**Reset Root Password**

|  |
| --- |
| **reboot**  **premere “e” al boot**  **aggiungere dopo “UTF-8” rd.break**  **Ctrl+x**  **mount -o remount r,w /sysroot**  **chroot /sysroot**  **passwd**  **touch /.autorelabel** |

**Grub Menu**

|  |
| --- |
| **Per ottenere i dettagli sul kernel attivo corrente, digitare:**  **# grub2-editenv list**  **saved\_entry=CentOS Linux (3.10.0-229.11.1.el7.x86\_64) 7 (Core)**  **Per ottenere l’elenco dei kernel visualizzati all’avvio, digitare:**  **# grep ^menuentry /boot/grub2/grub.cfg**  **menuentry 'CentOS Linux (3.10.0-229.20.1.el7.x86\_64) 7 (Core)' ...**  **menuentry 'CentOS Linux (3.10.0-229.14.1.el7.x86\_64) 7 (Core)' ...**  **menuentry 'CentOS Linux 7 (Core), with Linux 0-rescue-f19b719117b44bf3a3fb777bd4127' ...caf**  **Per definire il kernel da eseguire al momento dell’avvio (qui 0 per la prima voce), digitare:**  **# grub2-set-default 0 oppure 'CentOS Linux (3.10.0-229.20.1.el7.x86\_64) 7 (Core)'**  **Il file di configurazione principale per il grub è:**  **/boot/grub2/grub.cfg \***    **\*Do not edit this file directly. Use the grub2-mkconfig command to generate grub.cfg**    **/etc/default/grub**    **# cat /etc/default/grub**  **GRUB\_TIMEOUT=5**  **GRUB\_DISTRIBUTOR="$(sed 's, release .\*$,,g' /etc/system-release)"**  **GRUB\_DEFAULT=saved**  **GRUB\_DISABLE\_SUBMENU=true**  **GRUB\_TERMINAL\_OUTPUT="console"**  **GRUB\_CMDLINE\_LINUX="crashkernel=auto rd.lvm.lv=cl/root rd.lvm.lv=cl/swap rhgb quiet"**  **GRUB\_DISABLE\_RECOVERY="true"**  **Eseguire grub2-mkconfig per visualizzare le impostazioni correnti. Questo comando non va a sovrascrivere la configurazione esistente.**  **# grub2-mkconfig**  **Generazione del file di configurazione di grub ...**    **Se vengono apportate modifiche a uno di questi parametri, è necessario eseguire grub2-mkconfig per rigenerare il file /boot/grub2/grub.cfg. Per esempio:**  **grub2-mkconfig –o /boot/grub2/grub.cfg**  **Qualsiasi modifica qui può quindi essere inserita nel grub2-mkconfig che genererà l’output che sarà poi reindirizzato in /boot/grub2/grub.cfg** |

**Rescue mode and emergency mode**

|  |
| --- |
| **reboot**  **premere “e” al boot**  **aggiungere dopo “UTF-8” systemd.unit=rescue.target or emergency.target**  **Per troubleshooting rimuovere rhgb and quiet** |

**Configuring Network**

**Cambiare hostname**

|  |
| --- |
| **hostnamectl set-hostname station.domain11.example.com --static**  **ping hostname**  **ping ip, dns, and gw4** |

List all configured interfaces on the server:

|  |
| --- |
| # **nmcli connection ->** List all configured interfaces on the server:  # **nmcli device status/show->** Show the status of all available interfaces on the server: |

|  |
| --- |
| # **nmcli connection add ifname eth0 con-name eth0 type ethernet ip4 192.168.1.174/24 gw4 192.168.1.254**    **ifname** [ifname]: Interface to bind the connection to. A special value of “\*” can be used for interface-independent connections.  **con-name** [connection\_name]: Connection name. **This is optional. When not provided, a default name is generated**, [type][-ifname][-num]. |

|  |
| --- |
| # **nmcli connection modify eth0 connection.autoconnect yes -> ONBOOT=yes**  # **nmcli connection modify eth0 ipv4.method manual -> Set the IP configuration to a static IP address (BOOTPROTO=none)**  # **nmcli connection modify eth0 ipv4.dns 8.8.8.8** |

Restart the interface and the network:

|  |
| --- |
| # **nmcli connection down eth0**  # **nmcli connection up eth0**  **and**  **# systemctl restart network**  **in alcuni casi potrebbe essere necessario l’enable e il restart di NetworkManager**  **lshw -> per vedere nome del device** |

**Per calcolare la subnet mask direttamente da RHEL7**

|  |
| --- |
| **ipcalc -m 192.168.1.120/25 -> NETMASK=255.255.255.128** |

**Abilitare ip forwarding sul server**

|  |
| --- |
| **sysctl --all | grep ip\_forward** -> per la sintassi  **vim /etc/systcl.conf**  **net.ipv4.ip\_forward=1**  **sysctl -p -> per verificare (dopo rebootare per prenderle in modo permanente)** |

**Manipolazione del testo (SED-AWK-GREP)**

Copia tutti i file che iniziano per a,e,i dentro a /tmp

|  |
| --- |
| **# cp /etc/[aei]\* /tmp/** |

^ si intende l'inizio di una riga di testo quindi il comando greppa tutti i file che iniziano per n in /etc/passwd

|  |
| --- |
| **# grep ^n /etc/passwd**  nobody:x:99:99:Nobody:/:/sbin/nologin  nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin  ntp:x:38:38::/etc/ntp:/sbin/nologin |

$ si intende la fine di una riga di testo quindi il comando greppa tutti i file che finiscono per h in /etc/passwd

|  |
| --- |
| # **grep h$ /etc/passwd**  root:x:0:0:root:/root:/bin/bas**h**  user:x:1000:1000:user:/home/user:/bin/bas**h**  lello:x:1001:1001::/home/lello:/bin/bas**h** |

Sostituisce dal file tutte le linda con giulia

|  |
| --- |
| # **sed -i s/linda/giulia/g nome\_file** |

Printa la 5° riga di /etc/passwd

|  |
| --- |
| **# sed -n 5p /etc/passwd**  lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin |

Stampa il primo campo di tutte le righe del file /etc/hosts

|  |
| --- |
| **# cat /etc/hosts | awk ‘{print $1}’**  127.0.0.1  ::1  192.168.1.174 |

