ISE
PRACTICA 2

índice

NEWS

WUOLAH

LUN 29 OCT

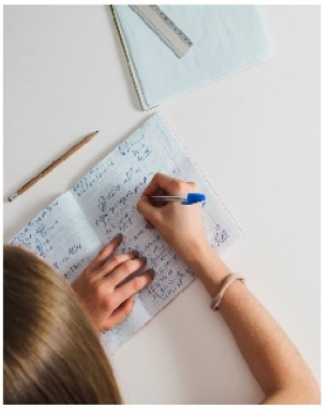
Noticias para el mundo universitario.

nº 1. Semana del 29 al 2

CIEE oferta becas estudiantiles para los programas Intership USA y Work and Travel

FCOM Videcanato de Movilidad y Relaciones Internacionales.

CIEE, organización norteamericana sin ánimo de lucro que desarrolla y administra programas de estudios, voluntariado, prácticas y trabajos de verano en el extranjero; vuelve a ofertar becas para los programas Intership USA y Work and Travel. En la primera se posibilita realizar prácticas profesionales en una empresa o institución de EE.UU. relacionadas con la titulación del estudiante. Y bajo la modalidad más popular, Work and Travel, se ofrecen trabajos no cualificados en parques de atracciones, parques acuáticos, hoteles, restaurantes, etc. De EE.UU durante el verano.
Para más información puedes visitar: cieeseville.com



Expulsados de la universidad varios alumnos por vender un examen

Fuente: ultimahora.es

La Universitat de Barcelona (UB) ha expulsado durante un periodo de dos años a cuatro estudiantes responsables del robo y la comercialización de un examen en julio de 2016, de la asignatura 'Fundamentos de la Fiscalidad' del tercer curso del grado de Administración y Dirección de Empresas (ADE), ha informado el centro en un comunicado este martes.

Ante la gravedad de los hechos, el centro puso el caso en manos de los Mossos d'Esquadra y, dos años después, la UB les ha sancionado con una expulsión temporal de la Facultad de Economía y Empresa, con la voluntad de actuar contra conductas fraudulentas en el ámbito académico que recoge el nuevo código ético.desconocido para el público general.
«Este tipo de conductas ponen de manifiesto, además de una falta absoluta de honradez académica, una actitud de desdén hacia las normas de convivencia y respeto» que deben regir las relaciones en la comunidad universitaria, ha afirmado la UB, que sentenció que no pueden ser toleradas.

WUOLAH GIVEAWAY

Fujifilm instax mini 9.

Fotografía todos tus viajes, tus fiestas, tus festivales...
Ahora no tendrás excusa para tener en fotos instantáneas todos tus recuerdos.

Party&Co Extreme 3.0

Reune a todos tus amigos por que ya tenéis las risas aseguradas... Puedes ganar este Party&Co que seguro será el alma de la fiesta....

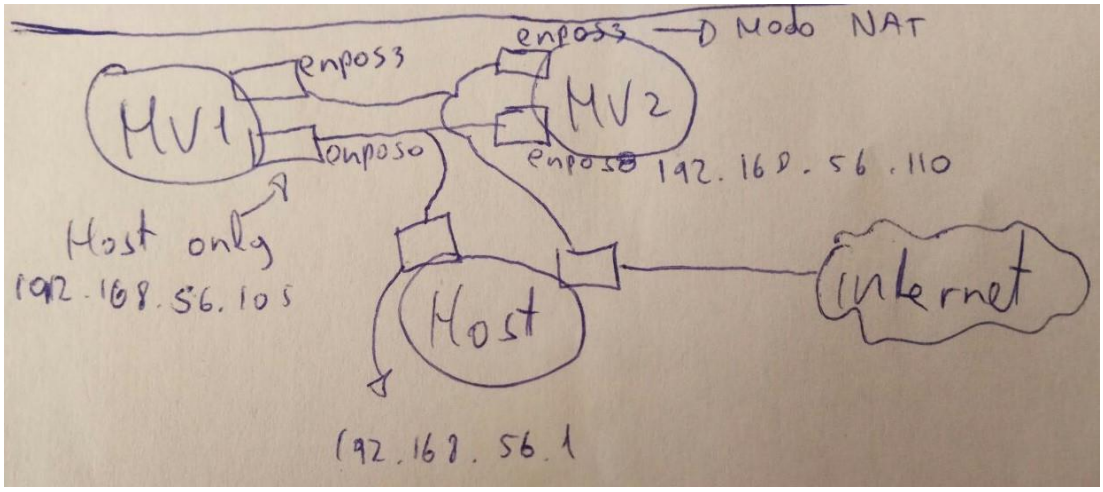
Somos la primera cátedra de eSports en una universidad pública de todo el país.

Fuente: www.diariosur.es

Manuel Fernández Navas es el codirector de la Cátedra Estratégica de eSports de la Universidad de Málaga, que lidera el proyecto de investigación 'Brain Gamer'. Este estudio, que se encuentra en su segunda fase, tiene como objetivo observar el comportamiento psicofisiológico de los jugadores profesionales de videojuegos, los denominados eSports. Para ello colaboran el Ayuntamiento de Málaga, la Universidad de Málaga, la Universidad Internacional de La Rioja (UNIR) y la Sapienza de Roma, además de diferentes empresas como Vodafone Giant y la Liga Survival.

Prerrequisitos:

Para realizar esta práctica, necesitamos que las maquinas tengan esta configuración de red:



Mv1 → Ubuntu

Mv2 → centOS

Como crear la estructura de red:

Para crear una red interna en virtual box debemos ir a archivo → preferencias → red.

Aquí creamos una red con configuración 192.168.56.0/24.

