

Chapter 0: Necesitati Virtualizare

Obiectivul cursului este ca studentii sa inteleaga implicatiile virtualizarii intr-un subsistem Linux. Acest lucru este necesar pentru a putea invata Bash scripting

Cunostinte de baza:

- Intelegerea conceptului general al masinilor virtuale si al containerelor; Intelegerea elementelor masinilor virtuale intr-un IaaS cloud (Infrastructure as a service) cum ar fi instante virtuale, block storage, networking;
- Intelegerea proprietatilor unui sistem Linux care trebuie schimbate atunci cand sistemul este clonat sau folosit ca si un template;
- Intelegerea cum imagini de sistem sunt folosite pentru "deployment" (instalare) o masina virtuala, o instanta de cloud, un container;
- Intelegerea extensiilor de Linux care il integreaza cu un produs de virtualizare; Constientizarea cloud-init.

Cuvinte cheie:

Application container (container de aplicatie)

Reprezinta aplicatii impachetate si deployate in asa fel incat ele ocolesc sistemele de operare traditionale si acceseze direct resursele calculatorului.

D-Bus machine id

Reprezinta fisierul de configurare care contine ID-ul unic al sistemului (uuid). Acest uuid trebuie generat si adaugat fisierului de configurare in timpul instalarii sau la bootarea VM-ului sau al containerului.

Guest drivers

Sunt soft-uri de virtualizare pentru sistemul de operare oaspete care aduc functionalitati aditionale precum support pentru mouse pentru a permite acestuia sa comunice cu un VM, setari de login automat, sincronizarea orei, transferul de fisiere intre gazda si oaspete.

Linux container

Reprezinta o aplicatie containerizata care depinde de un motor care sa comunice cu sistemul de operare gazda fara un hypervisor.

SSH host keys

Reprezinta un set de numere generate aleator de catre un algoritm care permite autentificarea unui user in alt sistem. (pentru mai multe detalii consultati [aici](#))

Virtual machine

Un sistem de calculare virtualizat dar care ofera acces la un sistem de operare complet instalat pe hardware virtual (obtinut prin intermediul unui hypervisor)

Si multe altele

0. Teorie:

0.1 Introducere

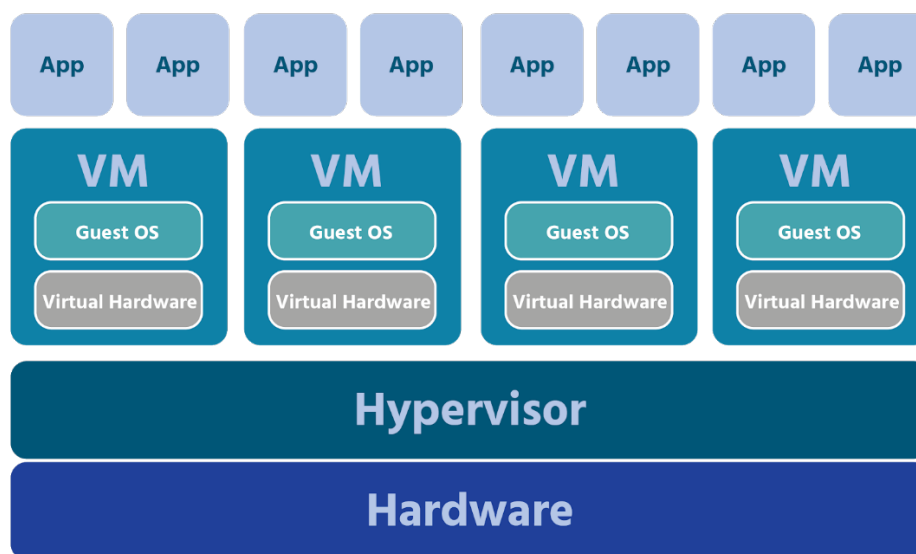
La ce va ganditi cand auziti termenul de virtualizare? Termenul vine din latinescul virtus care in timp s-a transformat in virtual si inseamna a poseda anumite virtuti. A avea anumite calitati. Dex-ul ne defineste acest concept ca fiind un lucru care exista numai ca posibilitate. Fara a se produce in fapt. Presupun ca suna abstract. Hai sa concretizam. Virtualizarea se refera la sisteme si programe care ruleaza pe platforme software mai degraba decat direct pe hardware-ul calculatorului. Eventual sistemele vor folosi resursele calculatorului, doar ca se pot obtine anumite avantaje prin faptul ca

Nota de subsol: desi neortodox, voi introduce link-uri catre pagini de wikipedia pentru a putea aprofunda si intelege mai usor un fenomen. Luati informatiile de pe wikipedia cu putina paine, ca sa nu vi se aplece de la faptul ca mai gasiti si anumite elemente neadevarate, dar care pot fi triate usor cu un pic de knowhow. Aceste link-uri pot fi accesate prin hyperlink (ctrl + click pe cuvantul underlined)

elementele respective sunt abstractizate. Ce inseamna acest lucru? Imaginati-va un sistem de calcul compus din placa de baza, memorie RAM, HDD, procesor, samd. Pe acest computer avem un sistem de operare instalat pe care o sa il numim in mod aleator Windows, si tot aleator o sa alegem o versiune de Windows. Sa zicem ca alegem Windows 10. Dar, pentru a putea lucra, avem nevoie si de Linux si nu putem avea un al doilea calculator. Presupunand ca ne dorim o mai mare flexibilitate astfel ca bootarea duala iese din discutie. Asadar vom instala o masina virtuala. O bucata de soft care ia din resursele Windows si le “traduce” in asa fel incat ele sa poata fi intelese de catre Linux. Formatarea HDD pentru Windows nu e compatibila cu Linux, asa ca ea va desprinde o bucata, si va crea un fisier de 10GB de memorie dedicata pentru sistemul de operare Linux. Ea nu exista fizic, in fapt nu partitionam direct HDD, pur si simplu creeaza un fisier, pe care sistemul de operare il va putea interpreta ca si cum ar fi ceea ce cauta el. “Rupe” din RAM-ul windows-ului o parte si il ofera masinii virtuale, “rupe” unul sau doua procesoare samd. Masina virtuala le va interpreta ca si cum ele exista de fapt, doar ca in fapt ele sunt consumate de catre o aplicatie, un hypervisor care face posibila aceasta munca. Practic, Hypervisorul creeaza resurse virtuale pe care SO oaspete le va folosi.

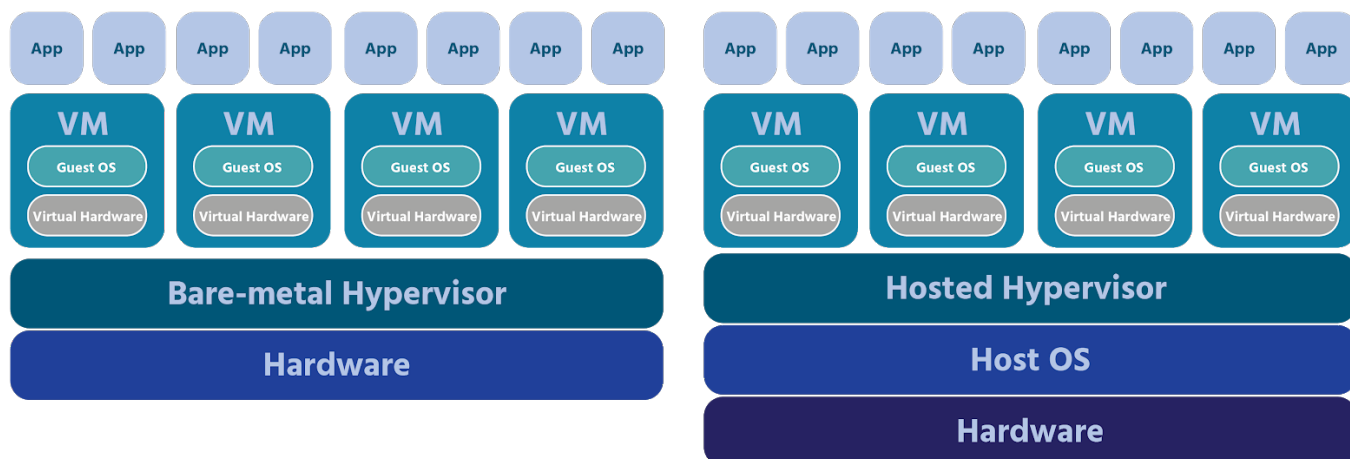
0.2 Masini virtuale si containere

(informatia din inceputul capitolului acesta se afla in 0.1 Introducere)



Virtual Machines

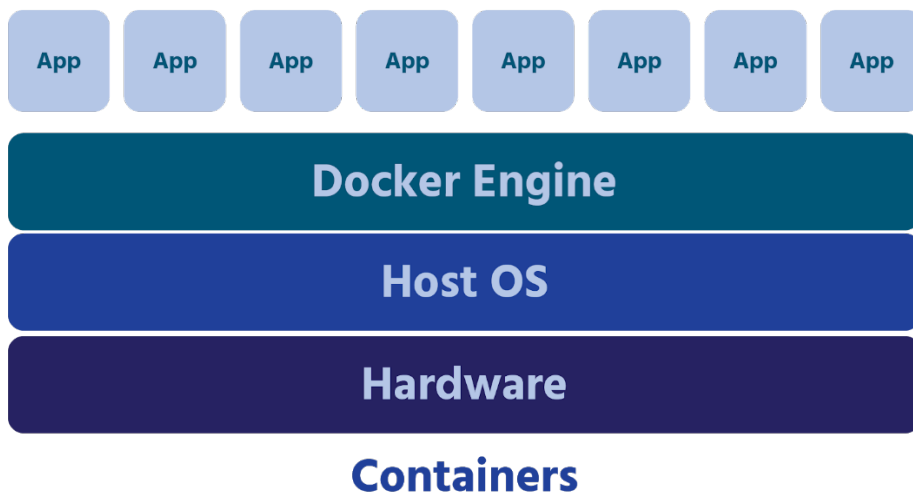
Sunt doua metode sa virtualizezi pe un computer. Prima metoda se cheama *Type 1 hypervisor*, cunoscut si ca bare-metal hypervisor – si asta implica instalarea direct pe hardware. Asta inseamna explicit ca hypervisorul de tipul 1 se afla direct intre hardware si masina virtuala. A doua metoda se cheama *hosted virtualization*. Pentru a seta acest mediu de lucru, este nevoie de *Type 2 hypervisor*, denumit si hosted hypervisor care se instaleaza intr-un sistem de operare deja existent, numit sistem gazda.



Sistemul de operare care ruleaza in interiorul unei masini virtuale (VM) se numeste sistem de operare oaspete. Acest este instalat de multe ori dupa o imagine preconfigurata, dar poate fii instalat si din surse precum CD, DVD, PXE, etc. Configurarea acesteia, impreuna cu instalarea aplicatiilor, patch-urilor de securitate, setupul network-ului este realizat de catre un administrator de sistem exact cum acesta seteaza si o masina fizica intr-un data center. Desigur, aceasta masina fiind virtuala ea poate sa imparta resurse cu o sumedenie de alte VM-uri. Totodata, aceasta ar putea fi pe partea cealalta a lumii fiind mai aproape de clientii care o utilizeaza.

Unul din beneficiile pe care le produce virtualizarea este faptul ca utilizeaza mai eficient resursele sistemului de calcul. In mod normal, pe orice hardware poate rula un singur sistem de operare, ceea ce este util pentru un user individual. Dar in cazul serverelor mari, acest lucru poate deveni inoportun intrucat pe o singura masina (desi foarte performanta) va exista un singur sistem de operare pe care vor rula o sumedenie de aplicatii. Daca sistemul nu va mai functiona, atunci toate acele aplicatii vor deveni imposibil de utilizat. Prin crearea de masini virtuale putem optimiza in asa fel incat fiecare aplicatie sa aiba mediul ei de lucru, cu setarile create corespunzator, crescandu-i performanta, diminuand costurile si obtinand high availability pentru masina respectiva. Iar daca pica sistemul acesteia, va fi afectata doar acea aplicatie, celelalte putand fi utilizate.

O tehnologie rezultata din conceptul de virtualizare este containerizarea. Aceasta este adoptata rapid de catre companiile software. O aplicatie containerizata depinde de un motor al containerului care va comunica cu SO-ul gazda fara un supervizor sau un VM oaspete. Cand un programator dezvolta o astfel de aplicatie, acesta va include toate dependintele necesare direct in container. Totodata, mai multe containere sunt construite sa comunice intre ele pentru a realiza task-uri care initial erau rulate de un singur program din interiorul SO.



In acest fel, se obtin mai multe avantaje precum: mai putine resurse necesare, timpii din ciclul de release sunt mai scurti si se abstractizeaza elemente ale programului (acestea obtin independenta

fata de SO) pentru o mai buna portabilitate. Evident acest concept vine cu propria lista de avantaje si dezavantaje.

Avantaje:

- containerele consuma mai putine resurse decat VM-urile
- sunt mai de incredere (n.r. reliable) si sunt scalabile
- necesita mai putin spatiu de stocare si ofera o performanta per total mai buna.

Dezavantaje:

- deoarece ele ruleaza direct pe hardware trebuie depuse eforturi deosebite pentru a le mentine secure
- Intrucat ele impart resurse, motorul container-ului trebuie acordat pentru a nu exista interferente intre containere, si totodata acesta trebuie sa obtina control la resurse precum memorie, conexiuni, device-uri I/P, etc.

Avantajele pe care acestea le ofera pot fi usor depasite de catre timpul si efortul depus in administrarea lor daca aceste lucruri nu sunt luate in considerare de la inceput. Din fericire, acest concept a fost adoptat rapid de catre comunitatea open-source care lucreaza la dezvoltarea lui intr-un ritm accelerat.

0.3 Solutii de virtualizare

Cand vine vorba de masini virtuale, sunt mai multe proiecte open-source si mai multi comercianti dintre care putem alege. Trebuie atentie sporita cand se ia decizia alegerii unei solutii particulare, intrucat compatibilitatea intre ele este in cel mai bun caz dificila. Masinile virtuale se "implementeaza" (they are deployed) dintr-o imagine care contine SO-ul oaspete, programele, si ce configuratii administratorul a facut pentru managementul de sisteme. Acesta este un proces intr-o oarecare masura automatizat, desi mai sunt pasi aditionali de facut manual pentru a configura reseaua si anumite setari de securitate.

- **VMware**

Una din cele mai vechi platforme de virtualizare. Detinuta de catre Dell Technologies. Platformele lor vSphere si ESXi de hypervisorii sunt apreciate de catre clientii care cauta stabilitate si support.