Modalità interactive+editing, serve per cancellare la 4°riga del file users

|  |
| --- |
| **# sed -i -e ‘4d’ users** |

Rimuove riga 2,20 e 25 del file users

|  |
| --- |
| # s**ed -i -e ‘2d,20,25d’ users** |

**Comandi utili**

|  |
| --- |
| **man -k nmcli**  **ls --help**  **man -k passwd | grep -e ‘1’ -e ‘8’ -**> greppa tutte le linee che contengono 8 e 1  **alt + 126 = ~**  **mandb**  -> aggiorna i database delle man pages  **cat /etc/services ->** mostra tutte le porte utilizzate  **cat /usr/lib/firewalld/services** -> xml dei servizi con le porte |

**Scambio chiavi SSH**

|  |
| --- |
| **# ssh-keygen** -> genera una chiave pubblica e una privata, dopo aver lanciato il comando ssh-keygen troveremo una nuova cartella /root/.ssh con al suo interno:  -rw-r--r--. 1 root root 176 6 set 08.41 **known\_hosts** (ip dell’host verso il quale si è fatto ssh)  -rw-r--r--. 1 root root 399 20 feb 17.23 **id\_rsa.pub** (chiave pubblica generata)  -rw-------. 1 root root 1675 20 feb 17.23 **id\_rsa** (chiave privata generata)  **# ssh-copy-id** **x.x.x.x** -> copia la chiave pubblica sul server target (dove si vuole accedere in ssh). Sul server target una volta lanciato il comando ssh-copy-id verrà generata sotto /root/.ssh il file authorized\_keys (sarà la stessa chiave che è presente sotto id\_rsa.pub nell'esempo di cui sopra)  **# ssh-copy-id -i id\_rsa.pub numas\_sftp\_client2@172.23.11.203** -> copia la chiave pubblica sul server target 172.23.11.203 con l’utenza numas\_sftp\_client2  **# systemctl restart sshd** |

**Change runlevel/Set runlevel**

|  |
| --- |
| **# systemct get-default -**> mostra il runlevel attuale  multi-user.target  **# systemctl isolate graphical.target** -> passa da N3 a N5 senza bisogno di reboot  **# systemctl isolate multi-user.target** -> passa da N5 a N3 senza bisogno di reboot  **# systemctl set-default multi-user.target** -> imposta al reboot runlevel3  **# systemctl set-default graphical.target** -> imposta al reboot runlevel5 |

**Archivia, comprimi, decomprimi e decomprimi i file usando tar, star, gzip bzip2**

|  |
| --- |
| **tar cvf = crea il file compresso**  **tar tvf= mostra il contenuto**  **tar xvf= estrae l'archivio compresso**  **# tar cvf homes.tar /home** -> crea nella dir corrente (se non diversamente specificato) un archivio di nome "homes.tar" di tutto il contenuto di /home  Di seguito tre metodi:  1) # **tar cvf /root/etc.tar /etc**  2) **# tar czvf /root/etc.tar.gz /etc** -> gzip  3) # **tar cjvf /root/etc.tar.bz2** /etc -> bzip2  **N.B: La sequenza delle opzioni in tar è fondamentale**  Quando vogliamo scompattare l’archivio e vogliamo dirgli la directory dove scompattare è importante dare l'opzione -C  # **tar xvf /root/etc.tar** **-C** /tmp -> estrae il .tar dentro /tmp  **Per verificare che sia stato compresso nel formato giusto: root@client tmp]# file ciccio.tgz**  ciccio.tgz: **gzip compressed data**, from Unix, last modified: Fri Apr 26 11:27:51 2019 |

**Find e opzioni (-exec)**

|  |
| --- |
| **# find / -user julice -exec cp -rvfp {} /root/findresults/ \;** -> find the file which owned by user julice and copy the file into /root/findresults directory  **# find /usr/share/ -iname "uming.\*" -exec cp -rvfp {} /tmp/ \;** -> trova in /usr/share il file insensitive che inizia con uming.\*  **# find / -size +10M -exec cp -rvfp {} /tmp \;** -> trova i file più grandi di 10 Mega e li copia in /tmp |

**Managing user and groups**

Quando viene creato un utente di default viene creato l’uid, il gid e automaticamente un gruppo con lo stesso nome

|  |
| --- |
| **# useradd utente1**  **# id utente1**  **uid=1004(utente1) gid=1004(utente1) groups=1004(utente1)** |

Per creare un account utente locale con **un gruppo secondario** (qui team), digitare:

|  |
| --- |
| **# useradd -G team user** |

Per dare i privilegi di root all’account user basterà associarlo al gruppo wheel

|  |
| --- |
| **# usermod -aG wheel user** ->il comando **-a** sta per append e aggiunge un nuovo gruppo oltre a quello secondario, se omesso sovrascrive il gruppo secondario |

Per creare un account utente locale senza accesso alla shell (utile per la configurazione di Samba), digitare:

|  |
| --- |
| **# useradd -s /sbin/nologin user** |

Elimina lisa dal gruppo wheel

|  |
| --- |
| **# gpasswd -d lisa wheel** |

Cambiare UID utente

|  |
| --- |
| **# usermod -u xxx nome\_utente** |

For instance, use bob to have the account for user bob **expire on December 31, 2015.**

|  |
| --- |
| **# chage -E 2015-12-31 bob** |

To see current password management settings, use chage –l nome\_utente

|  |
| --- |
| **# chage -l utente1**  **Last password change : Feb 26, 2019**  **Password expires : never**  **Password inactive : never**  **Account expires : never**  **Minimum number of days between password change : 0**  **Maximum number of days between password change : 99999**  **Number of days of warning before password expires : 7** |

**File di configurazione user e gruppi**

|  |
| --- |
| **/etc/skel ->** /etc/skel directory contiene i file e le directory che sono automaticamente copiati nelle home directory quando viene aggiunto un nuovo utente con useradd (Use **cd /etc/skel** to go to the /etc/skell directory. Type mkdir Pictures and mkdir Documents to add two default directories to all user home directories)  **vim /etc/login.defs -**> contiene diverse opzioni per la creazione degli utenti come (PASS\_MAX\_DAYS, UID\_MIN, CREATE\_HOME yes)  /etc/default/useradd ->  # useradd defaults file  GROUP=100 -> default group 100  HOME=/home -> default /home  INACTIVE=-1  EXPIRE=  SHELL=/bin/bash  SKEL=/etc/skel  CREATE\_MAIL\_SPOOL=yes  **/etc/shadow** -> ci sono le password degli utenti criptate, le scadenze ecc  **/etc/groups** -> group info  **/etc/gshadow**-> password has group admins |