Después de esto, en cada máquina virtual debemos activar la segunda interfaz de red y seleccionarla en modo solo-anfitrión (host-only en inglés).

Esto se consigue en configuración→ red → adaptador 2.

Para modificar la configura de red de las maquinas

UBUNTU

Modificamos el archivo /etc/network/interfaces con: *"sudo nano /etc/network/interfaces"* y añadimos al final

```
iface enp0s8 inet static
address 192.168.56.105
netmask 255.255.255.0
gateway 192.168.56.1
```

Para reiniciar la red:

/etc/init.d/networking restart

CENTOS

Vi /etc/sysconfig/network-scripts/ifcfg-enp0s8 (si no existe copiamos el ifcfg-enp0s3 con el nuevo nombre).

Borramos todas las líneas menos:

type

bootproto

name

device

onboot

Y añadimos

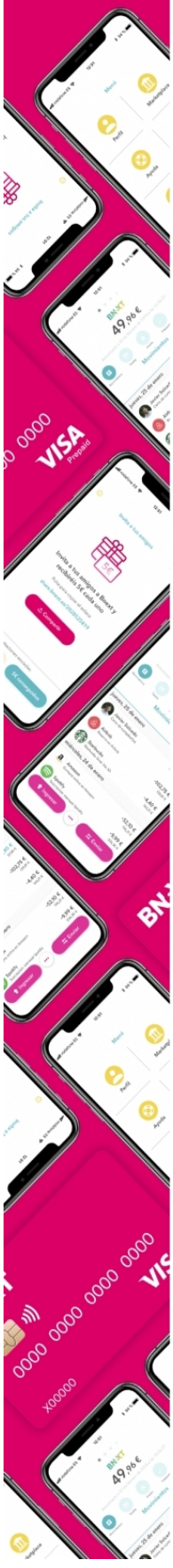
ipaddr=192.168.56.110

además, si el atributo onboot está en OFF, ponerlo en ON (VERIFICAR EN EL FICHERO /etc/sysconfig/network-scripts/ifcfg-enp0s3 QUE ESTE EN ON TAMBIEN ESTE ATRIBUTO).

para reiniciar la red:

Tiramos la interfaz con: ifdown enp0s8

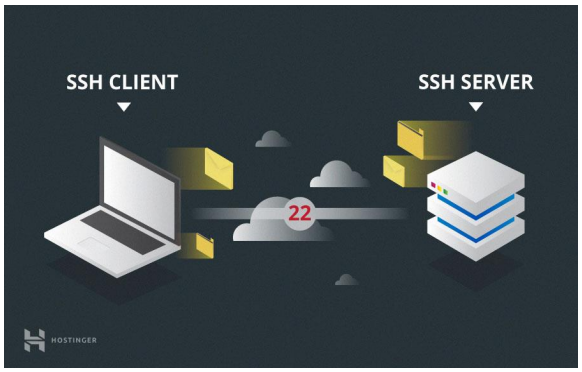
La levantamos con: ifup ifdown enp0s8



Dia 1

Anotaciones

Nuestro servidor será Ubuntu y nuestro cliente será centOS.



En esta sesión instalaremos shh y configuraremos algunos parámetros.

El fichero de configuración de ssh es: /etc/ssh/sshd_config.

Parámetros:

- **port:** <int> identifica el puerto de conexión.
- **PermitRootLogin** <Prohibit-password, no> permite conectarse con el usuario root.
- **passwordAuthentication** <yes,no> permite conectarse con usuario y contraseña.

Comandos

- **ps -xf:** muestra todos los procesos en ejecución.
  - **x:** procesos que no están siendo ejecutados por ninguna terminal (también).
  - **f:** formato "forest" ("bosque") de familias en forma de árbol.
- **ssh <ipservidor> -l <usuario> -v:** conecta con un servidor remoto a través de ssh.
  - **l:** usuario.
  - **v:** verbose (muestra información extra).
- **apt shearch <cadena>:** busca los paquetes que se pueden instalar con apt.
- **apt install <paquete>:** instala el paquete indicado.
- **tasksel:** abre la interfaz de instalación de servicios que lanza Ubuntu durante su instalación.
- **systemctl status ssh.service:** comprueba el estado del servicio ssh.
- **systemctl start ssh.service:** inicia el servicio ssh.
- **systemctl stop ssh.service:** para el servicio ssh.
- **ssh-keygen:** genera un par de claves pública y privada.
- **ssh-copy-id <usuario>@<ipservidor> -p <puerto>:** copia la clave publica en el servidor destino. (necesario usuario y contraseña a no ser que ya haya sido configurado para acceder con llaves previamente).

Instalación y pruebas

En el servidor:

Comprobamos que ssh no está instalado (ni ejecutado en la maquina) con las siguientes ordenes:

Usamos el comando para listar los procesos que están en ejecución: *"Ps -xf".*

Filtramos los procesos ssh con: "*ps -xf |grep ssh".*

Para buscar el nombre de los paquetes shh servidor utilizamos:

Sudo Apt shearch ssh | grep server.

Con este comando instalamos ssh: "*Apt install openssh-server".*

"Sudo tasksel" abre la terminal de instalación de canonical (otra opción para instalar ssh).

"*Vi /etc/ssh/sshd_config".*

Cambiamos el puerto a 22022 (parámetro **P**ort).

Reiniciamos el servicio con: *"systemctl restart ssh.service".*

Comprobamos el estado del servicio con: *"systemctl status ssh.service".*

Comprobamos que la conexión se establece con éxito.

"Ssh <usuario>@localhost -p 22022 -v".

