limitar el acceso a tu servidor. Para hacer esto solo tienes que indicar la siguiente directiva:

```
#ListenAddress [IP Servidor]
```

Reiniciar ssh y aplicar los cambios

Para que los cambios que sean realizados se apliquen, debes reiniciar el servicio sshd.

```
sudo systemctl reload sshd
```

Hacemos un backup del archivo original.

```
#sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.old
```

Para que el servidor OpenSSH inicie de forma automática cuando se encienda el Sistema Operativo CentOS 7.

```
#sudo chkconfig sshd on
```

Crear usuario para SSH

Para conectarnos de forma remota vamos a crear un nuevo usuario.

```
#sudo adduser userSSH  
#sudo passwd userSSH
```

Configurar el Firewall

Abrimos el puerto que cambiamos.

```
#sudo firewall-cmd --permanent --add-port=23033/tcp
```

Y reiniciamos el Firewall.

```
#sudo firewall-cmd --reload
```

Probar conexión SSH

Desde otra maquina ejecutamos este comando para ingresar remotamente con el nuevo usuario, donde 192.168.1.20 es la IP del servidor OpenSSH.

```
ssh -p 23033 userSSH@192.168.1.20
```

2. Otros programas de la plataforma

2.1 Servidor de base de datos

Teoría:

El motor del servidor de base de datos que se utilizara corresponde a MariaDB con el correspondiente Workbench MySql en su versión 6.3; Ambos servicios son de libre distribución por lo que su licenciamiento es gratuito, aunque no cuentan con un soporte a los usuarios del mismo por parte de la empresa; La instalación del software de base de datos será realizado por nuestra empresa al igual que la implementación del Schema correspondiente y las configuraciones necesarias para que el servidor quede instalado y totalmente funcional e operativo. Aunque las empresas distribuidoras del Workbench y el motor de Base de Datos no brinde un soporte oficial, nuestra empresa brinda un soporte a la base de datos de nuestros clientes

Instalar e iniciar MariaDB

Instalar servidor MariaDB:

```
#sudo yum install mariadb-server
```

Configurar MariaDB para que arranque al iniciar el sistema e inicie el servicio:

```
#sudo systemctl enable mariadb  
#sudo systemctl start mariadb
```

Instalar seguridad para servidor MariaBD:

Para instalar el añadido de seguridad ejecutar el comando.

```
#sudo mysql_secure_installation
```

Aun no se ha establecido una contraseña para el usuario *root*, así que presionaremos *Enter* en el siguiente mensaje:

```
#Enter current password for root (enter for none):
```

Luego de presionar *Enter* le pedirá una Contraseña para el usuario *Root*.

Inicio de sesión con usuario *root*:

```
#mysql -u root -p
```

Cuando se le solicite, ingrese la contraseña *root* que asignó al momento de ejecutar el script

```
mysql_secure_installation.
```

Si el inicio de sesión es correcto aparecerá un encabezado de bienvenida. Por cuestiones de seguridad no se otorgará al cliente la clave del usuario *Root*.

Crear nuevos usuarios MariaBD

'usuarioprueba' corresponde al nombre de usuario *localhost* es la IP de la máquina en la que se está creando el usuario, *SuContraseña* corresponde a la contraseña del usuario

```
>create user 'usuarioprueba'@'localhost' identified by 'SuContraseña';
```

Crear Base de Datos y asignar Usuario

```
>create database bdprueba;
```

```
>grant all on bdprueba.* to 'usuarioprueba' identified by 'SuContraseña';
```

De esta manera creamos la base de datos “bdprueba” y si el usuario “usuarioprueba” no está creado, lo crea y asigna la contraseña “SuContraseña”

Para finalizar salimos de MariaDB

```
>exit
```

Crear una tabla de muestra

Vuelva a ingresar a MariaDB usando ahora el usuario `usuarioprueba`:

```
#mysql -u usuarioprueba -p
```