**Gestione permessi**

|  |
| --- |
| **chown :sales sale**s (cambia solo il gruppo) = **chgrp sales sales**  **u**ser= **read 4** file (open) / dir (list)  **g**roup= **write 2** file (modify) / dir (create, delete)  **o**ther= **execute 1** file (run) / dir (cd)  Solitamente nelle directory si hanno sempre read and execute (altrimenti non si può lanciare un programma o entrare dentro la cartella)  **chmod +x afile** -> execute permission per tutti (ugo)  **chmod g+w,o-r somefile** -> da i permessi di scrittura al gruppo, e toglie quelli di lettura a other |

**Applying Advanced Permissions**

|  |
| --- |
| **set user ID (SUID=4) ->**Il permesso setuid viene tipicamente usato per permettere ad utenti non privilegiati di eseguire particolari programmi con i privilegi del proprietario o dell’amministratore (root). Si applica solo ai file.  **set group ID (SGID=2) ->** Se applicato su un file eseguibile, fornisce all'utente che esegue il file autorizzazioni del proprietario del gruppo di quel file. Il permesso setgid applicato ad una directory fa in modo che tutti i nuovi file e subdirectory creati al suo interno abbiano come gruppo assegnato quello del gruppo della directory che li contiene anziché quello principale dell'utente.  **sticky bit (=1) ->** Se applicato sulle directory protegge i file al suo interno in quanto quest’ultimi possono essere cancellati solo se si è l’owner del file o della directory parente. Si applica solo alle directory |

**Esempi di permessi avanzati**

|  |
| --- |
| SUID si applica con **chmod u+s nome\_file**  # ls -l /usr/bin/passwd  -rwsr-xr-x. 1 root root 32680 Jan 28 2010 /usr/bin/passwd  SGID si applica **chmod g+s nome\_directory** oppure **chmod 2770 nome\_directory**  # ls -ld account  drwxr-**s**r-x. 2 root account 4096 Apr 30 21:28 account  Lo sticky bit si applica con **chmod o+t nome\_cartella**  drwxrwxrwt 18 nico nico 4096 2012-03-10 18:48 nomecartella |

|  |
| --- |
| Umask è il numero sottratto dalle autorizzazioni standard durante la creazione di un file. Differisce da chmod, in quanto se setto l’umask a 444 significa che tolgo i permessi di lettura a tutti, se metterò 222 tolgo i permessi di scrittura a tutti.  Umask decide le autorizzazioni predefinite o le autorizzazioni di base fornite ogni volta che un nuovo file (o directory) viene creato su Linux. **Per impostazione predefinita, umask è impostato su 022 (0022) come UMASK predefinito**.  Quando abbiamo come default umask 022 e viene creato un nuovo file / directory, viene data la seguente autorizzazione:  1. Per i file: **666** – umask(022) = 644 (rw-r-r)  2. Per le directory: **777** – umask(022) = 755 (rwx-rx-rx)  Importante Umask non può dare permessi di esecuzione ai file ma può farlo con le directory  Per cambiare Umask per l'intero sistema, possiamo cambiare il valore in **/etc/profile** (raccomandato) o qualsiasi altro file di avvio del sistema.  # **cat / etc / profile | grep -i umask** # Per impostazione predefinita, vogliamo che umask venga impostato. Questo lo imposta per la shell di login  umask 002  umask 022  Come mai su i file i permessi sono 666? e non 777? Questo lo si spiega perché' in GNU/linux i file non sono mai eseguibili ,se non diversamente deciso dall'utente. Mentre per le directory è necessaria l’esecuzione per entrarci dentro.  Per settare l'umask su uno user specifico (es. linda) -> /home/linda aggiungere a questo file l’umask che si desidera:  **-rw-r--r--. 1 linda linda 193 11 apr 2018 .bash\_profile** |

|  |
| --- |
| Quando le autorizzazioni di base dei file non sono sufficienti, si può utilizzare ACL. ACL è l’acronimo di Access Control List e serve nel momento in cui su uno stesso file dobbiamo dare dei permessi specifici per degli utenti specifici.  Il comando **getfacl** nome\_file mostra l’acl sul file  Per consentire l’accesso in lettura/scrittura all’utente bob sul file chiamato file (-m per modifica, u per utente, rw- per accesso in lettura/scrittura), digitare:  # **setfacl -m u:bob:rw- file**  Per rimuovere le autorizzazioni permesse all’utente bob (-x per rimuovere, u per utente), digitare:  **# setfacl -x u:bob file**  Per rimuovere tutti gli ACL su un file chiamato f (-b per rimuovere tutto), digitare:  **# setfacl -b file**  Per consentire autorizzazioni di lettura/esecuzione al gruppo chiamato team su una directory dir e tutti i file all’interno (-R per ricorsiva, -m per la modifica, g per il gruppo, r-x per l’accesso in lettura/esecuzione), digitare:  **# setfacl -R -m g:team:r-x dir**  To remove all the permissions for a user, group, or others, use the -x option and do not specify any permissions:  **# setfacl -x rules files**  **# setfacl -m d:g:sales:rx account/** -> la d davanti sta per default per impostarlo di default |

**Imposta le modalità enforcing e permissive in SELinux**

|  |
| --- |
| **/etc/selinux/config**  # This file controls the state of SELinux on the system.  # SELINUX= can take one of these three values:  # enforcing - SELinux security policy is enforced.  # permissive - SELinux prints warnings instead of enforcing.  # disabled - No SELinux policy is loaded.  **SELINUX=enforcing**  # SELINUXTYPE= can take one of three two values:  # targeted - Targeted processes are protected,  # minimum - Modification of targeted policy. Only selected processes are protected.  # mls - Multi Level Security protection.  SELINUXTYPE=targeted |

**SElinux può funzionare in tre modalità diverse (enforcing, permissive e disabled) come si evince dal file di configurazione di cui sopra.**

|  |
| --- |
| **# sestatus** -> server per ottenere lo status di SELinux corrente |

**oppure**

|  |
| --- |
| **# getenforce** -> per vedere se è attualmente in enforcing mode o permissive mode |

**Per passare da permissive ad enforcing mode, si può usare il comando  
setenforce**

|  |
| --- |
| **# setenforce 0** -> setta SELinux in permissive mode  **# setenforce 1** -> setta SELinux in **enforcing** mode  Per far prendere le modifiche bisogna rebootare il server |

**Selinux**

SELinux segna ogni singolo oggetto su una macchina con un contesto (context). Ciò significa che ogni file, daemon e processo ha un context. Il context è suddiviso in tre parti: **utente (user), ruolo (role) e tipo (type).**

**type è quello che ci interessa per RHCSA e soprattutto la maggior parte dei problemi sono dovuti ad un errato settaggio del type.**