- **KVM**

KVM sau Kernel-based Virtual Machine este un hypervisor open-source construit in interiorul Linuxului. Sa nu se faca confuzia cu KVM switch (keyboard, video, mouse). Folosit extensiv in distributia SUSE are avantajul de a fi mai scalabil decat VMWare. Totodata, fiind open-source acesta nu are taxe de licenta.

- **Xen**

Proiectul **Xen** este un alt hypervisor open source. Este considerat hypervisor de tipul 1 intrucat ruleaza direct pe hardware, fara a avea nevoie de un SO gazda. Acest lucru, impreuna cu designul microkernel permit unui sistem lightweight sa obtina performante foarte bune. Desi este folosit frecvent impreuna cu Linux, acesta nu este legat de un singur SO. Xen este in momentul de fata singurul hypervisor de tipul 1 oferit cu licenta open source fiind un proiect colaborativ Linux Foundation

- **Hyper-V**

Hyper-V este un produs de virtualizare oferit de catre Microsoft. Popular in mediile care se bazeaza pe servere Windows, permite instalarea distributiilor Linux intr-un mediu Windows.

- **Virtual Box**

VirtualBox este un produs open-source Oracle facil de instalat si de configurat pe masini care ruleaza sisteme Linux, Macintosh sau Windows. Desi poate rula si intr-un mediu de servere, este mai indicat sa fie folosit de catre useri care au nevoie ocazionala sa ruleze cu un alt sistem de operare.

- **Docker**

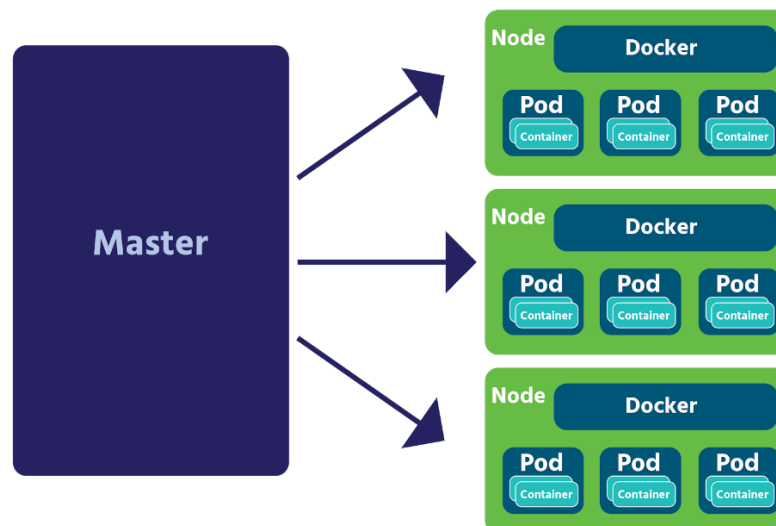
Containerele Linux devin un nou model standard in industrie pentru "networked application development". [Docker](#) si [Kubernetes](#) sunt principalele forte din miscarea de containerizare.

Docker este un motor de container care permite programatorilor si inginerilor de sistem sa creeze aplicatii containerizate. Acestea sunt componente de sine statatoare care nu au nevoie de un SO gazda pentru a isi putea executa functiile. `Containerd` faciliteaza impachetarea codului aplicatiei si dependintelor intr-un container care poate rula consistent indiferent de infrastructura de baza a sistemului. Fiind un proiect open source de top al Cloud Native Computing foundation, acesta evolueaza constant pentru a atinge nevoile industriei.

- **Kubernetes**

Kubernetes, dezvoltat initial de Google, a fost transformat in proiect open source in 2014. Acesta poate fi descris ca o platforma care ofera un mediu de management centrat pe containere pentru containere, microservicii si infrastructura cloud. Foloseste label-uri si adnotari pentru a ajuta clientii sa tina evidenta resurselor si orchestreaza calculele (tr. Computing), retelele si resursele pentru aplicatii containerizate. Cele doua aplicatii, Kubernetes si Docker sunt integrate si pot lucra impreuna.

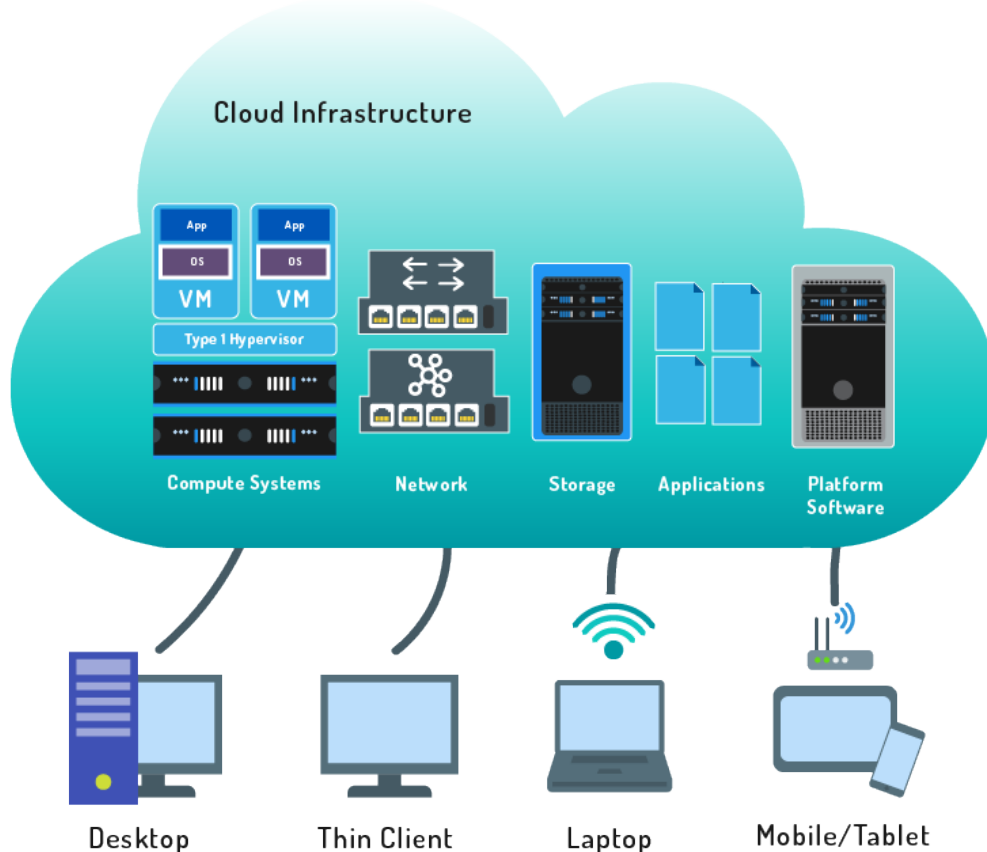
Kubernetes sunt organizati in clustere, noduri (numite anterior minions), pod-uri si containere. Nodul master este responsabil nu mentinerea starii dorite in cluster. Aceste comunica cu API-ul, porneste si opreste procese si balanseaza volumul de incarcare (workload)



0.4 IaaS

Elementele discutate mai sus reprezinta baza conceptului de IaaS (infrastructura as a Service). IaaS reprezinta o forma de cloud computing, care reprezinta livrarea de resurse de calcul la cerere

organizatiilor si utilizatorilor prin intermediul internetului.



Dar ce inseamna sa fii cu sistemul de calcul in nori? (Gluma foarte proasta

referitoare la expresia a fii cu capul in nori pornind de la o traducere mot-a-mot a expresiei “in the cloud”) Mai concret se refera la resurse ale sistemelor de calcul, aflate in data center-uri externe locatiei de lucru care pot fi accesate prin internet. Plasand instante virtuale pe sisteme de calcul administrate de un hypervisor si creand o retea prin care acestea sunt legate prin intermediul software-ului, se obtine performante mai bune cu necesitati hardware mai mici intrucat sistemele trebuie sa comunice intre ele prin intermediul API-urilor (application programming interface) unor tehnologii de orchestrare precum OpenStack, Apache Cloudstack si OpenNebula.

0.5 VM si configurarea containerelor

Cand o masina virtuala noua este creata, aceasta trebuie sa comunica cu lumea din exterior. Ca orice resursa dintr-o retea, ea va avea un hostname unic si o adresa IP, impreuna cu o masca subnet, DNS si default route. Intrucat este de dorit sa avem imagini si containere create si administrate de catre alte programe, vor fi create script-uri de initializare care vor modifica imaginea de baza. Aceasta imagine este stocata ca un set de fisiere gata de rulat, dar vor avea lipsa anumite informatii esentiale precum: setari corecte de retea, permisiuni si grupuri, rute catre database-uri, chei SSH si orice alta configuratie care ar fi in mod normal facut de catre un administrator de sistem. Aceste scripturi trebuie sa fie flexibile si adresabile de catre alte programe pentru a necesita un input uman minimal. Totodata securitatea lor este foarte importanta intrucat acestea pot contine informatii sensibile precum credentiale de logare.

Pentru acest lucru exista utilitare precum cloud-init care simplifica procesul. Aceste utilitare apeleaza programele printr-un API, citesc metadata dintr-un server si pot configura instante. In kubernetes ele sunt containere specializate numite Init container care ruleaza inainte de pornirea unei aplicatii.

Cheile SSH sunt folosite pentru a administra containerele si VM-urile. Acestea sunt generate automat, dar pot fi generate si manual prin comanda ssh-keygen. Lipsa acestora, sau chei eronate sunt unul din motivele pentru care sistemele nu pot comunica intre ele.

Un alt element important este reprezentat de UUID. Acesta trebuie adaugat in fisierul de configurare D-Bus machine ID la instalare sau la prima bootare. Comanda dbus-uuidgen genereaza sau citeste un UUID al unui sistem. Acest ID consta dintr-un numar de 128 de biti care este foarte putin probabil

sa fie replicat pana in anul 3400. Acesta ramane neschimbat pana cand sistemul este restartat. UUID-urile sunt folosite si pentru a identifica orice obiect din memoria sistemului de calcul si sunt folosite in baze de date SQL, sau limbaje de programare precum JS, Python.

```
1033_clim_antonio@sop.ase.ro:~$ cat /proc/cpuinfo
```

Output Omitted...

```
physical id      : 1
siblings        : 8
core id         : 0
cpu cores       : 4
apicid          : 32
initial apicid  : 32
fpu             : yes
fpu_exception   : yes
cpuid level     : 11
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca
cmov
pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb
rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop
_tsc aperfmperf pni pclmulqdq dtes64 monitor ds_cpl vmx smx est tm2 ssse3 cx16
xtprpdc m pcid dca sse4_1 sse4_2 popcnt aes lahfm lm tpr_shadow vnmi flexpriori
ty ept vpid dtherm ida arat
bugs           :
bogomips       : 4787.47
clflush size   : 64
```

Alte setari critice pentru functionarea corecta a VM-urilor sau containerelor o reprezinta extensiile de virtualizare. Acestea sunt suportul hardware pentru virtualizare si sunt mecanisme construite in CPU precum extensiile Intel VT-x si AMD-V. Aceste setari se pot gasi in fisierul /proc/cpuinfo si in BIOS.

In campul flags: lm indica faptul ca sistemul are o arhitectura pe 64 de biti si flag-ul vmx indica faptul ca extensia Intel VT-x este activata in bios.

Foarte pe scurt explicat, exista doua arhitecturi de procesoare. Acestea se gasesc sub denumirile x86 si x64. x86 reprezinta arhitectura pe 32 de biti iar x64 reprezinta arhitectura pe 64 de biti.

Solutiile de virtualizare sunt compatibile doar pe arhitectura de 64 de biti. In Linux se poate verifica arhitectura CPU-ului cu comanda lscpu.

```
1033_clim_antonio@sop.ase.ro:~$ lscpu
```

```
Architecture:      x86_64
CPU op-mode(s):    32-bit, 64-bit
```

Output Omitted...

Unele platforme de virtualizare necesita ca sistemul de operare oaspete sa fie compatibil cu anumite drive-uri specifice numite guest drivers sau guest agents care nu sunt incluse in distributiile standard. Acestea aduc functionalitate aditionala precum mouse support care permite mouse-ului sa comunice cu VM-ul, folder sharing intre gazda si VM, logare automata, etc. Acestea sunt necesare intrucat felul in care un hypervisor comunica cu hardware-ul fizic este diferit de felul in care SO-ul apeleaza in mod normal device-urile hardware.

Now laboratory:

A.1 Labs (Pregatire Virtual Box)

B.1 Labs (Instalare Debian)

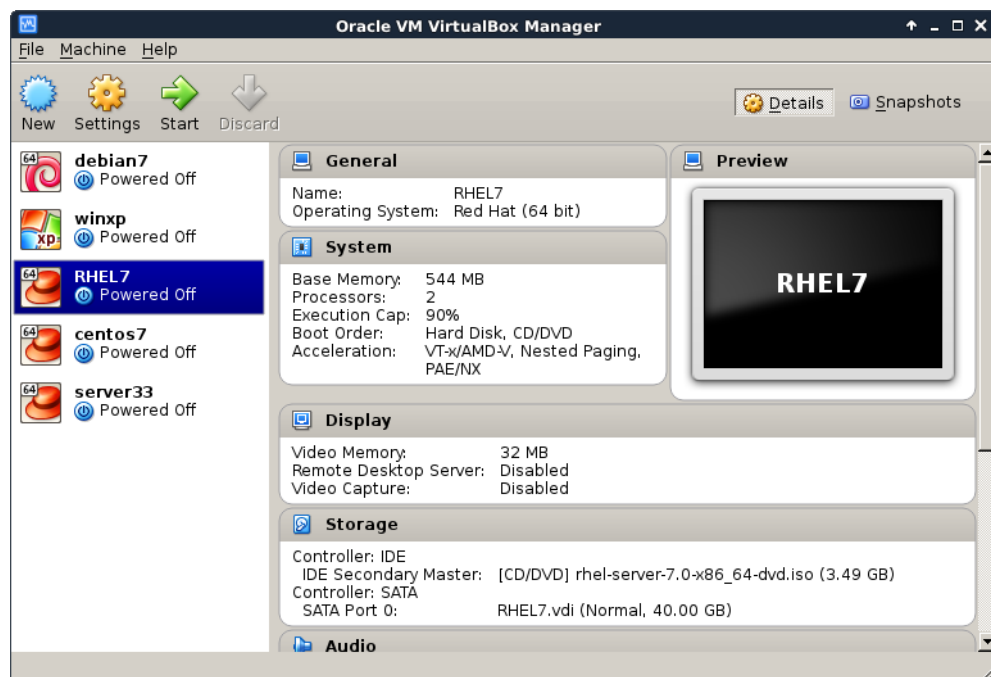
C.1 Labs (Instalare Ubuntu)

D.1 Labs (Instalare CentOS)

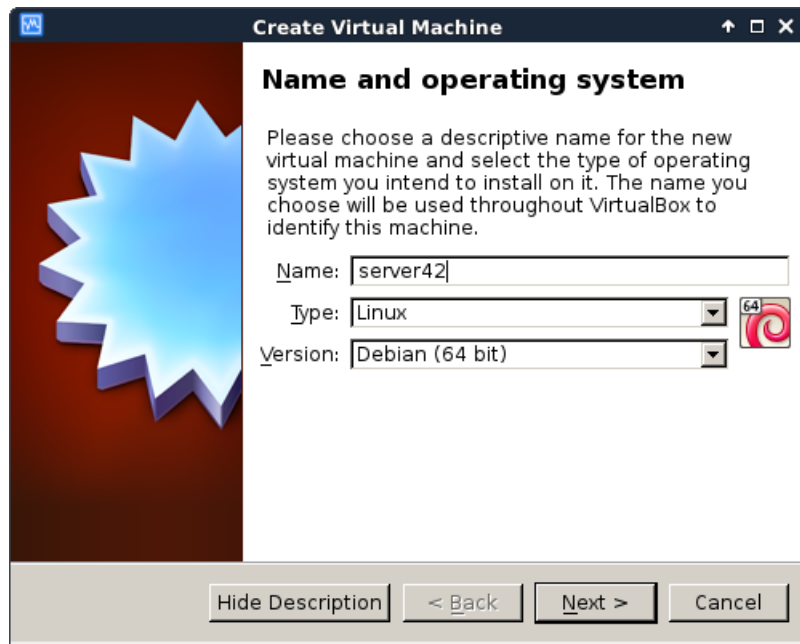
A.1 Labs (Pregatire Virtual Box):

In timpul instalarii si configurarii VirtualBox se poate descarca imaginea Linux preferata (Debian, Ubuntu, CentOS). Pregatirea instantei este la fel indiferent de varianta de Linux instalata.