Crearemos una tabla de muestra llamada `clientes`. Esto crea una tabla con un campo ID de tipo INT (número entero que va incrementando a medida que se añaden nuevos registros) usado como clave primaria, así como dos campos adicionales para almacenar el nombre del cliente:

```
>use bdprueba;
```

```
>create table clientes (id_cliente INT NOT NULL AUTO_INCREMENT PRIMARY  
KEY, nombre TEXT, apellido TEXT);
```

2.2 Software de Monitoreo

Zabbix

Zabbix es una solución de monitoreo de red altamente integrada que puede proporcionar una solución de monitoreo distribuido de código abierto a nivel empresarial. El software se puede descargar y usar libremente.

Instalación:

Se crea un directorio de respaldo y descargue la aplicación Zabbix

```
#wgethttp://sourceforge.net/projects/zabbix/files/ZABBIX%20Latest%20Stable/  
2.4.5/zabbix-2.4.5.tar.gz
```

Se extrae el paquete de la aplicación zabbix aquí

```
#ls
```

```
zabbix-2.4.5.tar.gz
```

```
#tar zxvf zabbix-2.4.5.tar.gz
```

Antes de comenzar la configuración de Zabbix, es necesario crear un usuario de Zabbix.

```
#useradd zabbix
```

Una vez que el usuario de zabbix haya terminado, se debe crear una nueva base de datos para el servidor de zabbix. Para hacerlo, realizaremos los siguientes pasos.

Inicie sesión en MySQL MariaDB

```
#mysql -u root -p
```

Crear nueva base de datos

```
> create database zabbix;
```

Otorgue todos los privilegios al usuario de zabbix en la base de datos de zabbix

```
> grant all privileges on zabbix.* to 'zabbix'@'localhost' identified by 'zabbix123'  
with grant option;
```

Ahora importe el esquema y los datos iniciales.

```
#mysql -u zabbix -p zabbix < /backup/zabbix-2.4.5/database/mysql/schema.sql
```

```
#mysql -u zabbix -p zabbix < /backup/zabbix-2.4.5/database/mysql/images.sql
```

```
#mysql -u zabbix -p zabbix < /backup/zabbix-2.4.5/database/mysql/data.sql
```

Inicie Zabbix Server y la instalación de su agente

Antes de ejecutar el script de instalación, hay que asegurarse de que todas las extensiones php requeridas y otras bibliotecas de soporte, incluidas mysql-devel php-mysql y net-snmp, estén instaladas sin que el proceso de instalación no esté completo.

```
#!/configure --enable-server --enable-agent --with-mysql --with-net-snmp
```

Ahora ejecuta make para instalar todo

```
#make install
```

Instalación de front-end usando PHP

Ahora hay que crear el directorio raíz del documento de Apache para luego mover todos los archivos php frontend en él.

```
#mkdir /var/www/html/zabbix
```

```
#cp -r frontends/php/* /var/www/html/zabbix/
```

Se configuran los parámetros básicos en el archivo de configuración

PHP php.ini

```
#vim /etc/php.ini
```

Tenemos que reiniciar el servicio Apache para que surta efecto después de realizar cambios en el archivo php.ini.

```
# systemctl restart httpd.service
```

En su navegador, abra la URL de Zabbix según la IP o el nombre de su servidor.

<http://172.20.3.174/zabbix>

Debería ver la primera pantalla del asistente de instalación de frontend.