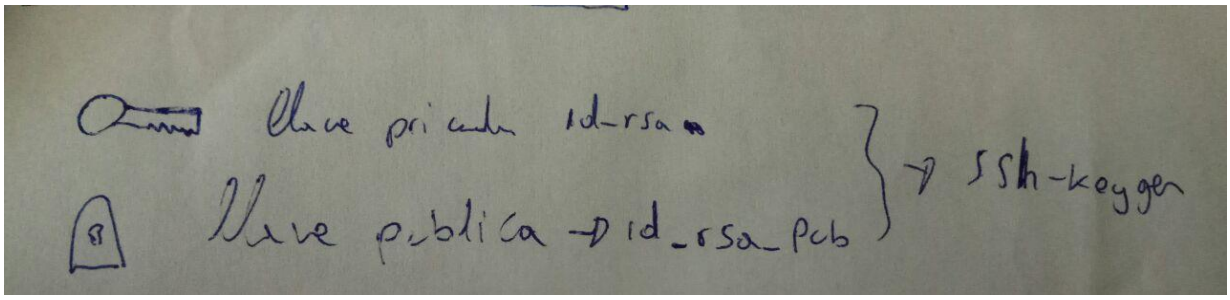
Modificamos el fichero de nuevo para denegarle el acceso a root por motivos de seguridad con: "*Vi /etc/ssh/sshd_config".*

PermitRootLogin Prohibit-password → PermitRootLogin no.

Reiniciamos el servicio con: *"systemctl restart ssh.service".*

Comprobamos el estado del servicio con: *"systemctl status ssh.service".*

En el cliente:



Comprobamos que ahora no nos deja con el usuario root con:

"Ssh 192.168.56.105 -p 22022 -l root -v" (no nos deja).

Generamos las dos llaves (publica y privada) con: *"Ssh-keygen".*

Copiamos la clave publica al servidor con: *"Ssh-copy-id <usuario>@192.168.56.105 -p 22022".*

(pregunta la contraseña del usuario).

Para comprobar que la llave se copió correctamente y que funciona, intentamos conectar usándola con:

"ssh -p 22022 <usuario>@192.168.56.105".

(no debe pedir contraseña).

En servidor:

Modificamos el servicio para que ahora no deje entrar con usuario y contraseña a nadie (solo podrá acceder el que tenga la llave privada).

"Vi /etc/ssh/sshd_config".

passwordAuthentication no (decomentamos y lo ponemos a no).

Reiniciamos el servicio con: *"systemctl restart ssh.service".*

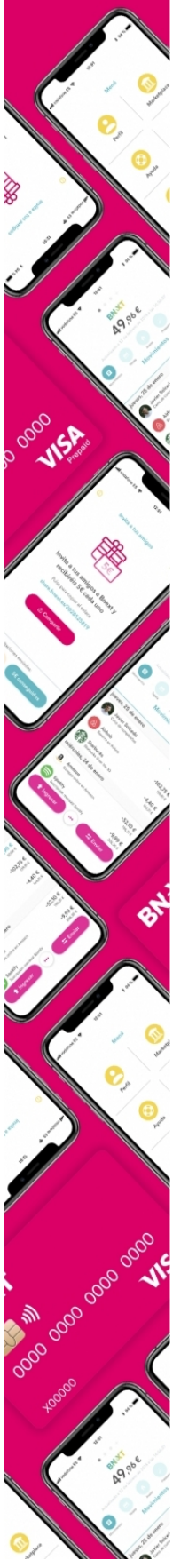
Comprobamos el estado del servicio con: *"systemctl status ssh.service".*

En el cliente:

Para comprobar que esta última modificación se ha hecho efectiva bastaría con intentar acceder con otro usuario que no sea en el que se copió la clave pública y/o desde otra máquina cliente:

"ssh -p 22022 <usuario2>@192.168.56.105".

Una vez aquí, podremos acceder desde la maquina que tenga la clave privada, pero desde ninguna más, aunque conozcamos los usuarios y contraseñas.



Dia 2

anotaciones

En esta sesion vamos a instalar ssh en centOs y vamos a lidiar con una serie de problemas con SElinux y el cortafuegos.

A demás veremos que son, para que se usan y como hacer un buen uso de las copias de seguridad, incluyendo control de versiones (git).

Direfencias entre centOS y ubuntu

	INSTALADO	NOMBRE	ARCHIVO CONF	FIREWALL ACTIVO
UBUNTU	No	Ssh/sshd	Sin comentarios	No
CENTOS	Si	Sshd	Con comentarios	si

Comandos

Comandos de practica:

- Sed
  - -i modifica una línea
  - -e sustituye una cadena por otra
- Journalctl (muestra info de errores del sistema)
  - -x muestra explicación de los errores producidos
  - -f follow, hace un seguimiento de los errores en el tiempo
- Yum provides
- Semanage
  - -l muestra una lista
  - -a añadir puerto nuevo
  - -p puerto
  - -t tipo de puerto
- Ufw status
- Ufw enable
- Ufw allow
- Firewall-cmd
  - --add-port=
  - --permanente
  - –reload

Comados de copias de seguridad:

- Dd (copia a nivel de bytes)
- Cpio (lista los ficheros de un directorio)
- Tar (comprime y descomprime)
  - c crear archivo
  - v vervose
  - z para comprimir
  - f archivo que creamos
  - x extract

- Rsync (sincroniza (dropbox antiguo))
    - -a guarda metadatos, permisos, privilegios…
    - -v vervose
    - -i informe final
    - -z comprime
    - --delete
- Rsnaptshot (crea un pantallazo del sistema)
- Programas de pago:
    - AMANDA
    - Bacula
    - C-Panel
    - Plesk

Comandos de control de versiones:

- git init (inicia el repositorio en la carpeta en la que estamos)
- git log (visualiza el log)
- git status (muestra el estado de git)
- git config (configura el usuario de git)
    - --global user.name <usuario>
    - --global user.email <correo>
- git add <nombre archivo> (añade un archivo al commit)
- commit (envia el commit y lo aplica)
    - -m (mensaje)
    - --ammend (modifica el mensaje)
    - -ammend -a (añade un archivo nuevo)
- Diff (muestra las diferencias entre el work directory y el staging area)
    - --taged (entre staging área y local repository)
    - HEAD (entre el working directory y local repository)

- checkout HEAD <nomb_archivo> (recupera el archivo que hay en el último commit)
- reset <7_primeros_caracteres_del_commit_SHA> (recupera el commit especificado)
- git revert <7_primeros_caracteres_del_commit_SHA > (recupera el commit y crea uno nuevo)
    - -n (evitamos que cree un commit nuevo)
- branch (nos muestra en qué rama estamos (master por defecto))
- checkout <nombre_rama> (para cambiar la rama que usammos)
- branch <nombre_rama> (crea una rama)
- checkout -b <nombre_rama> (crea una rama nueva y se une a ella)
- merge (une ramas)
- git init –bare (crea un repositorio en linea)
- git clone (clonamos el repositorio)
- git add remote (añadimos un origen remoto)
- git push (enviamos los commits al origen remoto)
- git pull (nos traemos los cambios del repositorio y unimos con los del directorio de trabajo actual)

Instalación y pruebas

En centOS:

Otra forma de modificar un archivo de configuración es con el comando sed.

Necesitamos modificar el puerto de ssh para ello usando sed reemplazamos la cadena que determina el puerto en el archivo /etc/ssh/sshd_config con:

'Sed -e "s/#Port 22/Port 22022/" -i /etc/ssh/sshd_config'

Reiniciamos el servicio con:

"Systemctl restart sshd"

"Systemctl status sshd"

Vemos que tiene error y que no se ha levantado el servicio.

Ahora vamos a ver el log de errores a ver que ha pasado con este comando:

"Journalctl -xf"

Vemos que el error se produce porque el cambio de puerto no se puede hacer efectivo.

En el archivo de configuración de ssh sale que debemos notificar a SElinux sobre el cambio de puerto, sino no funcionara.

Para ello instalamos el paquete semanage con:

"Yum install semanage"

Da también error porque le hacen falta paquetes que no tiene. para saber que comando usar:

"Yum provides semanage"

Nos dice que usemos el comando:

"Yum install policycoreutils-python"

Una vez instalado seguimos con lo que estábamos.

Para listar los puertos que están usando el sistema:

"Semanage port -l"

Para buscar el puerto de ssh:

"Semanage port -l | grep ssh"

Una vez vemos que el puerto de ssh es el 22, podemos **añadir otro puerto permitido** con el siguiente comando:

"Semanage port -a -p tcp -t ssh_port_t 22022"

hecho esto ya nos dejara **levantar el servicio ssh** con:

"Systemctl start sshd"

añadimos un usuario con:

"useradd -m usuario"

Le ponemos contraseña con:

"passwd usuario"

Modificación del cortafuegos:

En ubuntu:

Probamos que ssh 192.168.56.101 -p 22022

Pero no funciona porque el firewall corta la conexión.

En Ubuntu para modificar el cortafuegos:

Ufw status

Ufw enable

Ufw allow 22022

En centos:

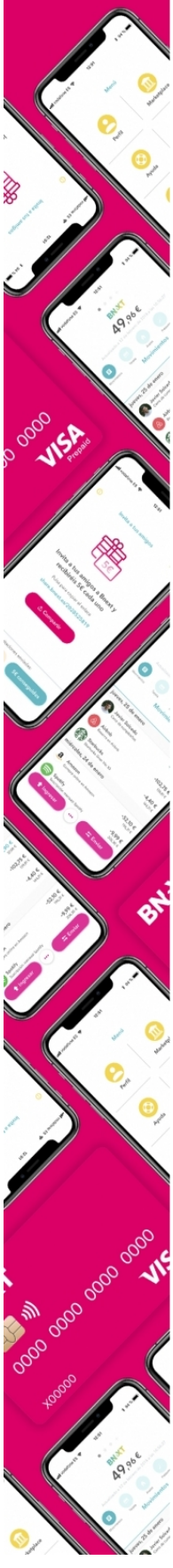
Firewall-cmd --add-port=22022/tcp

Funciona. Si reinicias ya no funciona, hay que ejecutarlo de nuevo

Para hacerlo permanente con:

Firewall-cmd --permanent --add-port=22022/tcp

Para reiniciar el cortafuegos con:



Firewall-cmd –reload

En Ubuntu:

Para comprobar que todo esta bien configurado, conectar a ssh con:

"Ssh 192.168.56.110 -p 22022"

Diapositivas de copias de seguridad y control de versiones

Copias de seguridad

Que diferencia una copia de archivos de una copia de seguridad??

Los metadatos (como la fecha)

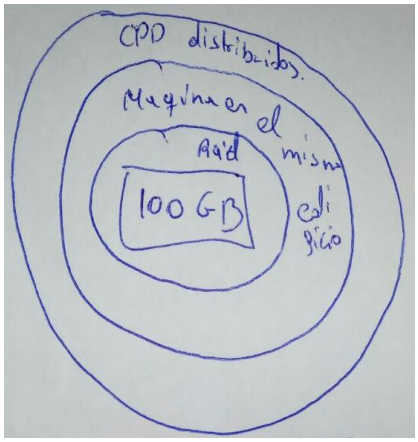
Con estos comandos podemos crear una copia de seguridad añadiendo la fecha al final del nombre del archivo:

dt=`date +%y%m%d`

Cp -a /var/. /varOLD-$dt

Estaría bien cifrarlo y distribuirlo en distintos CPDs

La mejor forma es la seguridad en anillos.