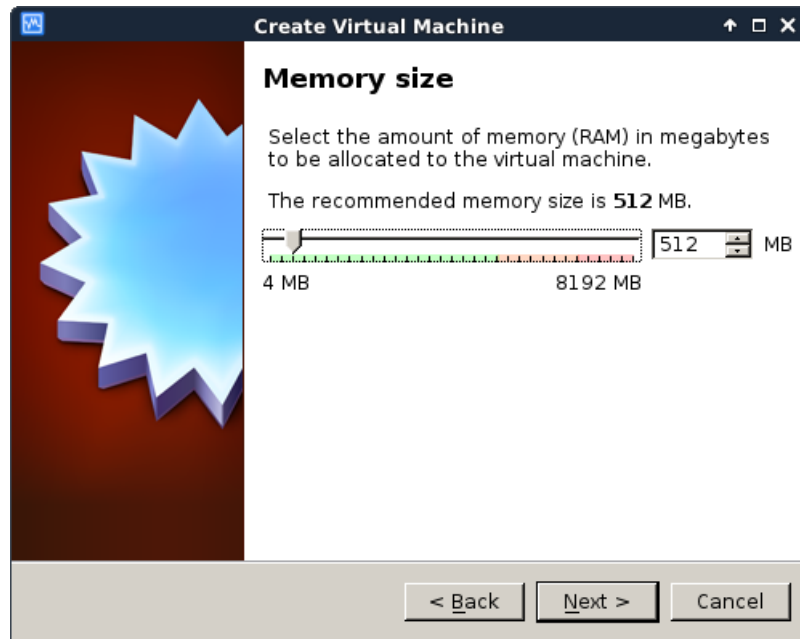
Pasul 1: Creati instanta pentru varianta de Linux preferata in VirtualBox. Apasati butonul New, iar un "vrajitor" va ajuta sa creati masina virtuala (install wizard)



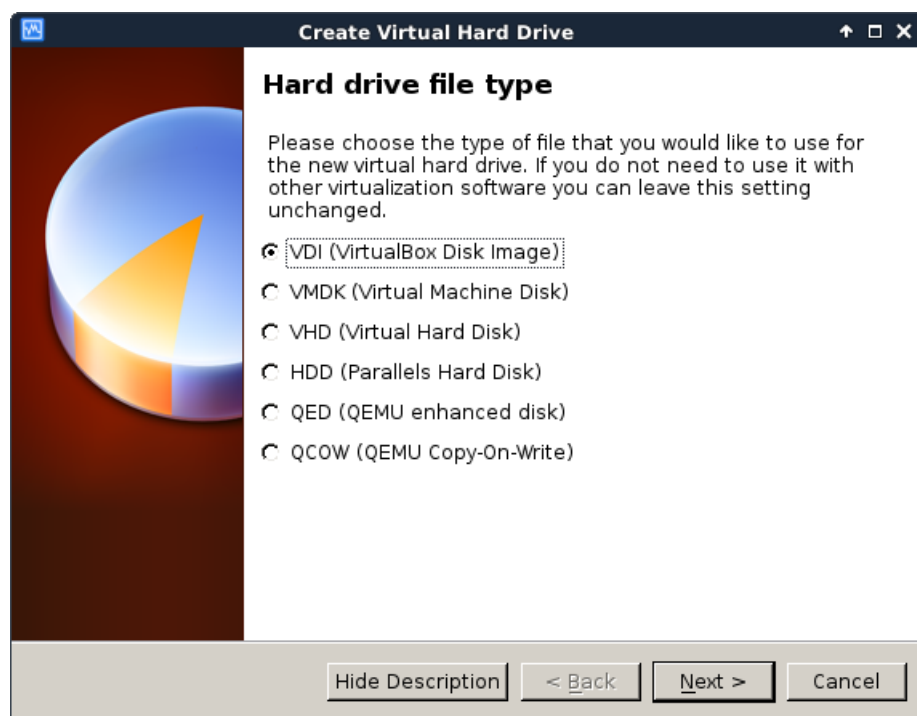
Numiti masina (**srv1030** de exemplu – 1030 reprezinta grupa de studiu), selectati Linux ca tip si alegeti versiunea de Linux pe care o instalati.

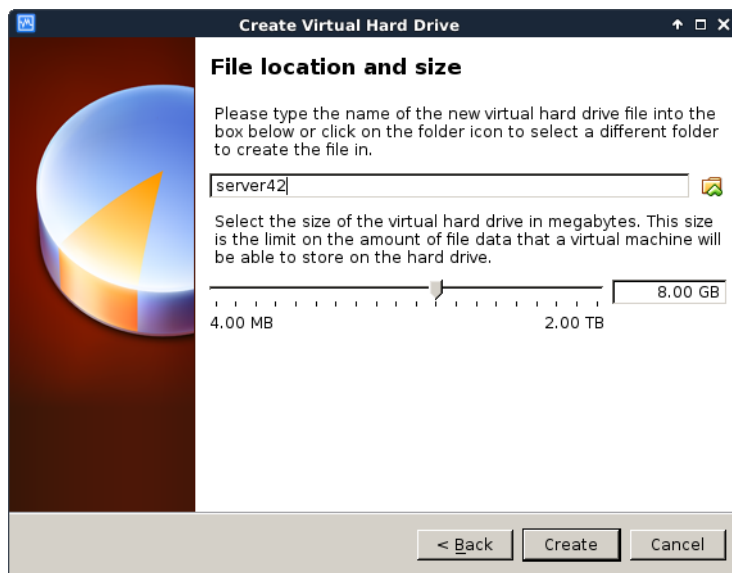


Majoritatea setarilor default sunt ok. Selectati memoria RAM necesara (512MB RAM sunt suficienti – desi puteti selecta si modifica aceasta memorie ulterior instalarii).

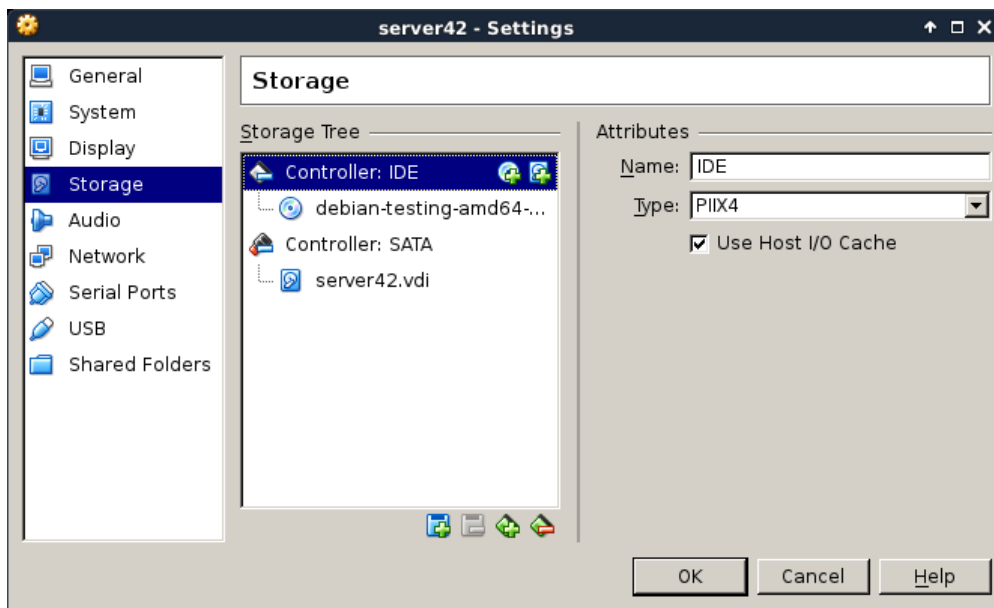


La Hard Drive file type selectati VDI (VirtualBox Disk Image), selectati **Dynamically Allocated** storage si alocati intre 8 si 10 GB masinii virtuale. Acum masina este aproape gata de instalare.

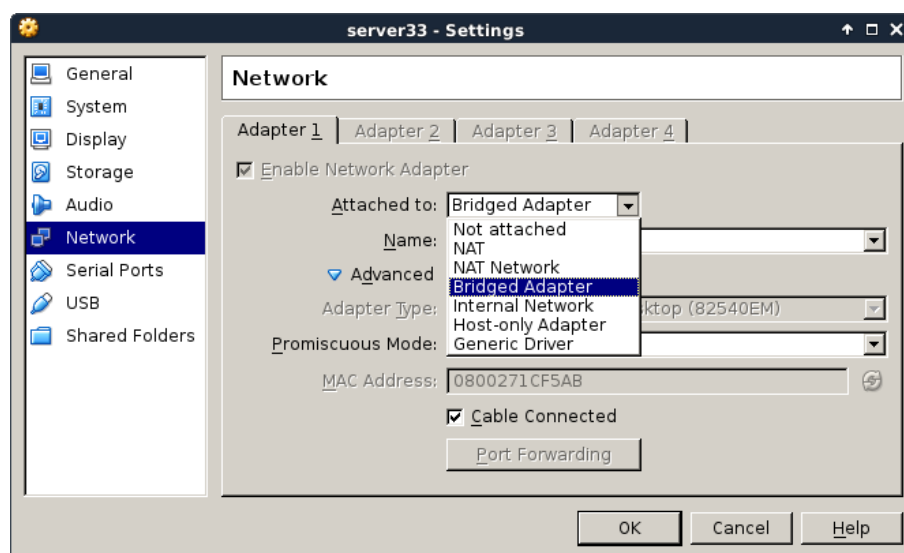




Apasati ce butonul Settings si mergeti in Tab-ul Storage pentru a putea incarca imaginea .iso descarcata.



Ulterior, mergeti la tab-ul Network si dati Enable Network Adapted si la attached to selectati Bridge Adapter in loc de NAT pentru a putea conecta masina virtuala la internet.



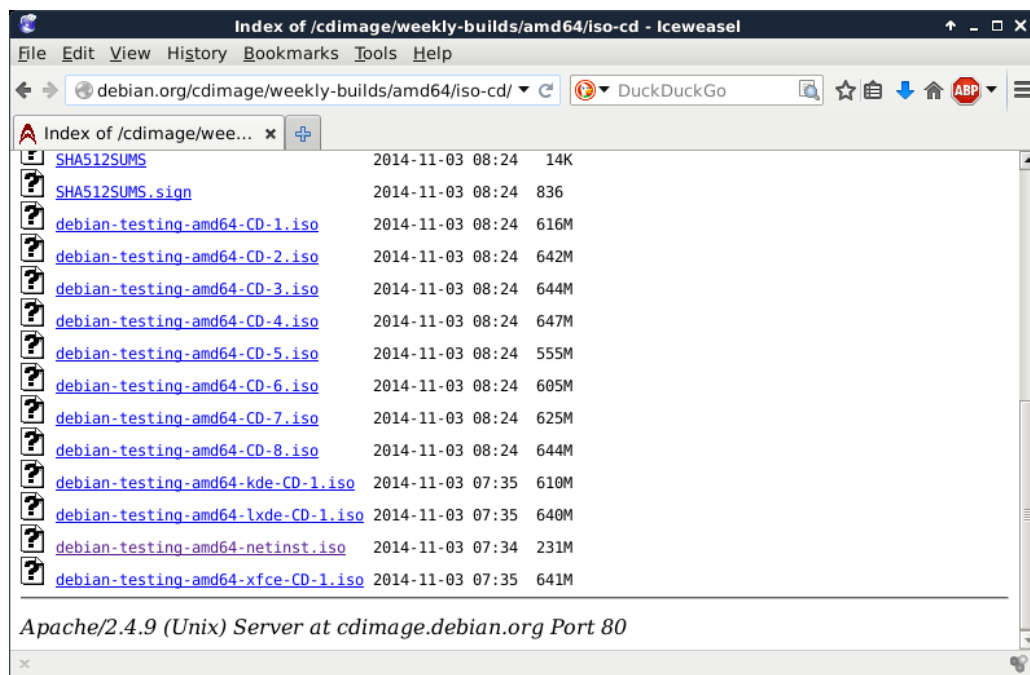
B.1 Labs (Instalare Debian)

Pasul 1. Instalare

Unul din cele mai vechi distro-uri Linux (1993) a fost distribuit gratis conform cu principiile din GNU project. Releasurile sunt denumite dupa personaje din Toy Story. In general, un release nou apare la doi ani. Limbajele folosite la scrierea distro-ului sunt ansic, cpp, python, java, xml, vhdl, asm, cobol, perl, javascript, sql, ruby, lisp, etc.