**Elencare e identificare il contesto di file, processi e porte di SELinux:**

|  |
| --- |
| **[root@client ~]# ls -Z /etc/hosts**  **-rw-r--r--. root root unconfined\_u:object\_r:net\_conf\_t:s0 /etc/hosts** |

|  |
| --- |
| **# ps -eZ -> per processi  # netstat -tulnapZ -> per porte** |

**Di fondamentale importanza i seguenti MAN:**

|  |
| --- |
| **man semanage-fcontext**  **man semanage-boolean**  **man setsebool** |

**Ripristinare i contesti dei file predefiniti:**

|  |
| --- |
| **# restorecon -R /path o restorecon -v file** |

**I file creati o copiati sotto una specifica dir ereditano i contesti di quest'ultima. per cui il restorecon del file andrà effettuato sotto la dir originaria altrimenti lascerà i contesti della cartella in cui è stato creato/copiato.**

**Un SELinux Boolean è una variabile che può essere attivata e disattivata senza dover ricaricare o ricompilare un criterio,**

|  |
| --- |
| **È possibile visualizzare l’elenco delle variabili booleane utilizzando il comando:**  **getsebool -a**  **Per otterenere l’elenco dei booleani di SELinux su un argomento particolare (qui ftp):**  **# getsebool -a | grep ftp**  **ftp\_home\_dir --> off**  **allow\_ftpd\_full\_access --> on**  **Per impostare un booleano SELinux specifico in base alle tue esigenze (qui ftp\_home\_dir on ), digita:**    **# setsebool -P allow\_ftpd\_anon\_write on**  Nota 1: è possibile utilizzare on o 1, off o 0 con il comando setsebool  Nota 2: l’opzione -P indica Permanente. Se non lo usi ripristinerà la sua configurazione permanente o predefinita precedente dopo il prossimo riavvio.    oppure in alternativa si può usare:    **# semanage boolean -m --on ftp\_home\_dir**    Nota3: il comando semanage boolean assegna solo configurazioni permanenti. |

|  |
| --- |
| **Per ottenere l’elenco di tutti i booleani SELinux con un valore corrente diverso da quello predefinito (opzione -C per la personalizzazione (Custom) locale), digitare:**  **# semanage boolean -l -C**  **SELinux boolean State Default Description**  **ftp\_home\_dir (on , on) Allow ftp to read and write files in the user** |

|  |
| --- |
| **I due passaggi per modificare il contesto SElinux di un file sono:**  **1. Prima aggiornare la politica di destinazione stessa (semanage fcontext)**  **2. Quindi si applica la politica aggiornata ai propri file e cartelle (restorecon)**    **Per aggiornare tutti i contesti di file e cartelle sull'intera macchina**  **$ touch /.autorelabel**  **# quindi riavvia la macchina.**  **ESEMPIO**  **Per prima cosa creiamo un file sotto /var/www/html**  **$ touch /var/www/html/hello.html**  **$ ls -lZ /var/www/html/hello.html**  **-rw-r - r--. root root unconfined\_u: object\_r: httpd\_sys\_content\_t: s0 /var/www/html/hello.html -> ha ereditato il type di httpd ovvero "httpd\_sys\_content\_t"**  **Ora diciamo che vogliamo cambiare l'attributo di sicurezza in "admin\_home\_t", quindi in base all'esempio della pagina man, facciamo:**  **$ semanage fcontext -a -t admin\_home\_t /var/www/html/hello.html**  **Quindi ora abbiamo aggiunto una nuova politica di destinazione al catalogo fcontext:**  **$ semanage fcontext -l | grep hello.html**  **/var/www/html/hello.html tutti i file system\_u: object\_r: admin\_home\_t: s0**  **(se si lancia ls -Z il file avrà ancora come type httpd\_sys\_content\_t, in quanto bisognerà lanciare il restorecon)**  **Ora per applicare questa regola dei criteri al file, dobbiamo usare il comando restorecon:**  **$ ls -lZ /var/www/html/hello.html**  **-rw-r - r--. root root unconfined\_u: object\_r: httpd\_sys\_content\_t : s0 /var/www/html/hello.html**  **$ restorecon /var/www/html/hello.html**  **$ ls -lZ /var/www/html/hello.html**  **-rw-r - r--. root root unconfined\_u: object\_r: admin\_home\_t : s0 /var/www/html/hello.html**  **il comando restorecon imposta essenzialmente il contesto di sicurezza per un file / cartella in base alle regole specificate dal comando semanage fcontext -l** |

**Analisi dei log per policy selinux:**

|  |
| --- |
| **yum -y install setroubleshoot\* selinux-policy**  [dandy@localhost ~]$ **yum -y install setroubleshoot setroubleshoot-server**  [dandy@localhost ~]$ **systemctl restart auditd**  **sealert -a /var/log/audit/audit.log**  **cat /var/log/audit/audit.log | grep -i den ---> che ti trova tutti i deny di selinux**  **grep "numero" /var/log/audit/audit.log | audit2why**  **oppure**  **grep sealert /var/log/messages**  **grep AVC /var/log/audit/audit.log**  **Per vedere nel dettaglio lanciare sealert -l "xxxxxxx" dove "xxxx" rappresenta l'id.** |

**Update the kernel package**

|  |
| --- |
| **# yum update kernel**  oppure  **# rpm -ivh kernel.rpm -> non usare l’opzione -U che sostituisce il vecchio kernel invece di aggiungerlo al boot menu**  oppure  **# vim /etc/yum.repos.d/redhat.repo**  **[kernel]**  **baseurl=http://content.example.com/rhel7.0/x86\_64/errata**  **gpgcheck=0**  **:wq**  **# yum clean all**  **# yum list all**  **# yum repolist**  **# yum install kernel -y**  **reboot per vedere se al boot sono presenti i due kernel** |

**Yum and repository**

|  |
| --- |
| **# yum repolist** -> mostra tutti i repository configurati sotto yum e abilitati per l’uso sul nostro server  # **yum repolist enabled** -> mostra solo i repository abilitati (dove enabled=1 o null)  **#** **yum repolist disabled** -> mostra solo i repository disabilitati (dove enabled=0)  **# man yum.conf** -> esempi su come si settano i parametri  **# rpm -qa (ql, qf, qd)** -> queries for rpm package |

**Yum client configuration**

|  |
| --- |
| **Make your system as yum client to the URL** [**http://192.168.2.55/rpms**](http://192.168.2.55/rpms) **->**  **cd /etc/repos.d**  **vim rhel7.repo**  **[rhel\_dvd]**  **name=Remote Yum repo**  **basurl=**[**http://192.168.2.55/rpms/**](http://192.168.2.55/rpms/)  **enabled=1**  **gpgcheck=0**  **yum clean all** -> pulisce la cache in /var/cache/yum |