Hay varios tipos de copias de seguridad:

Total: hace copia de todo el sistema

Parcial: hace copia de una parte del sistema

Incremental: compara las diferencias y las guarda con respecto a la última versión.

Al hacer la copia tenemos problemas con el espacio en disco y el tiempo

Con **LVM** evitamos poner en modo monousuario porque tiene **snapshots** (siempre que haya espacio suficiente)

Control de versiones

Hay algunas herramientas que controlan /etc directamente como: /etc/keeper

Utilizan git por debajo.

GIT es un sistema de archivos direccionable de contenido

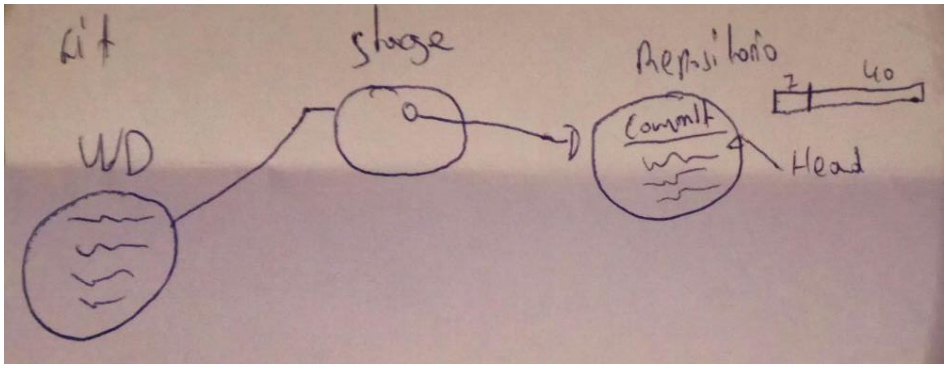
Lo programo Linux tolbar.

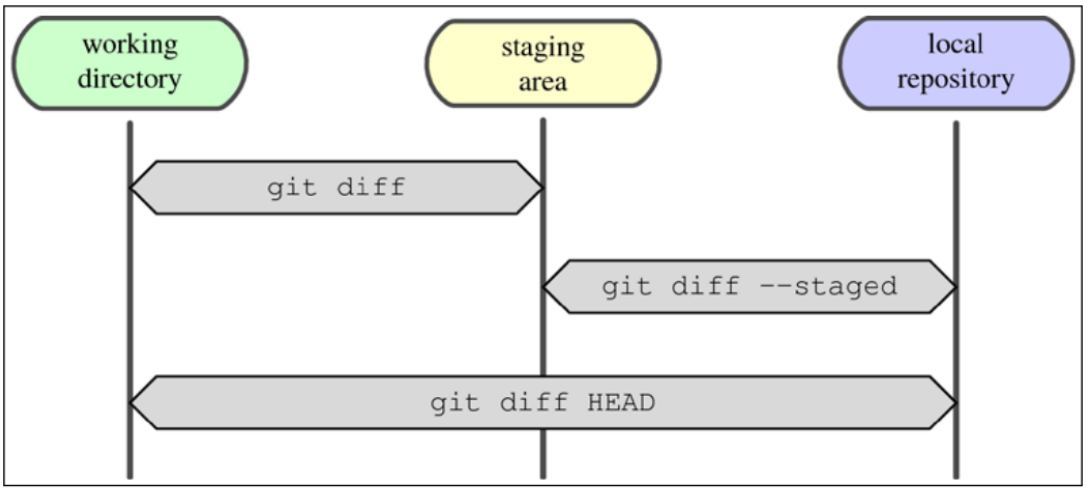
Ventajas (para ISE):

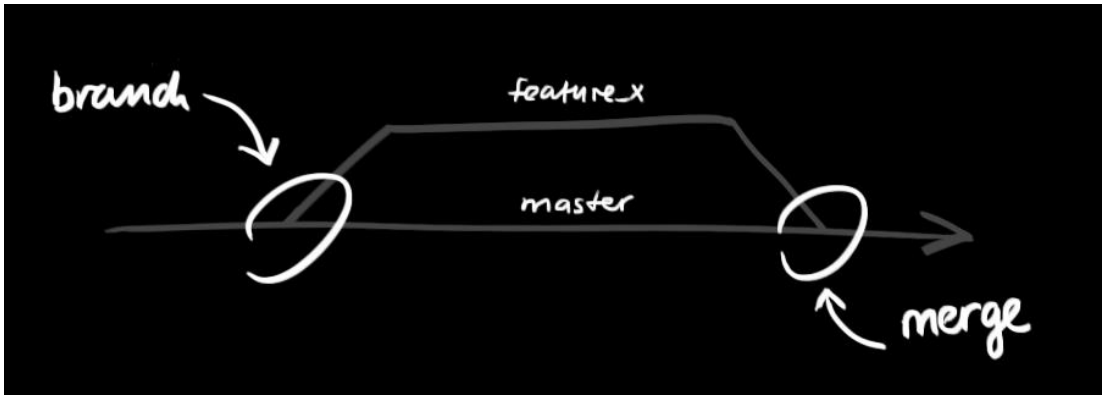
- Ponemos un pie en devops
- Seguimiento de los cambios (y de su autor)
- Permite planificar un entorno de prueba entre varias personas