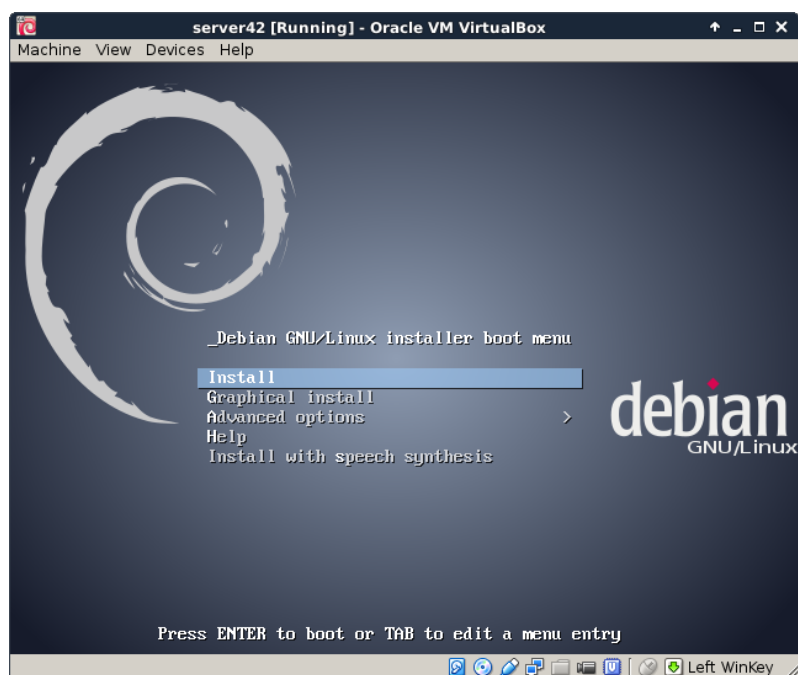
Pasul 1: Descarcati de aici. In scopul cursului a fost folosit Debian 8 – Jessie. Se poate alege oricare pachet, desi cele mai noi vor consuma mai mult RAM.

Pentru scopul cursului se va folosi o varianta fara interfata grafica (GUI – graphical user interface). In curs se va lucra exclusiv in CLI (command line interface).

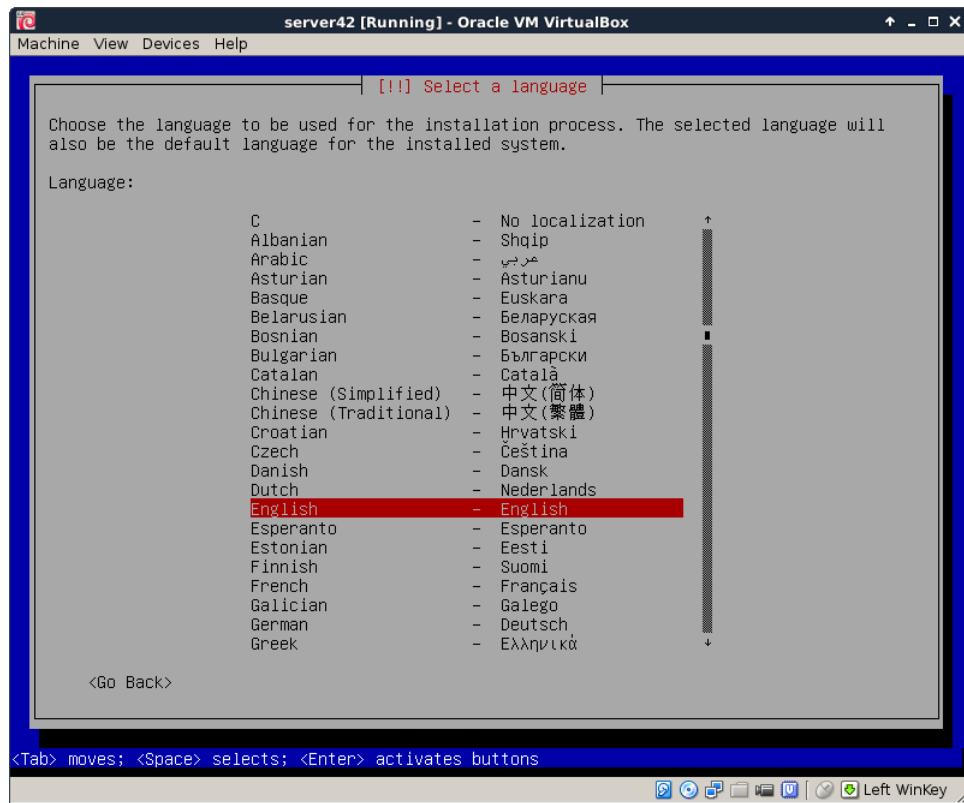


Dupa incarcarea imaginii in VirtualBox si configurarea instantei, selectati instanta si apasati butonul de start.

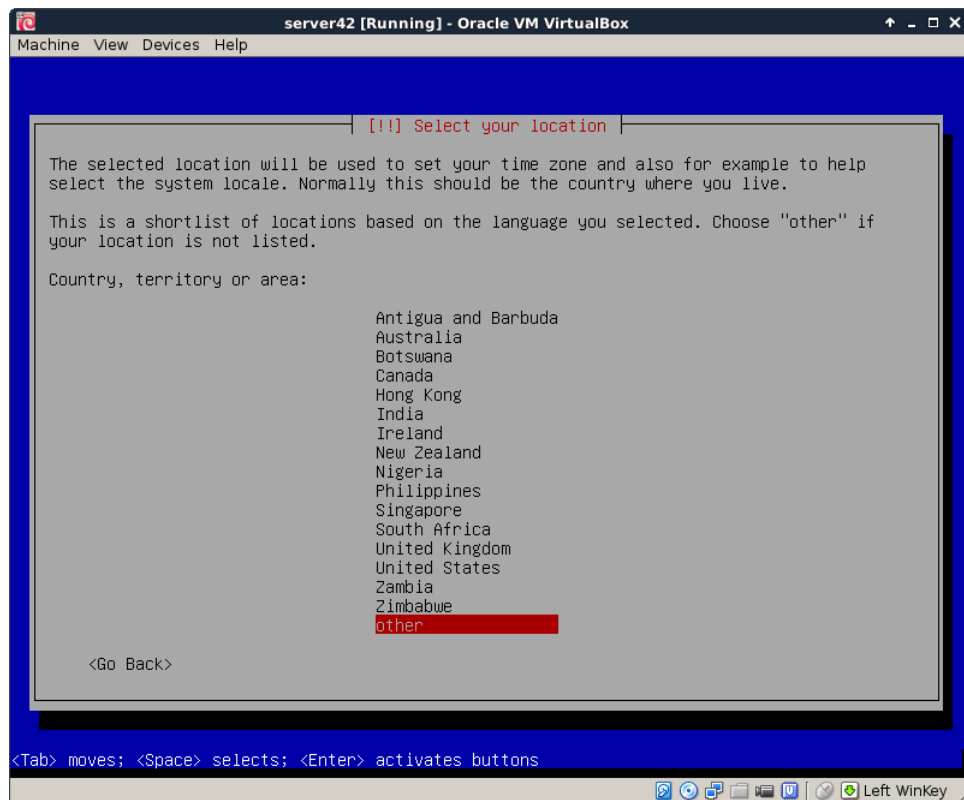
Imaginea va buta si veti vedea un ecran similar cu acesta. Selectati Install.



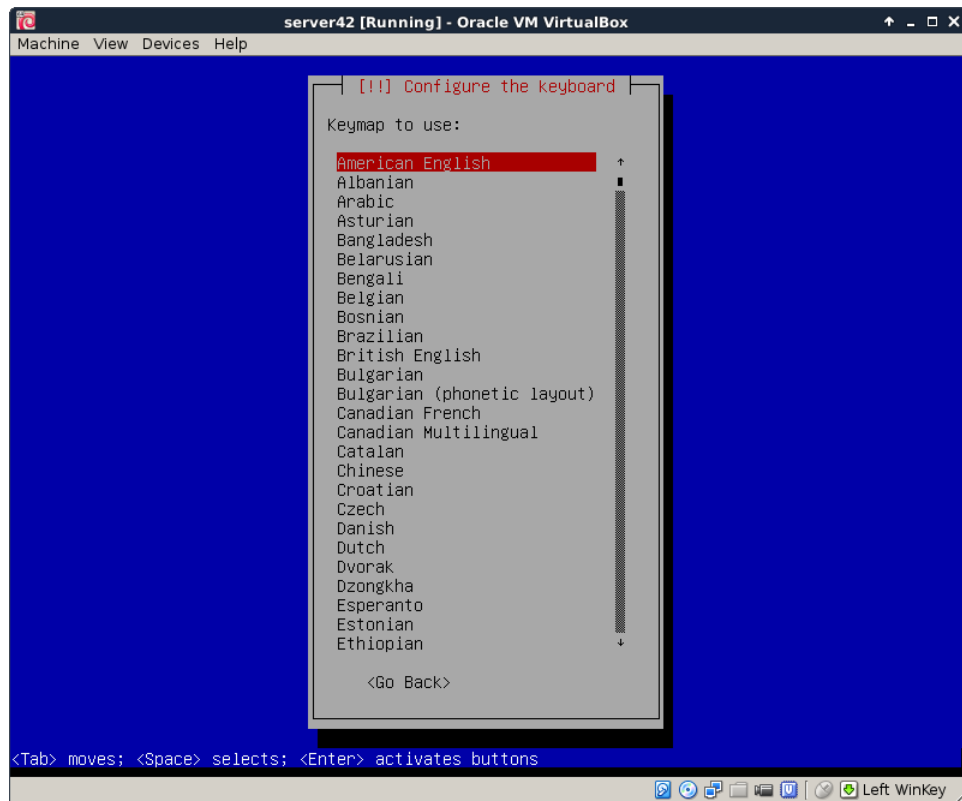
Alegeti limba pe care o veti folosi default a sistemului.



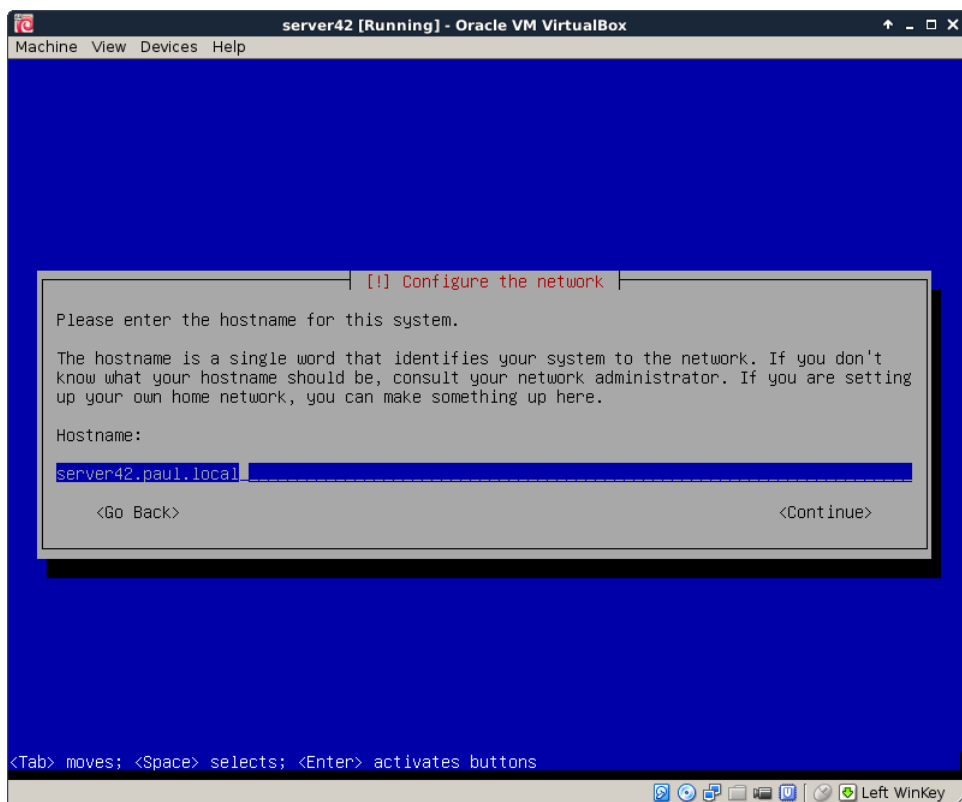
Alegeti tara. Aceasta informatie va fi folosita pentru a sugera un download mirror, si totodata va fi folosit ca informatie pentru time-zone si alte elemente necesare.



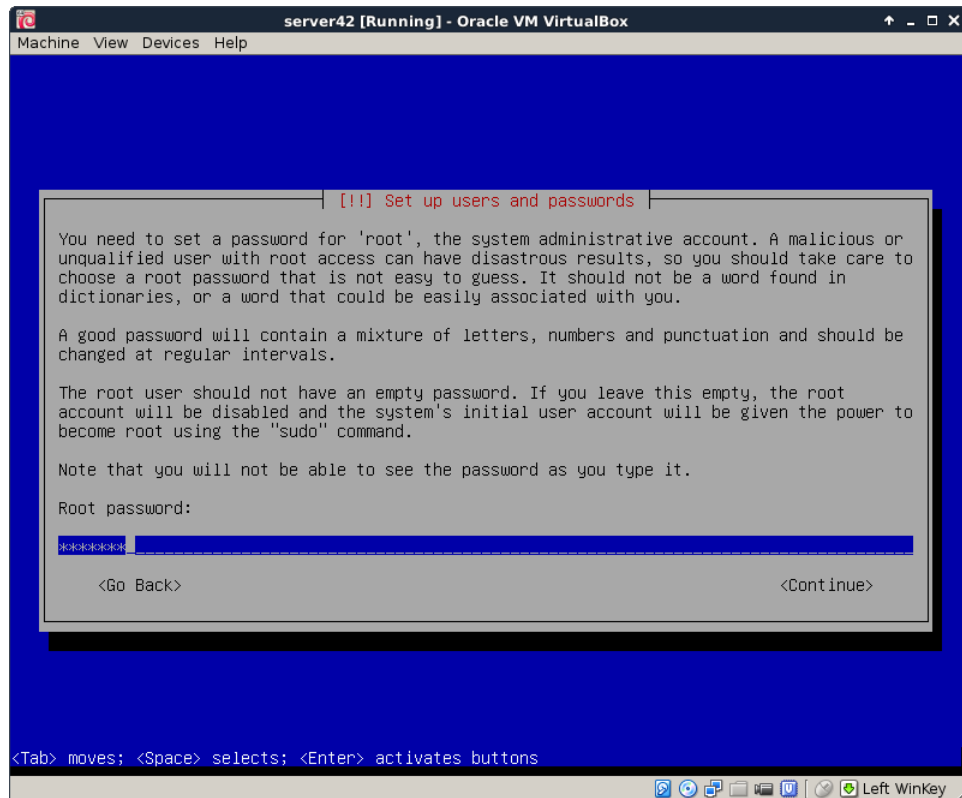
Selectati tastatura corecta (nu atat de important intrucat ne vom conecta la server prin protocolul ssh)