**Create local yum repo**

|  |
| --- |
| **# cd /**  **# mkdir repo**  **# cd repo**  **# yumdownloader q\*** -> per scaricare in locale gli rpm  # ls  qca2-2.0.3-7.el7.i686.rpm qimageblitz-examples-0.0.6-7.el7.x86\_64.rpm qrencode-devel-3.4.1-3.el7.i686.rpm qt-devel-4.8.7-2.el7.i686.rpm  **# createrepo /repo** -> per creare indice  # ls  4335446b46a1924ba7a212656b95038731609480b52dc3887f8f6534cbcdc724-primary.xml.gz  63de8c0ab874dd80aa994f375e01445c226a45b7da4ccaff653c360901671eab-other.sqlite.bz 2  786ce2444284dd1071004d5446de8a0950217d1e54696e3619fdf6408c8c1654-filelists.xml.g z  7b1f872ee104802639e3bb7a46af1f88719a9c1a8ea8586ab3b1562a2f915a3c-primary.sqlite. bz2  9e0a6d2bcf90019b33608a6493c8522239b7eeae283fc68ae88ad58c8a986daa-filelists.sqlit e.bz2  c419d7f54113abdd4354bcf017c755e9262847d766f28598a006c19e7b9449bf-other.xml.gz  repomd.xml  **Creare yum repository from local**  cd /etc/yum.repos.d  vim my.repo  Questo è il contenuto di un tipico repository:  **[myrepo] \***  **name=my demo repository \***  **baseurl=file:///repo \***  **gpgcheck=0**  Campi obbligatori  Un altro metodo è quello di utilizzare **yum-config-manager** per generare in automatico un repo. yum-config-manager --help cercare come si adda la url. Verrà generato automaticamente un repo. Unica cosa da ricordarsi è aggiungere gpgcheck=0 |

**How to setup yum repository using locally mounted DVD**

|  |
| --- |
| Copiare il contenuto di tutto il DVD dentro una directory  **cp -Rp RHEL-7.2\ Server.x86\_64/ /localrepo**  [DVD]  name=repo dvd  baseurl=file:///localrepo/  enabled=1  gpgcheck=0  per abilitare il repository locale lanciare  createrepo -v /localrepo  yum clean all  yum repolist |

**Mount an iso/dvd**

|  |
| --- |
| # **cd /root; wget ftp://192.168.0.254/pub/boot.iso**  # **mkdir -p /media/cdrom**  # **vim /etc/fstab**  **/root/boot.iso /media/cdrom iso9660 defaults,loop 0 0** -> ci va la path della iso e oltre a defaults ci va loop (che è il device a blocchi)  # **mount -a** |

**Individuare e interpretare i files di log e i journals**

|  |
| --- |
| **La maggior parte dei file di log di sistema si trovano nella directory /var/log a causa della configurazione predefinita di SYSLOG (vedere il file /etc/rsyslog.conf).**  **Inoltre, tutti gli eventi SELinux sono scritti nel file /var/log/audit/audit.log**  altri log importanti **/var/log/secure** e **/var/log/messages**  **journalctl**  **systemctl status systemd-journald**  **journalct -xe**  **journalct -b** -> eventi dall’ultimo boot  **journalct --since=yesterday** -> eventi che sono apparsi oggi nel journal  **rsyslogd**  **systemctl status rsyslog**  /etc/rsyslog.conf |

**Scheduling task (cron/at)**

|  |
| --- |
| **yum -y install cronie**  **systemctl enable crond && systemctl restart crond**  cron - crond (regular)  at - atd (once only)  **Cron**  # For details see man 4 crontabs  # Example of job definition:  # .---------------- minuti (0 - 59)  # | .------------- ore (0 - 23)  # | | .---------- giorno del mese (1 - 31)  # | | | .------- mese (1 - 12) OR jan,feb,mar,apr ...  # | | | | .---- giorno della settimana (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat  # | | | | |  # \* \* \* \* \* user-name command to be executed  **man 5 crontab** -> ci sono gli esempi  **non lasciare mai il primo pallino come asterisco, altrimenti ripete il task ogni minuti della giornata (importante)**  1) ***\*****/*= per reiterare il task ogni tot (l'asterisco va sempre per primo e va inserito e quando ci va lo slash i pallini rimangono sempre 5)  2) **,** = per specificare due valori precisi  3) **-** = per specificare un range  **N.B. ricordarsi di mettere sempre la path assoluta dei comandi**  **crontab -e** -> crontab di root  **crontab -eu ciccio** -> editing cron di ciccio  Puoi controllare l'accesso a crontab usando due file nella directory / etc: cron.deny e cron.allow.  Se il file cron.allow esiste, solo gli utenti in esso elencati possono utilizzare cron e il file cron.deny viene ignorato.  Se cron.allow non esiste, gli utenti elencati in cron.deny non possono utilizzare cron.  #vim /etc/cron.deny  aggiungi una voce in questo file per negare a un utente di utilizzare crontab file.cron.deny il file contiene solo il nome utente.  **At**  **yum install at**  **systemctl status atd**  **systemctl enable atd**  **at 20:00**  **at> logger hello it s 8pm**  **ctrl+d -> per salvare la config**  **Si può inserire qualsiasi comando da eseguire, è come se fosse una shell**  Per rimuovere il job -> **atrm 1** (se c’è un solo job)  Per vedere i job schedulati -> **atq** |

**Managing partitions**

|  |
| --- |
| **cat /proc/partitions**  -> to check disk devices  **fdisk /dev/sdc** -> partizionamento del device /dev/sdc -> **sul device**  **partprobe /dev/sdc** -> informa il kernel dei cambiamenti sulle partizioni  **mkfs.xfs /dev/sdc1** -> crea un fs (di tipo xfs) **sulla partizione /dev/sdc1**  **mount /dev/sda3 /data -**> monta la partizione /dev/sda3 sulla directory /mnt  **umount /mnt** -> smonta la partizione /dev/sda3 da /mnt  **mkswap /dev/sdc7** -> crea una partizione di swap  **swapon /dev/sdc7** ->per abilitare partizione allo swap  **swapoff /dev/sdc7** -> per disabilitare partizione swap  N.B. ricordarsi che sulle partizioni di swap si lavora sempre sulla partizione  **vim /etc/fstab ->** file di configurazione che elenca tutti i dischi e le partizioni disponibili/montate correntemente (**al reboot**)  **swapon -a -> monta tutte le partizioni di swap di fstab**  swapon -s -> mostra tutte le partizioni di swap  **blkid ->** Il comando blkid è utilizzabile sia con tabelle di partizionamento GPT che MBR. Uno dei vantaggi di tale comando sta nel fatto di poter visualizzare l'UUID delle partizioni, utile in particolare per il montaggio di esse. |