Haga clic en Next

ZABBIX

2. Check of pre-requisites

1. Welcome
2. Check of pre-requisites
3. Configure DB connection
4. Zabbix server details
5. Pre-Installation summary
6. Install

| | Current value | Required | |
|--------------------------------|---------------|----------|----|
| PHP version | 5.4.16 | 5.3.0 | OK |
| PHP option memory_limit | 128M | 128M | OK |
| PHP option post_max_size | 16M | 16M | OK |
| PHP option upload_max_filesize | 8M | 2M | OK |
| PHP option max_execution_time | 300 | 300 | OK |
| PHP option max_input_time | 300 | 300 | OK |
| PHP time zone | Europe/London | | OK |
| PHP databases support | SQLite3 | | OK |
| PHP bcmath | on | | OK |
| PHP mbstring | on | | OK |
| PHP mbstring.func_overload | off | off | OK |
| PHP sockets | on | | OK |
| PHP gd | 2.1.0 | 2.0 | OK |
| PHP gd PNG support | on | | OK |

OK

www.zabbix.com
 Licensed under [GPL v2](#)

Cancel

« Previous

Next »

Una vez que los parámetros de php están configurados en la pantalla, todos los parámetros deberían estar bien. Haga clic en Next



ZABBIX

3. Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database.

Press "Test connection" button when done.

| | |
|---------------|------------------------|
| Database type | MySQL |
| Database host | localhost |
| Database port | 0 0 - use default port |
| Database name | zabbix |
| User | zabbix |
| Password | ***** |

OK

Test connection

Cancel

« Previous Next »

www.zabbix.com
Licensed under [GPL v2](#)

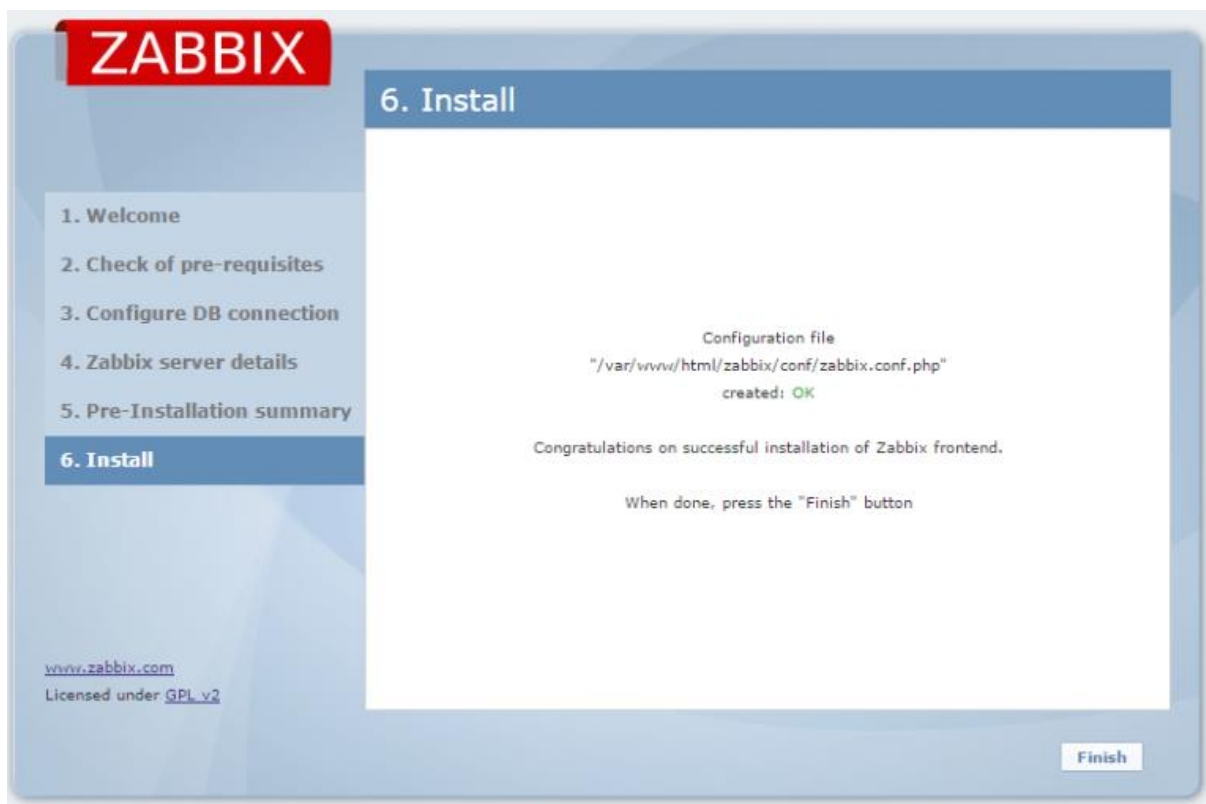
Ingresamos los detalles para la conexión a la base de datos. La base de datos Zabbix ya debe estar creada.
Luego ingrese los detalles de los servidores Zabbix y haga clic en Probar conexión para confirmar su conectividad con la base de datos.