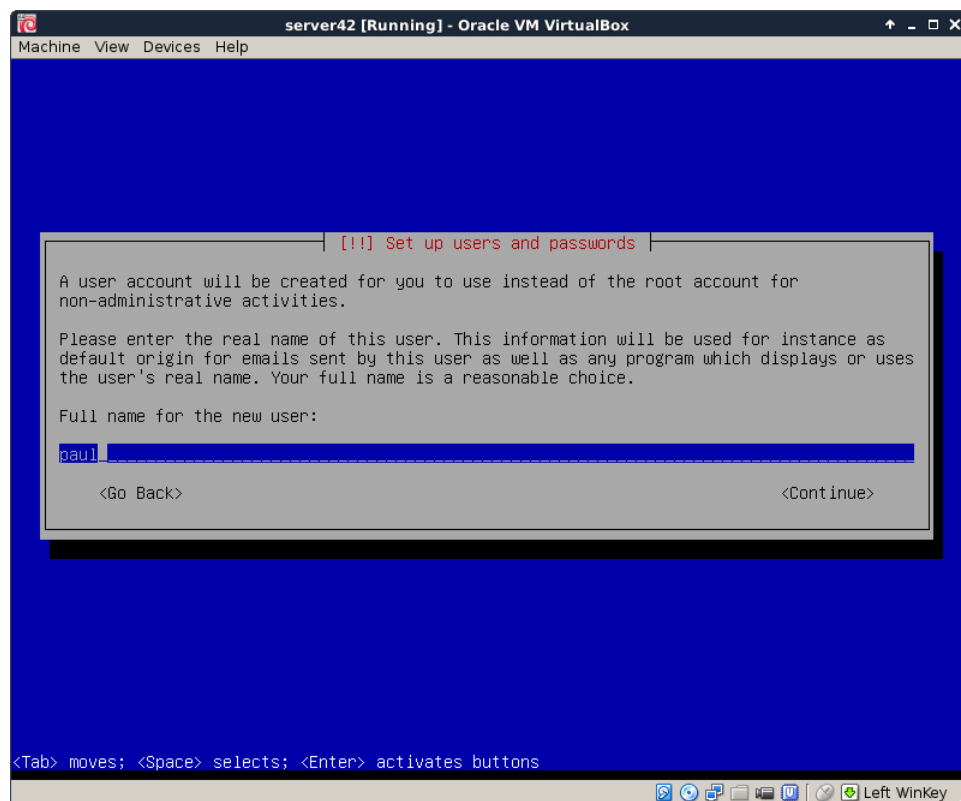
Introduceti un hostname. De exemplu: srv1130.ruscan.local. Sugestie de format: srv[grupa].[your family name].local



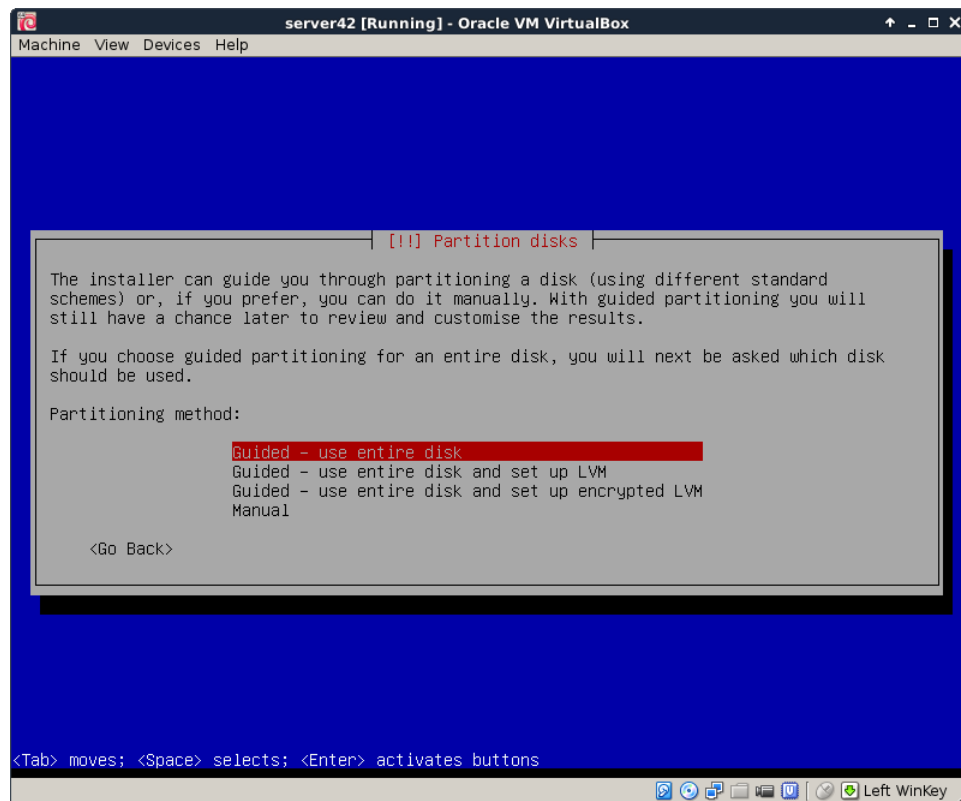
Alegeti o parola pentru root. (Acesta este singurul privileged user al sistemului Linux – nici un alt user creat ulterior nu va avea drepturi depline cu exceptia acestuia. Totdata, acesta este user-ul al carui identificator UID este 0). Pentru simplitate, puteti selecta parola student.



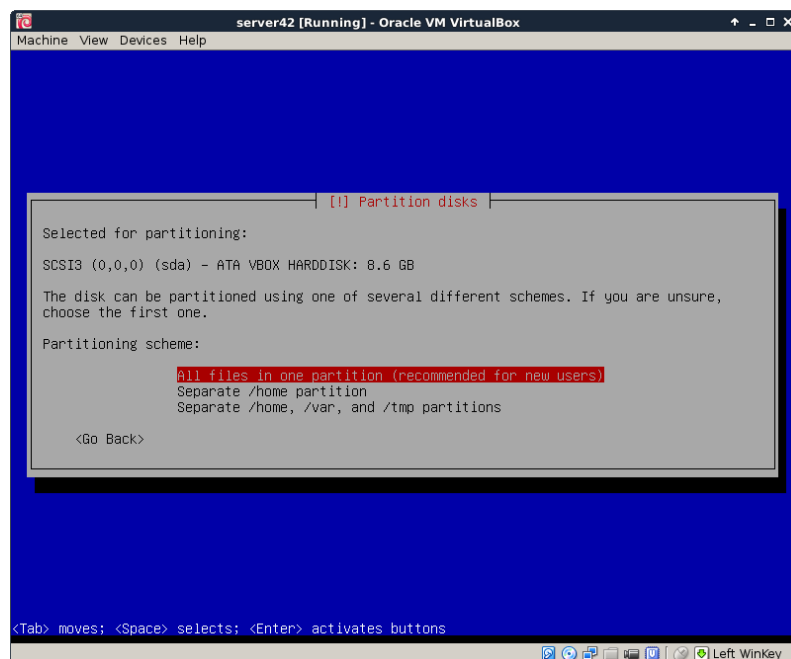
Urmatorul pas, introduceti un nume al primului user normal (neprivilegiat). Numele nu trebuie sa contina punct si trebuie sa inceapa cu o litera. Acesta va fi numele cu care va veti loga in sistem.



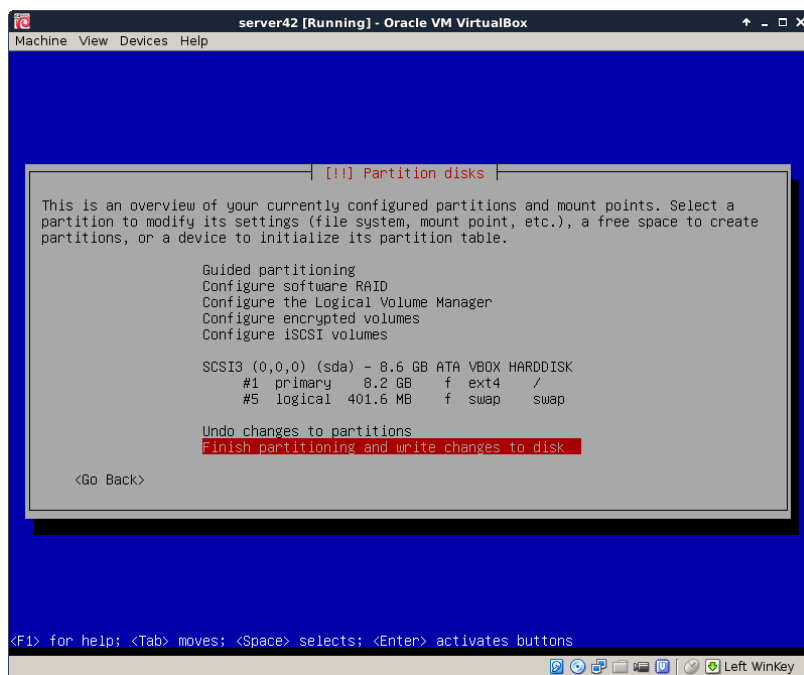
Pentru usurinta instalarii, configurati hard-urile default. Pentru setarile expert e nevoie de un nivel de cunostinte mai avansat. Selectati use entire disk. Aceasta se refera la acel disc virtual de 8-10 GB creat in VirtualBox



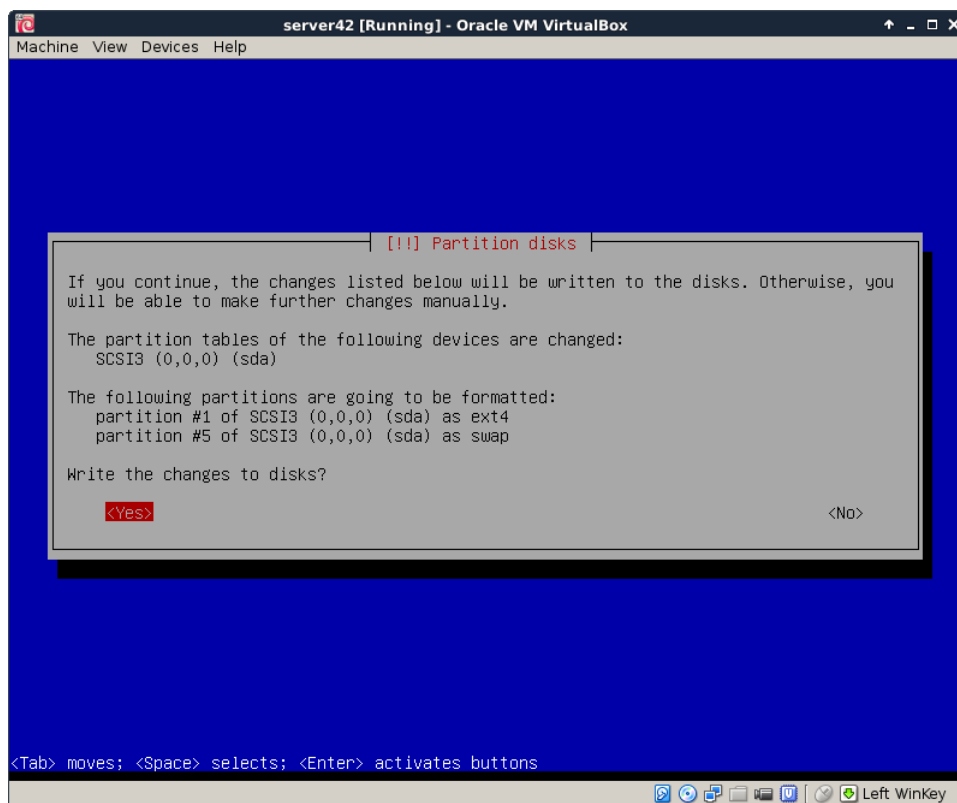
Din nou, selectati default. Alegeti alte variante de partitionare daca stiti ce faceti.



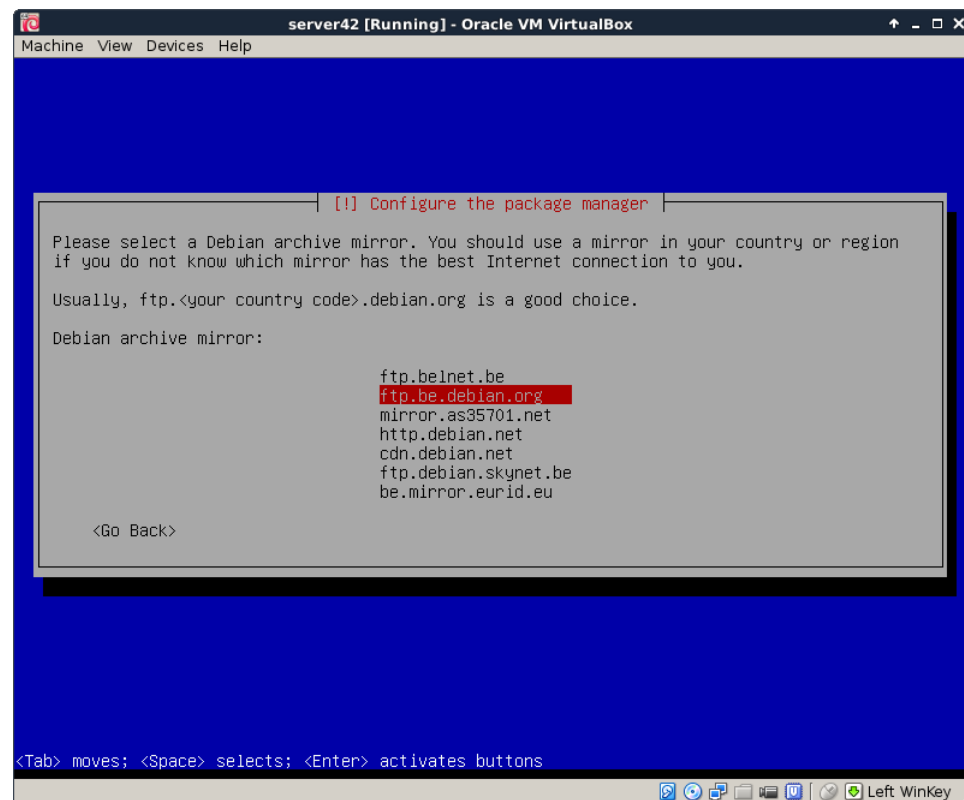
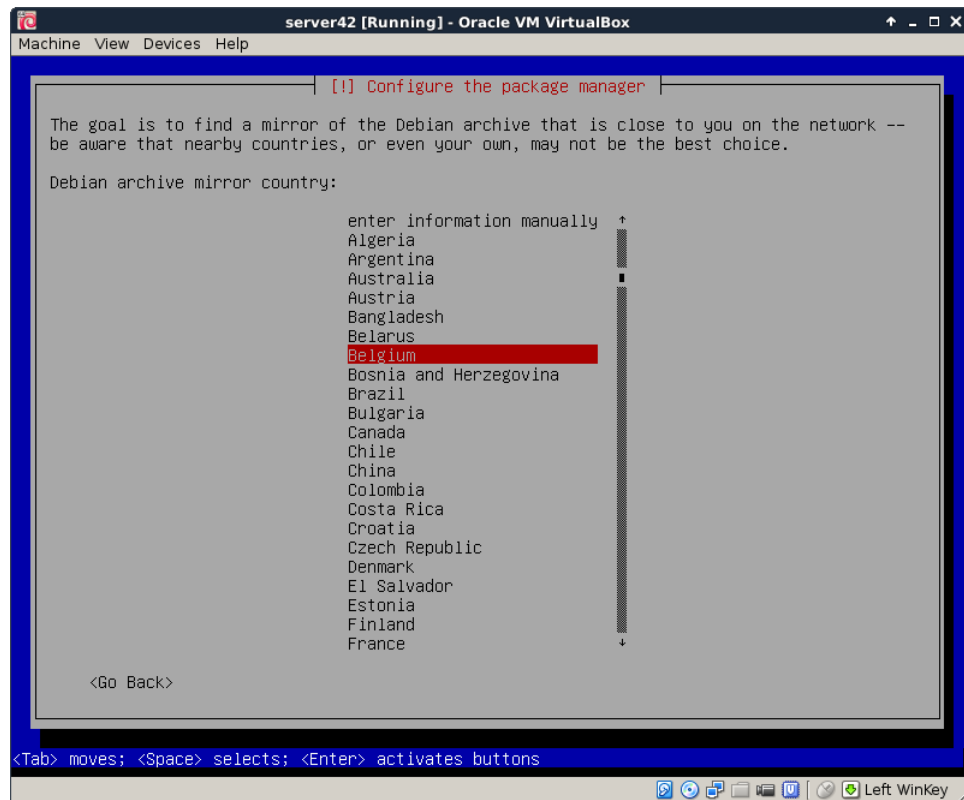
Acceptati layout-ul de partitionare.