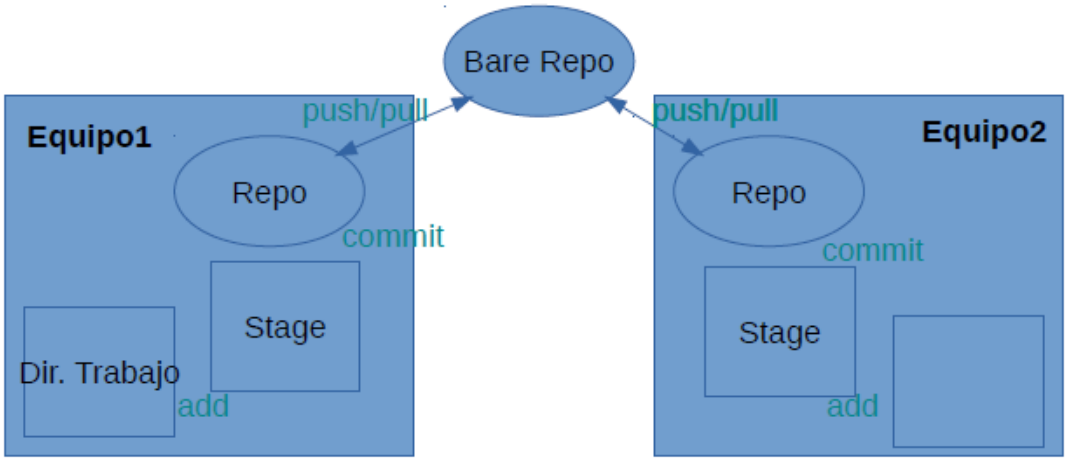
Cómo funciona

- Directorio de trabajo: lo que tenemos en desarrollo
- Stage: Zona donde se registran los cambios. Ahí incluimos los cambios que queremos seguir.
- Commit: Zona donde los cambios se convierten en permanentes (podemos hacer commits en diferentes ramas “branches” y ponerles etiquetas “tags”)
- La evolución del trabajo puede mostrarse como un grafo dirigido acíclico (DAG)









Dia 3

Anotaciones

Información Básica

HTTP:

Hyper Text Transfer Protocol.

Utiliza el puerto 80 por defecto.

Hipertexto:

Documento que puede hacer referencia a otro documento o a una parte de el mismo.

Las peticiones se hacen desde un navegador (o mediante el comando curl).

HTML:

HyperTextMake-upLanguaje.

Incluyó la etiqueta <script> para poder añadir JavaScript a las páginas y hacerlas más dinámicas.

JavaScript:

Lenguaje de programación del lado del cliente.

No es capaz de almacenar información sensible.

Al ejecutarse en el cliente el código fuente está expuesto.

JSON:

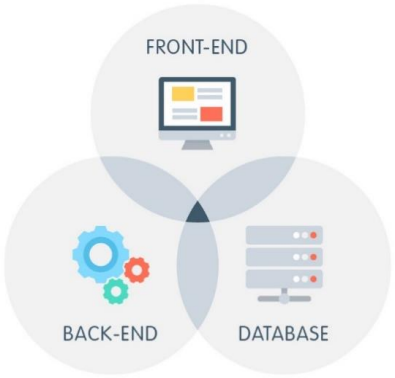
JavaScriptObjetNotation.

Se utiliza como estándar para que distintos lenguajes de programación puedan comunicar estructuras de datos entre sí.

Servidor web:

proceso que escucha el puerto 80 esperando peticiones de archivos y los transfiere a través del protocolo HTTP.

Apareció en el CERN en el año 89 y lo invento Tim Berners-Lee.



Para poder servir una página dinámica y segura son necesarios al menos 4 elementos:

- Servidor web.
- Lenguaje Front-end: lenguaje que se ejecuta en el cliente.
- Lenguaje Back-end: lenguaje que se ejecuta en el servidor.
- Base de datos: ya sea relacional o no, una página dinámica necesita almacenar la información que se va a tratar o a mostrar a sus usuarios.

Algunos servidores web:

- Apache
- Nginx
- IIS
- Ligthttpd

Algunas Base de datos:

- MySQL → mariaDB
- PostgreSQL
- MongoDB → no relaccional (noRDB) = NOSQL.

HDB → hybrid dataBase = es un tipo de bases de datos. Usan arquitectura hibrida entre relacional y no relacional.

Algunos lenguajes servidor (backend):

- PHP
- PYTHON
- PERL
- .NET
- NODE.JS
- GO

Los lenguajes backend tiene ciertas características en común. Estas son sus ventajas:

- no se tiene que compilar provocando menos fallos.
- La portabilidad.
- Librerías fáciles de usar para tratar strings y conectarse a BBDDs.

Pila:

Una pila es un conjunto de elementos software que nos permiten servir documentos manteniendo bajo control la información y la lógica del programa.

Algunas pilas:

- LAMP: LINUX APACHE MYSQL PHP
- NMP: NGINX MYSQL PHP

Framework:

Es una biblioteca de bibliotecas (metaBiblioteca)

Algunos framework pueden hacer de servidor web también, usando lenguajes como: Go, Python o node.js

Dato relevante:

servicio habilitado: cuando reinicio mi maquina system.d va a mirar si iniciarlo automáticamente.

servicio activo: el servicio está actualmente funcionando.

Buscar software malicioso npm

Npm es el gestor de paquetes de node.js

<u>El último ataque conocido:</u>

un hacker ha obtenido acceso a la cuenta npm de un desarrollador e inyectado código malicioso en una popular biblioteca de node.js. El código malicioso robaba las credenciales de los usuarios que lo importaban a sus proyectos.

El paquete se llamaba eslint-scope (versión afectada 3.7.2).

El ataque se llevó a cabo en la madrugada del 11 al 12 de octubre del 2018.