Descargamos el archivo de configuración y se coloca en el siguiente directorio conf.
"/var/www/html/zabbix/conf/zabbix.conf.php"



Después de cargar el archivo conf en la carpeta de destino mencionada, Hacemos clic en “Retry” y en “Finish” después de que aparezca el estado OK.



La interfaz de Zabbix ahora está lista para acceder con el nombre de usuario predeterminado “Administración” y contraseña “zabbix”.



Configuración del servidor Zabbix

Ahora nos dirigimos al archivo de configuración del servidor zabbix para configurarlo y poder comenzar a monitorear hosts. abrimos el archivo de configuración:

```
#vim /usr/local/etc/zabbix_server.conf
```

```
SourceIP= 127.0.0.1
```

```
LogFile=/tmp/zabbix_server.log
```

```
DBName=zabbix
```

```
DBUser=zabbix
```

```
DBPassword=*****
```

Configuración del agente Zabbix

Como ya instalamos el agente zabbix ahora solo se necesita configurar sus parámetros para que se comuniquen con el servidor. Así que abrimos el archivo de configuración del agente zabbix y configuramos los parámetros.

```
#vim /usr/local/etc/zabbix_agentd.conf
```

```
SourceIP=172.20.3.174
```

```
EnableRemoteCommands=1
```

```
Server=127.0.0.1
```

```
ServerActive=127.0.0.1
```

Hostname=Zabbix server

Timeout=30

Iniciar los servicios de Zabbix

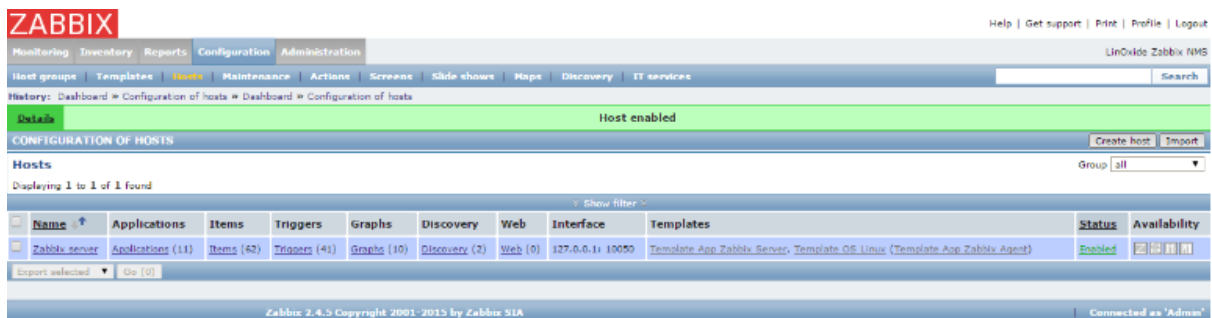
Después de realizar cambios en el servidor zabbix y sus configuraciones de agente, inicie el servidor zabbix y los servicios del agente zabbix como archivos.

```
[root@el_wili]#zabbix_server
```

```
[root@el_wili]#zabbix_agentd
```

Habilitamos el primer

Señalemos a zabbix Configuración y luego Hospedadores para permitir que el servidor zabbix comience su monitoreo