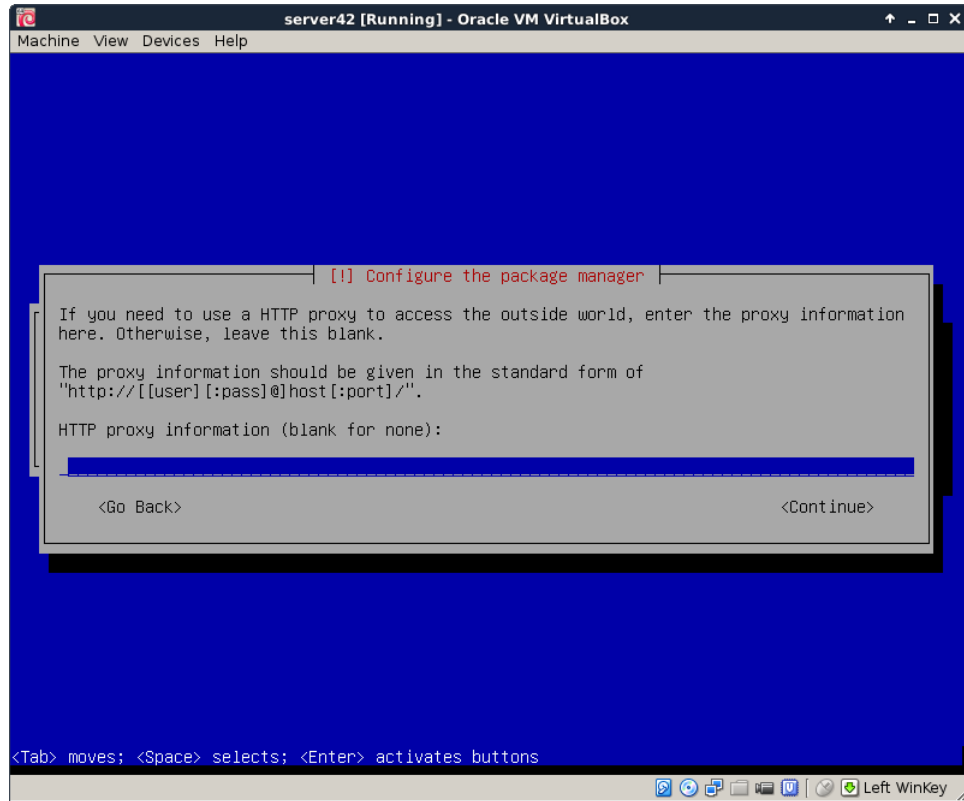
Urmatoarea fereastră reprezintă acel punct din poveste din care tragicul erou nu mai poate da înapoi. În povestea noastră asta înseamnă că toate datele vor fi șterse permanent de pe disc. Din fericire pentru noi, acest lucru se referă doar la datele de pe mașina virtuală.



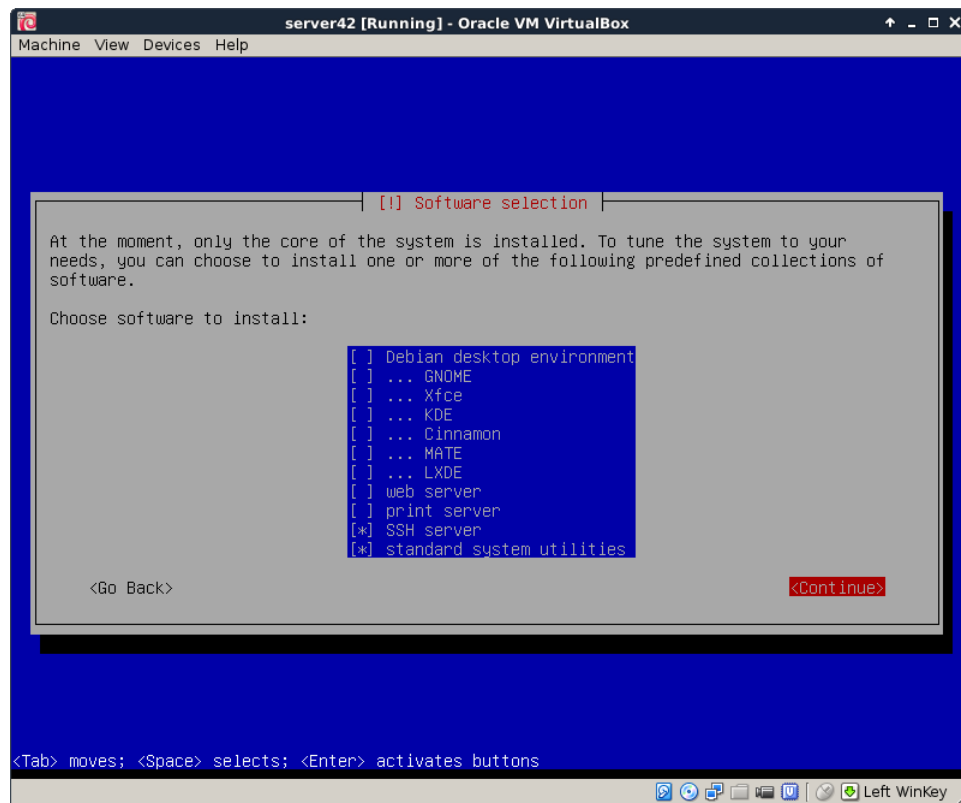
Alegeti o locatie cat mai apropiata, pentru a avea o conexiune cat mai rapida la un mirror de unde sa descarcati software aditional.



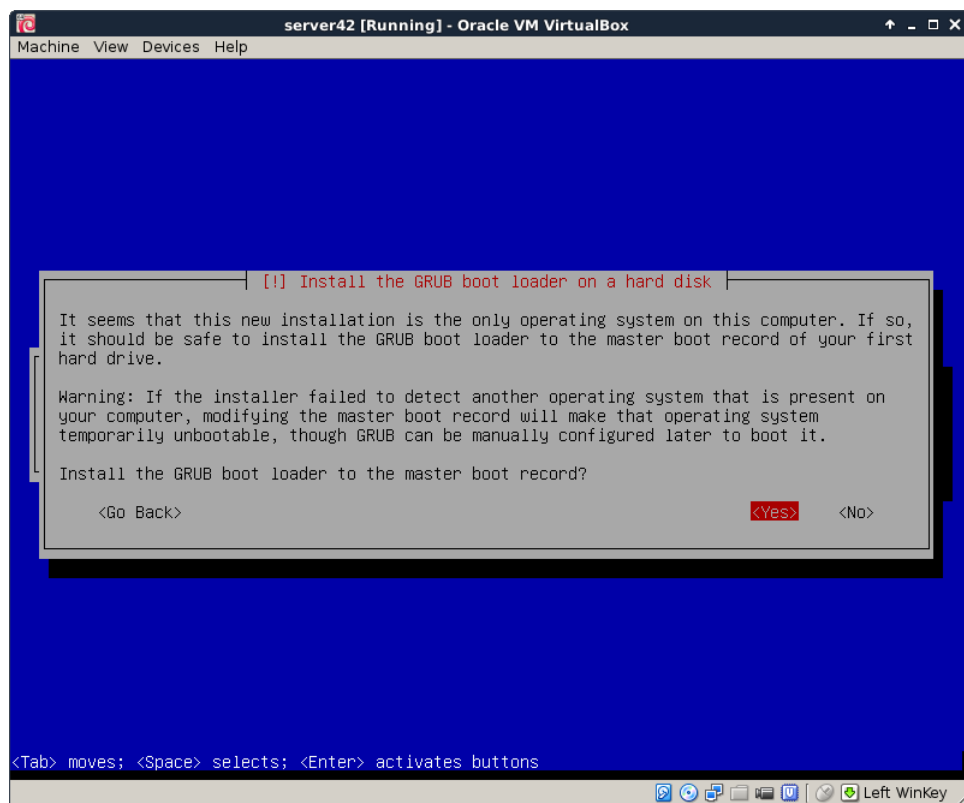
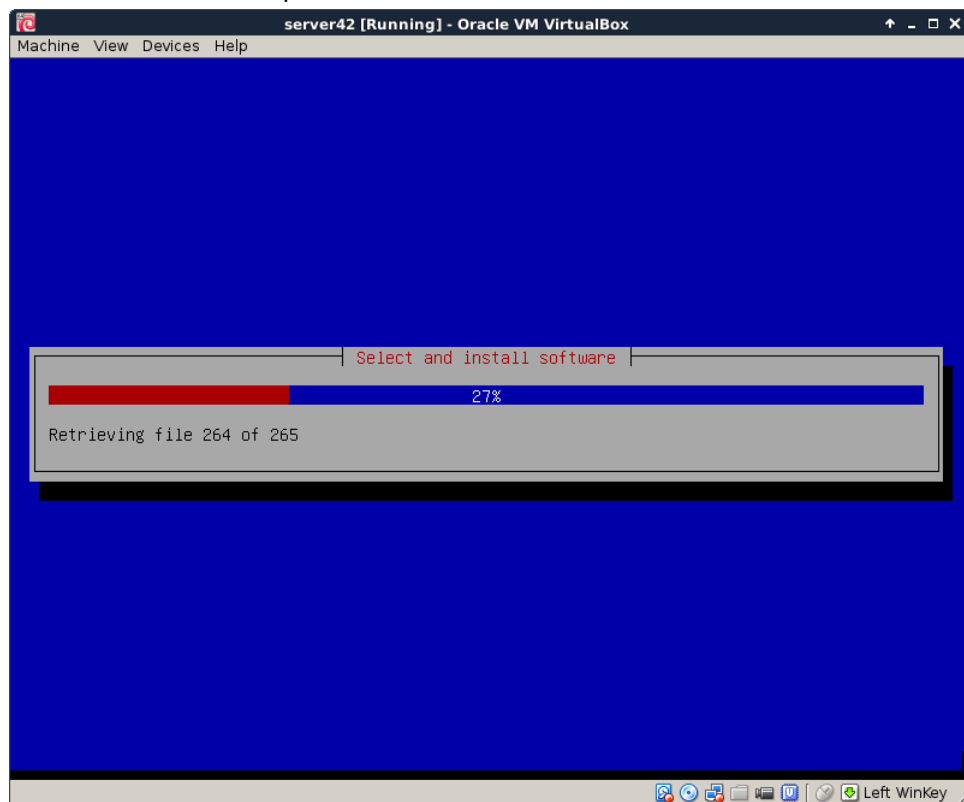
Nu completati field-ul proxy. (decat daca va aflati in spatele unui server proxy)



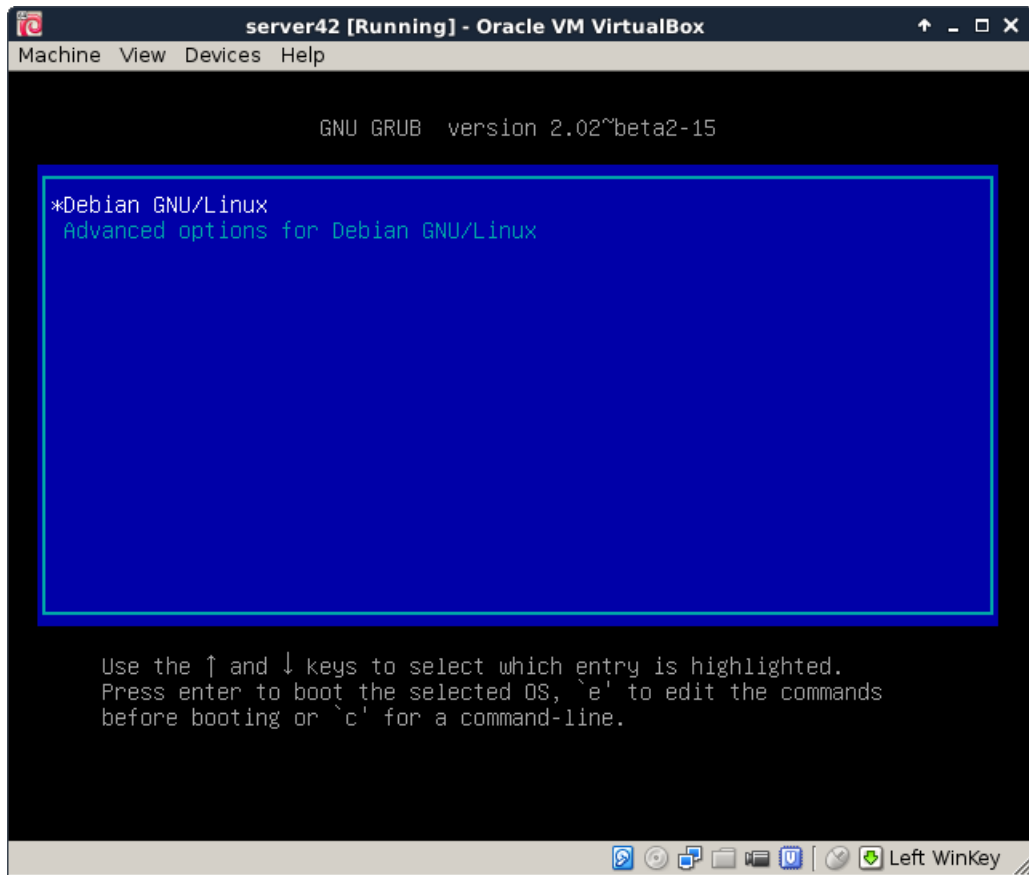
Selectati software-ul pe care sa il instalati initial. Alegeti standard system utilities si SSH server. Restul nu sunt necesare.