**Esempio di partizionamento**

|  |
| --- |
| **fdisk /dev/sdc**  Command (m for help): **n -> aggiunge una nuova partizione**  Partition type:  p primary (0 primary, 0 extended, 4 free)  e extended  Select (default p): p  Partition number (1-4, default 1): 1  First sector (2048-20971519, default 2048):  Using default value 2048  Last sector, +sectors or +size{K,M,G} (2048-20971519, default 20971519): **+2G -> per creare una partizione da 2GB**  Partition 1 of type Linux and of size 2 GiB is set |

**Controllare partizioni in fdisk**

|  |
| --- |
| Command (m for help): **p -> to check partitions**  Disk /dev/sda: 8589 MB, 8589934592 bytes, 16777216 sectors  Units = sectors of 1 \* 512 = 512 bytes  Sector size (logical/physical): 512 bytes / 512 bytes  I/O size (minimum/optimal): 512 bytes / 512 bytes  Disk label type: dos  Disk identifier: 0x0006cdcd  Device Boot Start End Blocks Id System  /dev/sda1 \* 2048 1026047 512000 83 Linux  /dev/sda2 1026048 16777215 7875584 8e Linux LVM |

Prima di salvare i cambiamenti sul disco, è sempre meglio printare le partizioni create e poi salvare premendo **w (write),** se si vuole annullare q, o CTRL+C

**Cambiare file system**

|  |
| --- |
| Command (m for help): **t -> per cambiare o selezionare il tipo di fs**  Partition number (1,2, default 2): 1  Hex code (type L to list all codes): **L -> lista tutto i tipi di fs applicabili** |

Prima di salvare i cambiamenti sul disco, è sempre meglio printare le partizioni create e poi salvare premendo **w (write),** se si vuole annullare q, o CTRL+C

|  |
| --- |
| Per formattare la partizione come swap space lanciare il comando **mkswap**  **# mkswap /dev/sdc7**  Setting up swapspace version 1, size = 1048572 KiB  no label, UUID=3259aad5-164a-43db-8103-955b78ac8391  **La swap space aggiunta ancora non è utilizzata per montarla usare swapon:**  # **swapon /dev/sdc7 =** mount per partizioni normali  ora lanciando free -m si vedrà la nuova partizione di swap "attiva" aggiunta a quella esistente  Per montare la partizione di swap automaticamente al reboota manipolare il file fstab aggiungendo:  /dev/sda2 swap **swap**  defaults 0 0 **( mettere il device e non l’UIID per la swap)**  Per non usare più la partizione di swap basterà digitare **#swapoff /dev/sdc7** |

**Montare il fs**

|  |
| --- |
| #**mount /dev/sda3 /data ->** monta la partizione /dev/sda3 sulla directory /mnt lanciando solo il comando mount è possibile vedere se è stato montato correttamente il fs. Se si vuole smontare il fs bisogna utilizzare il comando **umount** sulla directory montata: # **umount /mnt**  **vim /etc/fstab ->** due modalità di montaggio permanente   1. tramite UUID   **UUID=2721e9cb-88f3-4f82-a228-27bbd2f2cdba /boot xfs defaults 0 0**  **UUID=efdebcac-eabb-4368-bd97-be459f5f7380 swap swap defaults 0 0 (rivedere se per la swap bisogna mettere UUID o device)**   1. tramite device   /dev/sda2 swap **swap**  defaults 0 0  /dev/sda3 /home ext4 defaults 0 0 |

**Creare un Logical Volume**

|  |
| --- |
| **Comandi per creare LVM**  **# lsblk -a** -> dovrebbe essere il primo comando da digitare quando si ha a che fare con le partizioni. L’opzione -a visualizza anche i dispositivi vuoti  1) **fdisk /dev/sdb -> /dev/sdb1** (8e per mbr, 8e00 per gpt)  2) **partprobe /dev/sdb**  3) **pvcreate /dev/sdb1 -**> creazione del volume fisico  4) **vgcreate vgdata /dev/sdb1** -> crea un volume group con nome “vgdata” e associ il physical volume in esso  5) **lvcreate -n lvdata -L 1020M vgdata** -> crea un volume logico di nome “lvdata” di 1020M sul volume group “vgdata” oppure se si vuole dare tutto il disco  **lvcreate -n lvdata -l 100%FREE vgdata**  6) **mkfs.xfs /dev/vgdata/lvdata** -> formatta il volume logico in xfs  7) **mount /dev/vgdata/lvdata /mnt/**  -> montaggio temporaneo del volume logico su /mnt  8) **mount** -> per verificare che sia montato  9) **fstab** -> if permanent is required  **Comandi per visualizzare**  1) **pvs, vgs, lvs**  2) **pvdisplay, vgdisplay, lvdisplay** |

**Estendere un volume logico**

|  |
| --- |
| Per estendere un volume logico (lvextend) in ordine bisogna prima:  1) fdisk + partprobe  2) **pvcreate** -> per creare il volume fisico da aggiungere  3) **vgextend** -> per estendere il vg con il nuovo phisical volume  4) **lvextend**  **1) fdisk /dev/sdb -> /dev/sdb2** add a physical device (8e per mbr, 8e00 per gpt)  **+ partprobe /dev/sdb** -> update kernel partition table  **2+3) vgextend vgdata /dev/sdb2** -> crea il physical volume + estende il volume group vgdata su /dev/sdb2  **3) lvextend -L +100M -r /dev/vgdata/lvdata** -> aggiunge +100M al volume logico /vgdata/lvdata  \* l’opzione -r serve fare il resize del logical volume+file system  in alternativa  If you have an EXT4 partition use the following command:  **resize2fs /dev/vda2**  If it's an XFS, you'll need to use this:  **xfs\_growfs /dev/vda2** |

**Ridurre un volume logico**

|  |
| --- |
| **mount** (per verificare che si tratti di un ext4 fs) **->non si può ridurre xfs file system**  **lvreduce -L -100M -r /dev/vgdata/lvext**  Do you want to unmount "/mnt"? y -> non si può ridurre ext4 fs senza smontarlo  **e2fsck -f** -> per verificare integrità dati  **mount** -> verificare che sia rimontato il fs  If you have an EXT4 partition use the following command:  **resize2fs /dev/vda2**  If it's an XFS, you'll need to use this:  **xfs\_growfs /dev/vda2**  **lvreduce -r -L 400M /dev/rhcsa/data**  ( questo comando fa diventarere data a 400MB )  **lvreduce -r -L -100M /dev/rhcsa/data** ( questo comando la riduce di 100M )  **Non si può fare un reduce di xfs filesystem** |