ZABBIX

Help | Get support | Print | Profile | Logout

Monitoring | Inventory | Reports | **Configuration** | Administration

Host groups | Templates | **Hosts** | Maintenance | Actions | Screens | Slide shows | Maps | Discovery | IT services

History: Dashboard » Configuration of hosts » Dashboard » Configuration of hosts

Host enabled

CONFIGURATION OF HOSTS

Hosts

Displaying 1 to 1 of 1 found

| Name | Applications | Items | Triggers | Graphs | Discovery | Web | Interface | Templates | Status | Availability |
|---------------|-------------------|------------|---------------|-------------|---------------|---------|-----------------|---|---------|--------------|
| Zabbix server | Applications (11) | Items (62) | Triggers (41) | Graphs (10) | Discovery (2) | Web (0) | 127.0.0.1:10050 | Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent) | Enabled | |

Export selected | Go [0]

Zabbix 2.4.5 Copyright 2001-2015 by Zabbix SIA

Connected as 'Admin'

2.3 Antivirus

ClamAV

ClamAV es una de las mejores soluciones de seguridad de Linux para deshacernos de numerosas amenazas de malware. Es fiable, fácil de configurar, gratuito pero no cuenta con representante oficial en Uruguay.

ClamAV se instalará en la terminal (PC) del “Gestor de Servido”

Instalar ClamAV en CentOS 7

Como Clam no incluye repositorios de software de CentOS por defecto, se debe agregar el repositorio adicional ejecutando los comandos:

```
#sudo yum -y install epel-release  
#sudo yum clean all
```

Ahora es el momento de instalar ClamAV en CentOS 7. Sólo ejecuta el siguiente comando:

```
#yum -y install clamav-server clamav-data clamav-update clamav-  
filesystem clamav clamav-scanner-systemd clamav-devel clamav-lib clamav-  
server-systemd
```

Se acaba de instalar ClamAV, pero todavía se necesitan hacer algunas configuraciones adicionales para que funcione correctamente.

Configurar SELinux

SELinux es una medida de seguridad destinada a proteger los cambios en algunos archivos. Se requiere una configuración adicional si quieres utilizar ClamAV con el módulo SELinux kernel activado. De lo contrario, Clam no podrá leer una parte de los archivos.

Para configurar SELinux, ejecutan los siguientes comandos:

```
#sudo setsebool -P antivirus_can_scan_system 1  
#sudo setsebool -P clamd_use_jit 1
```

Se verifican los cambios los cambios:

```
#getsebool -a | grep antivirus
```

Los resultados esperados:

```
antivirus_can_scan_system --&gt; on  
antivirus_use_jit --&gt; off
```

Una vez que ClamAV esté listo para usarse junto con SELinux, es hora de configurar el antivirus.

Configurar ClamAV

Antes de habilitar la configuración de ClamAV, se debe eliminar la cadena de caracteres Example del archivo de configuración:

```
#sed -i -e "s/^Example/#Example/" /etc/clamd.d/scan.conf
```

A continuación, se debe especificar el tipo de servidor. Se debe editar el archivo de configuración:

```
#sudo nano /etc/clamd.d/scan.conf
```

Desplázate hasta esta línea:

```
#LocalSocket /var/run/clamd.scan/clamd.sock
```

Se debe quitar esta símbolo “#” “y guarda los cambios.

A continuación, se debe elimina la cadena de caracteres Example del archivo de configuración del motor de actualización freshclam de ClamAV:

```
#sed -i -e "s/^Example/#Example/" /etc/freshclam.conf
```

Una vez hecho esto, de debe ejecutar la actualización de la base de datos de definición de virus:

```
#sudo freshclam
```

Por último, inicia el servicio Clamd y ejecútalo en inicio:

```
#systemctl start clamd@scan  
#systemctl enable clamd@scan
```

3. Respaldo, recuperación de datos, aplicaciones de base y de usuarios

3.1 Backup de Datos

Procedimiento backup de datos DB:

Se realizarán dos backups individuales para el contenido de cada tabla, esto con el fin de que el restore sea más preciso y rápido en caso de alguna falla o pérdida de datos. Estos backups se realizan diariamente (exceptuando los días en los que el spa no está abierto) al término de la jornada laboral

Implementación:

Con el siguiente comando realizaremos el backups del contenido de la DB:

```
$ ssh -p 23033 NombreUsuiroSSH@IPserver "mysqldump -v --opt --no-create-  
info --skip-triggers --default-character-set=utf8 -u NombreUsuarioMySQL --  
password=PasswordUsuarioMySQL NombreDeDB " >  
RespaldoDatosDeDB_`date +%Y%m%d_%H%M%S`.sql
```

Esto nos realizará un backup del contenido de la DB, este respaldo será guardado en la terminal en donde se ejecutó dicho comando, el respaldo tendrá el nombre de "RespaldoDatosDeDB" y la fecha exacta en el cual fue realizada, este archivo será redireccionado a Una carpeta Backup/BackupDatosDB.

Este comando estará dentro de un Script que mediante el comando "crontab -e" se ejecutara todos los días como se mencionó anteriormente.

Procedimiento backup de Estructura de DB:

Se realizarán dos backups únicos para la estructura de la base de datos ya que esta no sufrirá modificaciones ya que por seguridad el cliente no tendrá permisos para realizar este tipo de modificaciones.

Implementación:

Con el siguiente comando realizaremos el backups de la estructura de la DB:

```
$ ssh -p 23033 NombreUsuarioSSH@IPserver "mysqldump -v --opt --no-data --  
default-character-set=utf8 -u NombreUsuarioMySQL --  
password=PasswordUsuarioMySQL NombreDeDB " >  
RespaldoEstructuraDeDB_`date +%Y%m%d_%H%M%S`.sql
```

Esto nos realizará un backup de la estructura de la DB, este respaldo sera guardado en la terminal en donde se ejecutó dicho comando, el respaldo tendrá el nombre de "RespaldoEstructuraDeDB" y la fecha exacta en el cual fue realizada, este archivo sera redireccionado a Una carpeta Backup/BackupEstructuraDB.

Este comando estará dentro de un Script que mediante el comando "crontab -e" se ejecutará una vez al mes.

3.2 Backup de Sistemas

Se utilizará el comando scp para realizar los respaldos de los directorios /home de los usuarios, lo cual será guardado en la carpeta "BackupSistemas" correspondiente al pc del Gestor de servidor, también se establece que el ancho de banda de la transferencia será de un máximo de 7 MB/s para evitar sobreestimar al servidor.

```
scp -l 7000 -r usuario@dominio.com:/ /BackupSistemas
```

También se puede ejecutar este comando desde otro equipo si se utiliza el comando ssh para iniciar una conexión con el tercer equipo, por ende este respaldo quedaría en un tercer equipo.

Estos backup se realizaran mediante el comando "crontab -e" una vez al mes.

3.3 Restore de Datos

Se utilizará el siguiente comando para realizar los restore de datos:

```
$ ssh NombreUsuariSSH@IPserver "mysql -u NombreUsuarioMySQL --  
password=PasswordUsuarioMySQL NombreDB" <  
NombreDeArchivoDeRespaldo.sql
```

Este comando se ejecutará cuando sea necesario realizar una restauración de los datos de la DB o la estructura de la misma.

Recordar que para hacer una correcta restauración de los mismos es necesario asegurarse que las tablas de la base de datos están totalmente vacías, o deben estar dropeadas.

3.4 Restore de Aplicaciones

Se utilizará el comando scp para restaurar los datos anteriormente respaldados

```
scp -l 7000 -r /BackupSistemas usuario@dominio.com:/
```

Se establece que el ancho de banda de la transferencia será de un máximo de 7 MB/s para evitar sobreestimar al servidor.

4. Glosario

Hoja Testigo