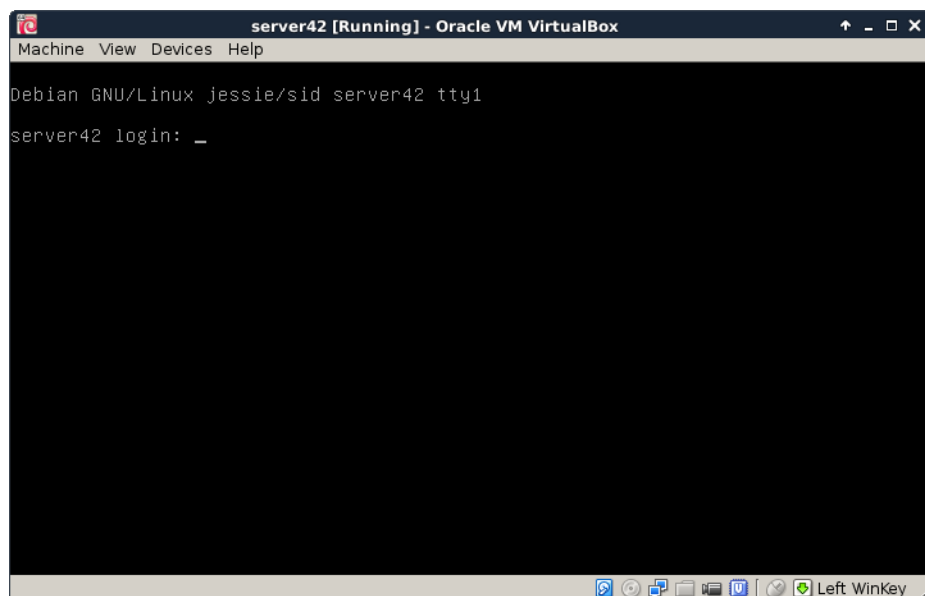
Dupa descarcare si instalare, acceptati instalarea GRUB boot loader-ului.

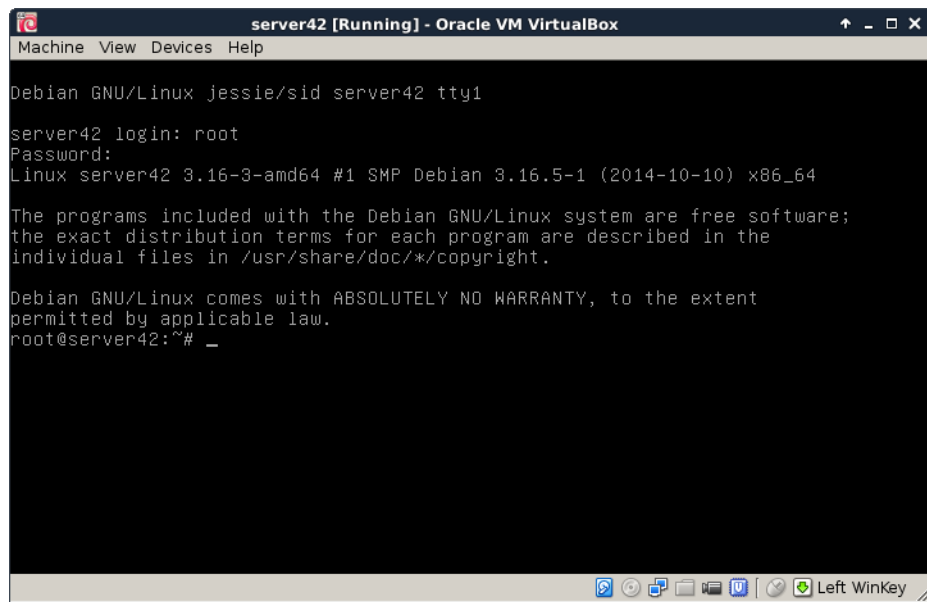


Booting for the first time shows the grub screen



Cateva secunde mai tarziu, masina va boota. Veti fi intampinati de catre bootloader. Lasati-l sa isi faca treaba si veti fi intampinati in continuare de catre un ecran care va cere username-ul si parola. Introduceti user-ul si parola setata mai sus si bine ati venit in CLI-ul Linux.





Puteti da niste comenzi simple precum pentru a va familiariza cu Linuxul.

```
root@server42:~# whoami
root      tty1      2018-11-10 18:21
root@server42:~# hostname
server42
root@server42:~# date
Mon Nov 10 18:21:56 CET 2018
```

Pasul 2 – crearea retelei in virtual box

```
root@server42:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:f5:74:cf
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fef5:74cf/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:11 errors:0 dropped:0 overruns:0 frame:0
          TX packets:19 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2352 (2.2 KiB)  TX bytes:1988 (1.9 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Pentru a facilita lucrul in masina virtuala ne putem conecta la masina virtuala prin ssh sau putty (utilizatorii Windows au optiunea de a folosi putty, utilizatorii Mac se pot conecta direct din terminalul mac cu comanda ssh). Dar pentru a face acest lucru, cativa pasi sunt necesari. Primul il reprezinta conectarea la retea si aflarea IP-ului prin care ne putem conecta la ssh. Presupunand ca configurarea retelei la bridged adapter a fost facuta inaintea instalarii atunci se poate rula comanda si in dreptul inet se poate gasi adresa ip. Portul standard de ascultare pentru ssh este 22

Inchizand aceasta interfata de retea si deschizand-o din nou, fortam Debian sa reinnoiasca adresa IP din bridged network.

```
root@server42:~# # do not run ifdown while connected over ssh!
root@server42:~# ifdown eth0
Killed old client process
Internet Systems Consortium DHCP Client 4.3.1
Copyright 2004-2014 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/08:00:27:f5:74:cf
Sending on   LPF/eth0/08:00:27:f5:74:cf
```

```
Sending on   Socket/fallback
DHCPRELEASE on eth0 to 10.0.2.2 port 67
root@server42:~# # now enable bridge in virtualbox settings
root@server42:~# ifup eth0
Internet Systems Consortium DHCP Client 4.3.1
Copyright 2004-2014 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/08:00:27:f5:74:cf
Sending on   LPF/eth0/08:00:27:f5:74:cf
Sending on   Socket/fallback
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 8
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 8
DHCPREQUEST on eth0 to 255.255.255.255 port 67
DHCPOFFER from 192.168.1.42
DHCPACK from 192.168.1.42
bound to 192.168.1.111 -- renewal in 2938 seconds.
root@server42:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 08:00:27:f5:74:cf
          inet addr:192.168.1.111  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fef5:74cf/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:15 errors:0 dropped:0 overruns:0 frame:0
          TX packets:31 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3156 (3.0 KiB)  TX bytes:3722 (3.6 KiB)

root@server42:~#
```

Ulterior, se introduc in putty username@ipul_obtinut si se apase pe connect. In cazul utilizatorilor mac se va folosi comanda ssh. Exemplu:

```
ssh ruscanvali@192.168.1.111 -p 22
```

Pentru aflarea adresei IP se mai poate folosi `ip add sh | grep inet` (prescurtat de la `ip address show`) – aceasta comanda impreuna cu comanda `grep inet` va afisa doar linia care ne intereseaza.

Aici e un exemplu de **ssh** pe computerul nou instalat. Debian 8 a revocat accesul remote cu contul root, asadar prin ssh ne vom conecta cu user normal.


```
paul@debian8:~$ ssh paul@192.168.1.111
paul@192.168.1.111's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
paul@server42:~$
paul@server42:~$ su -
Password:
root@server42:~#
```

Pas 3 – setarea hostname-ului

Intrucat acesta este setat la instalare, nu mai trebuie provizionat acum.

```
root@server42:~# hostname
server42
root@server42:~# cat /etc/hostname
server42
root@server42:~# dnsdomainname
paul.local
root@server42:~# grep server42 /etc/hosts
127.0.1.1      server42.paul.local    server42
root@server42:~#
```

Pas 4 - adaugarea unei adrese IP statice.

Acest exemplu prezinta cum se poate obtine o adresa statica prin intermediul comenzii **ifconfig** (activa pana la urmatorul reboot sau urmatorul ifdown)

```
root@server42:~# ifconfig eth0:0 10.104.33.39
```

Daca adaugati urmatoarele linii in fisier **/etc/network/interfaces** veti avea posibilitatea sa activati o adresa IP extra permanenta.

```

root@server42:~# vi /etc/network/interfaces
root@server42:~# tail -4 /etc/network/interfaces auto
eth0:0
iface eth0:0 inet static
address 10.104.33.39
netmask 255.255.0.0
root@server42:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:f5:74:cf
          inet addr:192.168.1.111  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fef5:74cf/64 Scope:Link UP
          BROADCAST RUNNING MULTICAST      MTU:1500  Metric:1
          RX packets:528 errors:0 dropped:0 overruns:0 frame:0 TX
          packets:333 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:45429 (44.3 KiB)  TX bytes:48763 (47.6 KiB)

eth0:0    Link encap:Ethernet  HWaddr 08:00:27:f5:74:cf
          inet addr:10.104.33.39  Bcast:10.255.255.255  Mask:255.0.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

root@server42:~#

```

Step 5 - Debian package management

```

root@server42:~# aptitude update
Get: 1 http://ftp.be.debian.org jessie InRelease [191 kB]
Get: 2 http://security.debian.org jessie/updates InRelease [84.1 kB]
Get: 3 http://ftp.be.debian.org jessie-updates InRelease [117 kB]
Get: 4 http://ftp.be.debian.org jessie-backports InRelease [118 kB]
Get: 5 http://security.debian.org jessie/updates/main Sources [14 B]
Get: 6 http://ftp.be.debian.org jessie/main Sources/DiffIndex [7,876 B]
... (output truncated)

```

Pentru a putea obtine cele mai noi pachete din repository-ul online folositi comanda `sudo aptitude update` si ulterior `sudo aptitude upgrade` pentru a si aplica toate acele update packetelor instalate.

```

root@server42:~# aptitude upgrade
Resolving dependencies...
The following NEW packages will be installed:
  firmware-linux-free{a} irqbalance{a} libnumal{a} linux-image-3.16.0-4-amd64{a}
The following packages will be upgraded:
  busybox file libc-bin libc6 libexpat1 libmagic1 libpaper-utils libpaper1 libsqlite3-0
  linux-image-amd64 locales multiarch-support
12 packages upgraded, 4 newly installed, 0 to remove and 0 not upgraded.
Need to get 44.9 MB of archives. After unpacking 161 MB will be used.
Do you want to continue? [Y/n/?]
... (output truncated)

```

Ulterior acestor upgrade-uri instalati via comanda **sudo aptitude install numele_pachetului** urmatoarele: mc, bc, [webmin](#), vim si tmux

```

root@server42:~# aptitude install vim tmux
The following NEW packages will be installed:
  tmux vim vim-runtime{a}

```