**Creare un volume logico o un volume group con un preciso PE, extends**

|  |
| --- |
| **# vgcreate -s 16M vg1 /dev/sda7** -> Create a volume group, and set 16M as a extends  **# vgcreate farming -s 32M /dev/sdc2** -> create a volume group farming with a PE size of 32M  **# vgcreate -s 8M redhat /dev/vda4** -> Create VG with PE size as “8MB”  **#** **lvcreate -l 50 -n lvm02** -> Create a logical volume that contains 50 extends  **#lvcreate -n cattle -l 25 farming** -> Create a new logical volume cattle with a size of 25 extents  **# lvcreate -l 100 -n fedora redhat** -> Create the “LVM” with the name “fedora” by using 100PE’s |

**Creare swap file**

|  |
| --- |
| **dd if=/dev/zero of=/swapfile bs=1M count=600**  **swapon /swapfile**  **/swapfile swap swap defaults 0 0**  **swapon -a** |

**Configuring time services**

|  |
| --- |
| **In RHEL 7 vengono utilizzati due pacchetti principali per impostare il lato client: ntp e chrony** Attenzione: ntpd e chronyd non dovrebbero essere eseguiti allo stesso tempo. Scegli uno e solo uno di loro! Ci sono rapporti da parte dei candidati RHCE che notano che uno di loro è già intenzionalmente in esecuzione all’inizio dell’esame  **# yum -y install chrony** **# systemctl enable chronyd.service && systemctl start chronyd.service**  Prima di ogni altra cosa, è necessario assegnare il fuso orario corretto.Per ottenere la configurazione corrente, digitare:  **# timedatectl** -> To get the current configuration timezones  **# timedatectl list-timezones** -> To get the list of all the available time zones  **# timedatectl set-timezone America/Los\_Angeles** -> to set-timezone  **# timedatectl set-ntp yes -> NTP enabled # vim /etc/chrony.conf -> add “server classroom.example.com iburst” decommentando gli altri # systemctl restart chronyd  # chronyc sources -V** -> per verificare!!!! **# system-config-date** -> yum system-config-date-1.10.6-2.el7.centos.noarch |

**Gestione firewalld**

|  |
| --- |
| **Firewall is implemented in kernel (netfilter) to manage netfilter**  **firewalld (firewall-cmd & firewall-config)**  **firewall-cmd --list-all**  **is better to work with services (and not port)**  **cd /usr/lib/firewalld/services -> .xml**  **firewall-cmd --get-services -> service available**  **firewa-cmd --list-services -> currently available**  **/etc/firwalld/services -> create an .xml custom service**  Aggiungi il servizio HAProxy alla zona predefinita in modo permanente e ricaricare la configurazione del firewall:  **# firewall-cmd --permanent --add-service=haproxy**  **success**  **# firewall-cmd --reload**  **success**  Gestione delle porte  **La gestione delle porte segue lo stesso modello della gestione dei servizi.**  **Per consentire temporaneamente la porta 443/tcp nella zona interna, digitare:**  **# firewall-cmd --zone=internal --add-port=443/tcp --permanent**  **success**  **# firewall-cmd --reload**  **success**  **# firewall-cmd --list-ports**  **443/tcp**  Gestione da interfaccia  **firewall-config** |

**NFS**

|  |
| --- |
| **Server: 192.168.92.138**  **Client: 192.168.92.137**  Installare i seguenti pacchetti sul server:  **# yum install nfs-utils\***  **# systemctl status nfs && systemctl enable nfs && systemctl start nfs**  **verificare anche**  **# systemctl status rpcbind && systemctl enable rpcbind && systemctl start rpcbind**  **# systemctl status nfs-mountd.service && systemctl enable nfs-mountd.service && systemctl start nfs-mountd.service**  **SERVER SIDE**  Se una share NFS è esportata da un NFS Server, la sintassi della share sarà:  **servername:/sharename**  Digitare **# mkdir /sharednfs** (ricordarsi di dargli tutti permessi, chmod 755 o 777) per creare sul server la directory che si vuole sharare col client.  Una volta creata la directory sul server 192.168.92.138 la directory /sharednfs bisognera includere nel file **/etc/exports:**  **/sharednfs 192.168.92.137(rw)**  -> per uno specifico client  **/sharednfs \*(rw)** -> per tutti i client  **/sharednfs 192.168.92.0/24(rw)**  -> per tutta la subnet  Digitare **# exportfs -rav** per rendere effettiva le modifiche su /etc/exports e restartare nfs  # exportfs -rav  exporting 192.168.92.137:/sharednfs  exporting \*:/sharednfs  exporting \*:/home/ldap  exporting \*:/data  exporting \*:/srv/nfs  firewall-config -> interfaccia grafica per firewalld  firewall-cmd --list-all  **firewall-cmd --permanent --add-service=nfs**  **firewall-cmd --permanent --add-service=mountd**  **firewall-cmd --permanent --add-service=rpc-bind**  firewall-cmd --reload  **CLIENT SIDE**  Installare i seguenti pacchetti sul client:  **# yum install nfs-utils\***  **# systemctl status nfs && systemctl enable nfs && systemctl start nfs**  **verificare anche**  **# systemctl status rpcbind && systemctl enable rpcbind && systemctl start rpcbind**  **# systemctl status nfs-mountd.service && systemctl enable nfs-mountd.service && systemctl start nfs-mountd.service**  firewall-cmd --list-all  **firewall-cmd --permanent --add-service=nfs**  **firewall-cmd --permanent --add-service=mountd**  **firewall-cmd --permanent --add-service=rpc-bind**  **firewall-cmd --reload**  Ora l'NFS client può montarla come **192.168.92.138:/sharednfs**  **# showmount -e 192.168.92.138** -> IP del Server  Export list for 192.168.92.138:  /home/ldap \*  /data \*  /srv/nfs \*  /sharednfs (everyone)  Il comando completo per montare temporaneamente la share sulla directory /mnt in questo caso è:  **# mount 192.168.92.138:/sharednfs /mnt**  # df -h  File system Dim. Usati Dispon. Uso% Montato su  /dev/mapper/centos-root 47G 2,9G 45G 7% /  devtmpfs 475M 0 475M 0% /dev  tmpfs 487M 0 487M 0% /dev/shm  tmpfs 487M 7,8M 479M 2% /run  tmpfs 487M 0 487M 0% /sys/fs/cgroup  /dev/sda1 1014M 162M 853M 16% /boot  tmpfs 98M 0 98M 0% /run/user/0  **192.168.92.138:/sharednfs 11G 8,2G 2,1G 80% /mnt**  # mount  192.168.92.138:/sharednfs on /mnt type nfs4 (rw,relatime,vers=4.1,rsize=131072,wsize=131072,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=192.168.92.137,local\_lock=none,addr=192.168.92.138)  Per montare permanentemente la partizione al reboot in fstab:  **192.168.92.138:/sharednfs /mnt nfs \_netdev (oppure defaults) 0 0**  oppure  redirigere /etc/mtab | grep share >> /etc/fstab  Selinux:  **nfs\_export\_all\_ro --> on**  **nfs\_export\_all\_rw --> off** |