Se cree que el ciberdelincuente usó el token npm recién generado para autenticar e insertar una nueva versión de la biblioteca eslint-scope en el repositorio npm de paquetes JavaScript.

<u>Otros ataques:</u>

El primer incidente ocurrió en agosto del 2017 cuando se detectaron 38 paquetes que robaban variables de entorno

El segundo incidente ocurrió en mayo de 2018 en el que alguien intento esconder una puerta trasera en el paquete getcookies.

El tercero y ultimo es el anteriormente descrito.

Calcular la diferencia de espacio para entre LAMP y servidor en centos

En centOs cuando instalas los paquetes te sale cuando ocupan (52,3MB totales):

Httpd: 10MB

Mariadb-server: 33MB

Php: 9.4MB

En Ubuntu con tasksel:

Ocupa 335MB (comprobado viendo cuando ocupa una instalación limpia y otra con LAMP)

Comandos

- curl <direccion> (hace una petición http a un servidor)
- php -a (abre un intérprete de php en la terminal)
- systemctl status/stop/restart mysql.service (Ubuntu)
- systemctl status/stop/restart apache2.service (Ubuntu)
- mysql -u <usuario> -p (conecta al servidor del localhost)
    - -u (usuario)
    - -p (pide contraseña)
- yum install httpd (instala el servidor http)
- yum install php (instala php)
- Yum install mariadb (instala el cliente mysql)
- yum install mariadb-server (instala el servidor mysql)
- systemctl status/stop/restar/enable httpd (centOs)
- systemctl status/stop/restar/enable mariadb (centOs)
- mysql_secure_installation (configura mysql de modo seguro)
- firewall-cmd --add-port=80/tcp --permanent (abre el puerto 80)
- Firewall-cmd –reload (reinicia el cortafuegos)
- *yum install php-mysql -y (instala la librería para conectar a mysql con php)*
- *getsebool -a | grep httpd (lista las variable de SE asociada a httpd)*
    - -a lista las variables
- *setsebool -P httpd_can_network_connect_db on (permite a httpd conectar a db)*
    - -P (hace el cambio permanente)
- yum install epel-release (instala epel, un conjunto de paquetes para Linux empresarial)
- yum install fail2ban (instala fail2ban)
- systemctl status/stop/restar/enable fail2ban
- fail2ban-client status (muestra las cárceles activas)
- fail2ban-client set sshd unbanip <ip>” (para sacar a alguien de la cárcel)
- fail2ban-client set sshd banip <ip>” (Para meterlo manualmente)
- Yum install tmux (instala tmux)
- Yum install screen (instala screen)
- Screen -list (lista los ficheros colgando de screen)
- Screen -r <número de tarea> (sigue con la tarea salvada)
- Tmux ls (lista los ficheros colgando de tmux)
- tmux attach -t <número de tarea> (sigue con la tarea salvada)

Instalación y pruebas

ubuntu:

Para instalar LAMP usamos el comando:

"tasksel"

seleccionamos solo LAMP server.

Comprobamos que el servicio este activo con:

"Systemctl status apache2.service"

Para ver que funciona correctamente usamos el comando

"curl localhost"

Para ver si php esta activo:

"php -a"

"echo('hola');"

"exit" (para salir)

Para comprobar el estado de mysql:

"Systemctl status mysql.service"

Para asegurarnos 100% nos conectamos con:

"Mysql -u root -p (para que nos pida el password)"

Podemos conectar a mysql

"exit" (para salir)

centOS:

Para instalar apache:

"Yum search httpd"

"Yum install httpd"

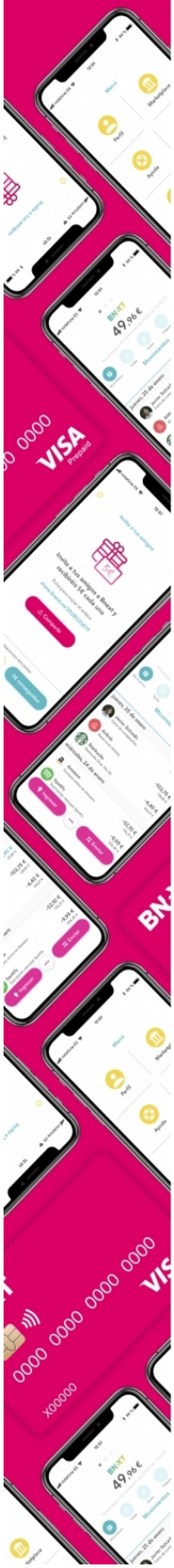
(Versión 2.4.6)

Para que apache se inicie cuando enciende el ordenador:

"Systemctl enable httpd"

Para activarlo ahora:

"Systemctl start httpd"



Para instalar php:

"Yum search php"

"Yum install php"

Para comprobar que php funciona correctamente:

"Php -a"

"echo('hola');"

Para instalar mariadb:

"Yum search mariadb"

"Yum install mariadb" (instala el cliente)

"Yum install mariadb-server" (instala el servidor)

Para que se inicie al reiniciar el ordenador:

"systemctl enable mariadb"

"systemctl start mariadb"

Para comprobar el estado del servidor:

"Systemctl status mariadb"

para conectar al servidor mysql:

"mysql -u root -p" (sin contraseña, al principio)

para hacer segura la instalación de mysql:

"mysql_secure_installation" (pide la contraseña, eliminar usuarios anónimos, desactivar el login remoto de root, eliminar base de datos de test)

intentamos conectar al servidor web de centOs

comprobamos red y vemos que tienen comunicación:

"ping <ipservidor>"

Y hacemos la petición:

"curl <ipservidor>"

El firewall corta el acceso porque el puerto 80 está cerrado

Para solucionar el problema abrimos el puerto con:

"firewall-cmd --add-port=80/tcp –permanente"

Y reiniciamos el cortafuegos para cargar la nueva configuración con:

"firewall-cmd --reload"

Ahora ya si se puede hacer la petición:

"curl <ipservidor>"

¿Se comunican mysql php y apache?