**Accessing Samba Server**

|  |  |  |
| --- | --- | --- |
| **yum -y install samba cifs-utils samba-client**  **# firewall-cmd --permanent --add-service=samba**  **# firewall-cmd --reload**   |  | | --- | | **smbclient -L //host**  **smbclient //host/share**  **tree connect failed: NT\_STATUS\_ACCESS\_DENIEDmount -t cifs //host/share /cifsmount**  **//host/share /cifsmount cifs defaults 0 0** |  |  | | --- | | **smblclient -L 192.168.1.200 oppure smbclient -L labipa \***  **\*va sul server samba senza autenticazione, infatti non serve neanche la password (non vengono specificate le share per specifici utenti)**  **smbclient -L 192.168.1.200 -U sambaguru \***  **password: guru**  **\* Sharename Type Comment**  **sambaguru Disk Home Directories**  **Per identificare share specifiche per un utente**  **smbclient //192.168.1.200/sambaguru -U sambaguru**  **Enter sambaguru's password:**  **smb: \> ls**  **mkdir /cifsmount \***  **\* directory dove montare la share**  **mount -t cifs -o username=sambaguru,password=guru //192.168.1.200/sambaguru /cifsmount**  **oppure**  **mount -o username=sambauser //labipa/data /mnt**  **fstab**  **//192.168.1.200/sambaguru /cifsmount cifs \_netdev,username=sambaguru,password=guru 0 0**  **//labipa/data /samba cifs \_netdev,username=sambauser,password=password 0 0**  **samba\_share\_nfs --> off -> setsebool -P** | |

**Autofs**

|  |
| --- |
| CLIENT SIDE  **yum install autofs -y**  **systemctl enable --now autofs && systemctl start autofs**  **# showmount -e 192.168.92.138**  Export list for 192.168.92.138:  /home/ldap \*  /data \*  /srv/nfs \*  /sharednfs (everyone) -> vogliamo settare questa come autofs  # **cat /etc/auto.misc -> esempio di setting**  #  # This is an automounter map and it has the following format  # key [ -mount-options-separated-by-comma ] location  # Details may be found in the autofs(5) manpage  cd -fstype=iso9660,ro,nosuid,nodev :/dev/cdrom  # the following entries are samples to pique your imagination  **#linux -ro,soft,intr ftp.example.org:/pub/linux**  #boot -fstype=ext2 :/dev/hda1  #floppy -fstype=auto :/dev/fd0  #floppy -fstype=ext2 :/dev/fd0  #e2floppy -fstype=ext2 :/dev/fd0  #jaz -fstype=ext2 :/dev/sdc1  #removable -fstype=ext2 :/dev/hdd  editare il file **/etc/auto.master**  # Sample auto.master file  # This is a 'master' automounter map and it has the following format:  # mount-point [map-type[,format]:]map [options]  # For details of the format look at auto.master(5).  #  /misc /etc/auto.misc  **/sharednfs /etc/auto.sharednfs** -> aggiunta questa riga  poi creare un file auto.”nomeshare”  **vim /etc/auto.sharednfs ->** creare questo file e aggiungere:  **linux -fstype=auto,rw,sync 192.168.1.59:/testb -> ricordarsi fstype!!!!!**  subdir -rw,sync 192.168.92.138:/sharednfs -> la subdir va inventata  ldapuser0 -rw,sync 172.25.254.254:/home/guests/& -> è la home di ldap  \* -rw,nfs4 instructor.example.com:/home/guests/& -> è la home di ldap  # **systemctl restart autofs**  per verificare che sia ok al reboot, entrare nella cartella /sharednfs/subdir |

**Setting up an LDAP client**

|  |
| --- |
| **# yum -y install sssd\* nss-pam\* autofs\* krb5\* auth\* openldap-client\* nfs-utils\***  **o in alternativa**  **# yum group install “Directory Client”**  **systemctl enable sssd**  **systemctl enable nfs**  **systemctl enable autofs**  **firewall-cmd --add-service=nfs --permanent**  **firewall-cmd --add-service=rpc-bind --permanent**  **firewall-cmd --add-service=mountd --permanent**  **showmount -e kikko.deriso.com**  **Export list for kikko.deriso.com:**  **/home \***  **autconfig-tui**  **- 1 ) Selezionare “ldap password”**  **2) Inserire LDAP search Base DN: dc=example,dc=com 3) Inserire ldap server : ldap://classroom.example.com**  **4) Quindi è necessario selezionare la casella di controllo certificato tls, se viene fornita la posizione di un file .pem.**  **5) inserire nel box certificato ftp://**[**certs.codingbee.net/resources/cert.pem**](http://certs.codingbee.net/resources/cert.pem)  **autconfig-gtk**  **5) use LDAP, use shadow, use LDAP auth, Local authorization**  **6) ldap://kikko.deriso.com ; dc=kikko,dc=deriso**  **Put the LDAP server certificate into the**  **/etc/openldap/cacerts directory when asked.**  **systemctl restart enable sssd && systemctl start sssd**  **Configurazione autofs**  **in /etc/auto.master ->**  **inserire /home/ldap /etc/auto.home -> bisogna sempre inserire dopo la /home/ldap altrimenti monta lo user su /**  **in /etc/auto.home ->**  **\* -fstype=auto,rw,sync kikko.deriso.com:/home/ldap/&**  **restorecon /etc/openldap/cacerts/cert.pem**  **# getent passwd ldapuser1**  **[root@client ldap]# su - ldapuser1**  **Ultimo accesso: lun apr 1 17.58.24 CEST 2019 su pts/0**  **/usr/bin/id: cannot find name for group ID 458400003**  **authconfig --enablemkhomedir --update ???????** |

**Certificato da LFTP**

|  |
| --- |
| **yum -y install lftp**  **cd /etc/openldap/cacerts**  **lftp labipa.example.com**  **cd pub**  **ca.crt**  **get ca.crt**  **exit** |