Hagamos un ejemplo para probarlo

Creamos un archivo php en /var/www/html/:

"Cd /var/www/html/"

"Touch miscript.php"

(Copiamos el ejemplo de la página del manual de php dentro del script)

http://php.net/manual/es/function.mysql-connect.php

```
<?php
$enlace = mysql_connect('localhost', 'root', 'practicas,ISE');
if (!$enlace){
        die('No pudo conectarse: ' . mysql_error());
}
echo 'Conectado satisfactoriamente';
mysql_close($enlace);
?>
```

Hacemos petición desde el cliente

"curl <ipservidor>/miscript.php"

no funciona el código php

Modificamos el archivo de configuración de apache (/etc/httpd/conf/httpd.conf):

Tocar la directiva directoryIndex (ifModule dir_module) en apache para que ejecute todos los archivos .php (*.php)

Reiniciamos el servicio:

"systemctl restart httpd"

"systemctl status httpd"

"curl <ipservidor>/miscript.php"

Error al pedir página igual

solucionar:

php miscript.php

nos dice que no tenemos la biblioteca mysql_connnect

para instalarlo:

"yum search php | grep mysql"

"yum install php-mysql -y"

y ya funciona el script:

"php misript.php"

Aunque podamos ejecutar el script, todavía no podemos hacer peticiones a paginas php a nuestro servidor web porque SElinux nos bloquea

para listar las restricciones de SElinux:

"getsebool -a | grep httpd"

activamos que http pueda conectar a una bbdd:

"setsebool -P httpd_can_network_connect_db on" (la p mayúscula)

para que se habilite el cambio en selinux hay que reiniciar el servicio:

"systemctl restart httpd"

Ya debe funcionar todo perfectamente, para comprobar:

Con otra máquina*: "curl <ipservidor>/script.php"* o lo abrimos con el navegador

Atentos al contexto:

Si escribimos el script en el home de un usuario y lo movemos a /var/www/html seLinux no nos deja hacer peticiones a este archivo porque tiene contexto de una carpeta home

Lo arreglamos con:

"restorecon <ruta al archivo>"

Herramientas de seguridad:

Fail2ban:

Banea ips que atacan al servidor haciendo demasiadas peticiones seguidas. (mete esas ips en cárceles)

Fail2ban → EPEL (hay que instalarlo para poder usar fail2ban)

Para instalar y comprobar el estado de fail2ban:

"Yum install epel-release"

"Yum install fail2ban"

"Systemctl enable fail2ban"

"Systemctl start fail2ban"

"Systemctl status fail2ban"

Para mostrar información de las cárceles activas:

"Fail2ban-client status"

(número de cárceles 0)

Para añadir una cárcel hay que modificar el fichero de configuración de fail2ban:

Esta en la ruta: /etc/fail2ban

jail.local (no existe al inicio) será nuestro archivo de configuración porque si jail2ban se actualiza modifica el jail.conf y nos da esta posibilidad

entonces, para añadir una cárcel, primero copiamos el archivo y después modificamos jail.local:

"cp -a /etc/fail2ban/jail.conf /etc/fail2ban/jail.local"

modificamos el archivo:

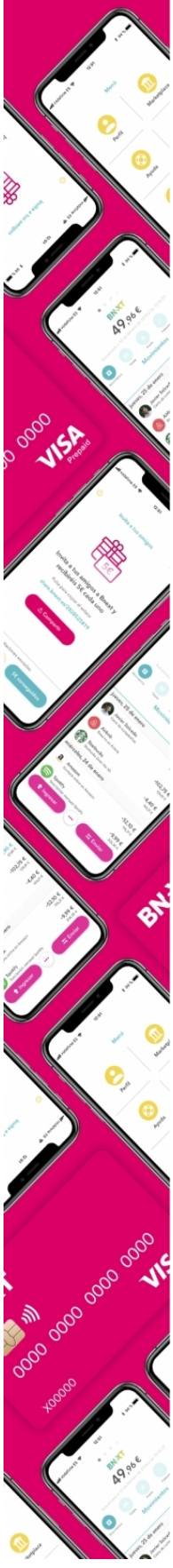
"vi /etc/fail2ban/jail.local"

enable = true

cambiar el puerto a 22022

reiniciamos servicio con:

"systemctl restart fail2ban"



para probar que funciona bien intentamos conectar mal al servidor ssh. Como vemos, mete en la cárcel a la ip tras 5 intentos fallidos (se puede cambiar el número de intentos en el archivo de configuración anterior <maxretry>)

para sacar a alguien de la cárcel:

"fail2ban-client set sshd unbanip <ip>"

Para meterlo manualmente:

"fail2ban-client set sshd banip <ip>"

Tmux y screen

Tmux y screen (reengancha los procesos que se mueren por herencia)

Yum install tmux

Yum install screen

Screen -list (lista los ficheros colgando de screen)

Screen -r <número de tarea> (sigue con la tarea salvada)

Tmux ls (lista los ficheros colgando de tmux)

tmux attach -t <número de tarea> (sigue con la tarea salvada)

Rootkit hunter

(tambien es necesario instalar EPEL).

Rootkit = vulnerabilidades.

Rootkit hunter: rkhunter (Busca malware en el ordenador)

Sería conveniente hacer un análisis con este programa al menos una vez a la semana (o más veces)