```

0 packages upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 6,243 kB of archives. After unpacking 29.0 MB will be used. Do
you want to continue? [Y/n/?]
Get: 1 http://ftp.be.debian.org/debian/ jessie/main tmux amd64 1.9-6 [245 kB]
Get: 2 http://ftp.be.debian.org/debian/ jessie/main vim-runtime all 2:7.4.488-1 [5,046 kB]
Get: 3 http://ftp.be.debian.org/debian/ jessie/main vim amd64 2:7.4.488-1 [952 kB]

```

C.1 Labs (Instalare Ubuntu):

- Pasul 1. Descarcati [Ubuntu](#). Pentru curs am folosit 16.04.06, mai specific [ubutntu-16.04.6-server-i386.iso](#).
- Pasul 2. In timp ce se descarca puteti configura instanta de VM in Virtual Box conform pasilor de la capitolul A.1
- Pasul 3. Urmati pasii detalati in instalarea Debian mentionata in capitolul A.2

Dupa cum am mentionat mai sus Ubuntu este o distributie Linux free bazata pe Debian. Aceasta primeste release-uri noi aproximativ la 6 luni, iar o data la doi ani primeste o versiune LTS (long-term support). Acesta a aparut din dorinta unui mogul sud-african de a avea o distributie de Linux puternica, dar care in acelasi timp sa fie mai user-friendly. Acesta a luat distributiile Debian, care deja erau populare si foarte stabile si a lucrat sa le imbunatateasca si sa le faca mai accesibile utilizatorilor normali. A creat compania Canonical Ltd al carei scop este sa dezvolte si sa ofere suport Ubuntu Linux. Asadar, in ziua de azi, Ubuntu a devenit una din cele mai beginner-friendly distributii Linux, iar producatorii de calculatoare precum Dell aleg sa vanda unitati cu Ubuntu Linux preinstalat pe ele.

D.1 Labs (Instalare CentOS)

This module is a step by step demonstration of an actual installation of **CentOS 7**.

We start by downloading an image from the internet and install **CentOS 7** as a virtual machine in **Virtualbox**. We will also do some basic configuration of this new machine like setting an **ip address** and fixing a **hostname**.

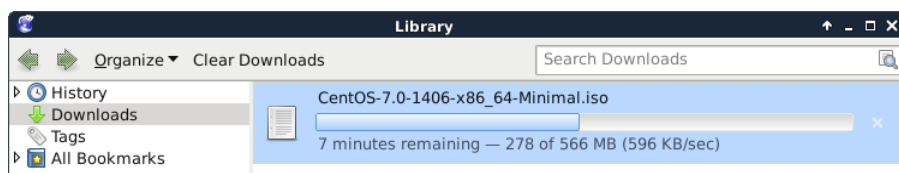
This procedure should be very similar for other versions of **CentOS**, and also for distributions like **RHEL** (Red Hat Enterprise Linux) or **Fedora**. This procedure can also be helpful if you are using another virtualization solution.

*(For our lessons and homeworks I would advise you to use a username like **1032_climantonio** (but do not worry – you can change it later)*

Pas 1 – descarcati imagine Linux

Aceasta demonstratie foloseste un computer cu **VirtualBox** pentru a instala [CentOS 7](#) ca si masina virtuala. Primul pas e descarcarea unei imagini **.iso**.

Se poate descarca un DVD full care permite instalarea offline a CentOS grafic. Dar daca mergeti la **alternative downloads** puteti gasi **CentOS 7** versiune **x86_64 minimal** iso care este mai mica ca si dimensiune.

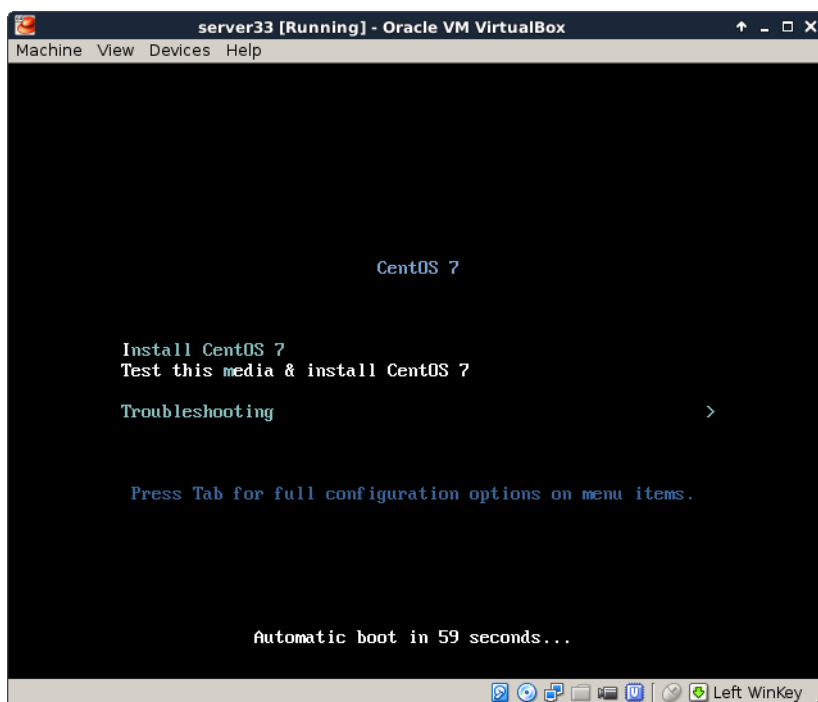


Pas 2 - Virtualbox

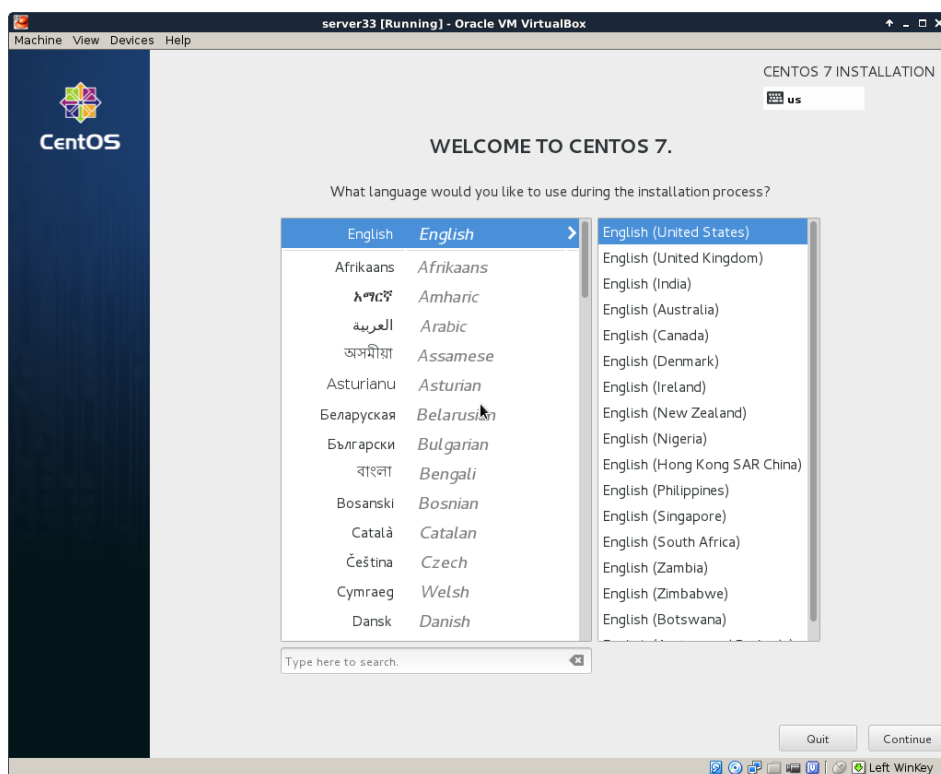
Executati pasii descrisi in **A.1 (pregatire VirtualBox)**

Step 3 – installing CentOS

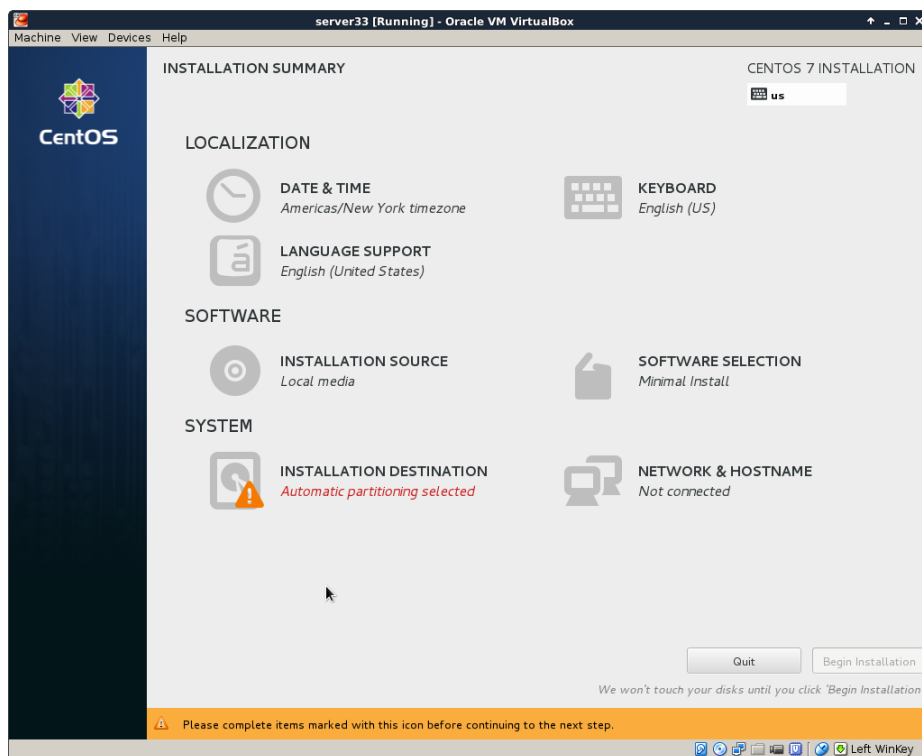
Screenshot-uri de mai jos vor arata fiecare pas de la pornirea masinii virtuale prima oara (cu fisierul .iso atasat) pana la primul login. Ar trebuie sa identificati aceasta imagine. Daca nu apare, verificati atasamentul .iso si restartati sesiunea. Selectati **Test this media and install CentOS 7**.



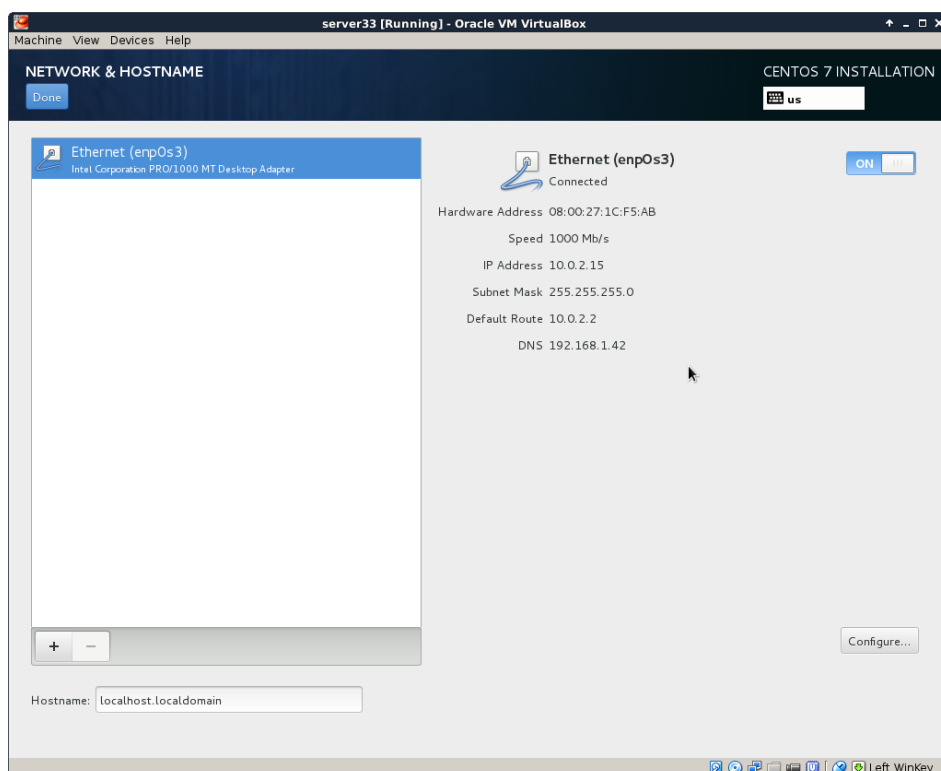
Alegeti limba de instalare **CentOS**. Recomand instalarea sistemelor de operare in limba engleza. Alegeti layout-ul de tastatura corect (in cazul asta US qwerty, la voi ar putea sa difere)



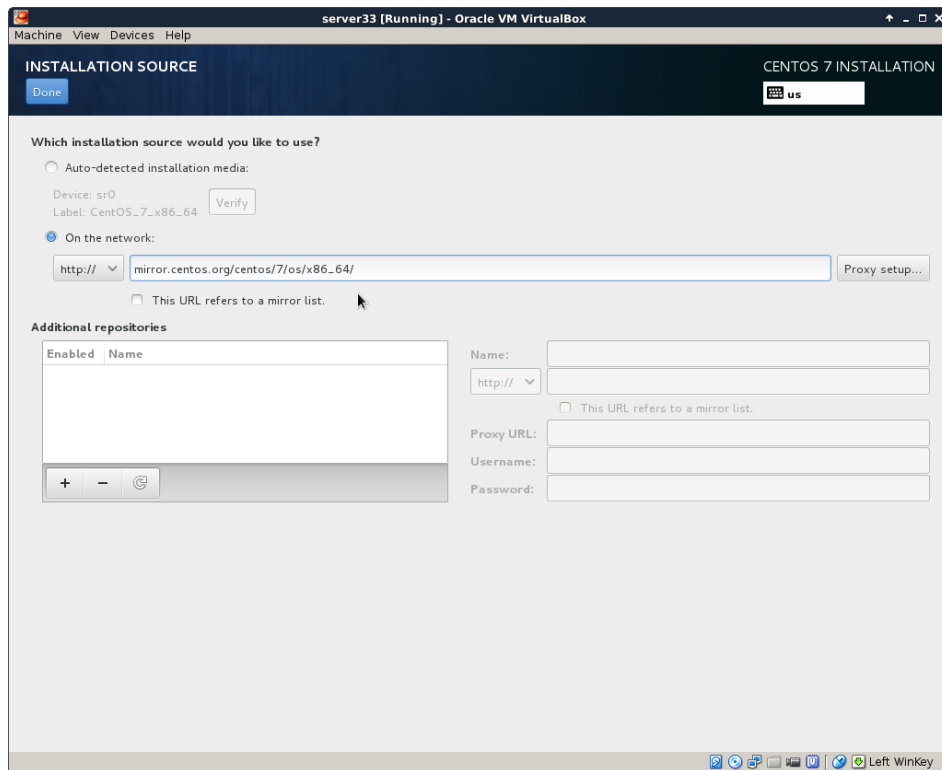
Ar trebui sa ajungeti la pagina INSTALLATION SUMMARY (cu unul sau mai multe warning-uri)



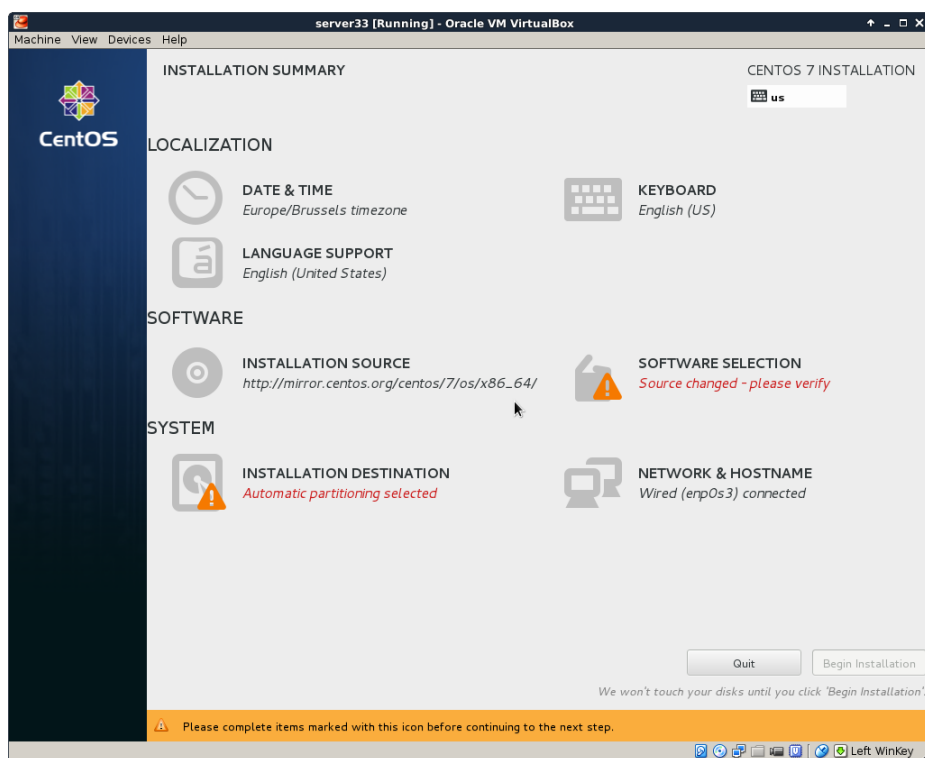
Incepeti prin a configura retea. In timpul instalarii aveam un server DHCP care rula pe 192.168.1.42, al vostru e diferit. Daca nu reusiti, incercati sa optineti ajutor (eventual de la un administrator de retea)



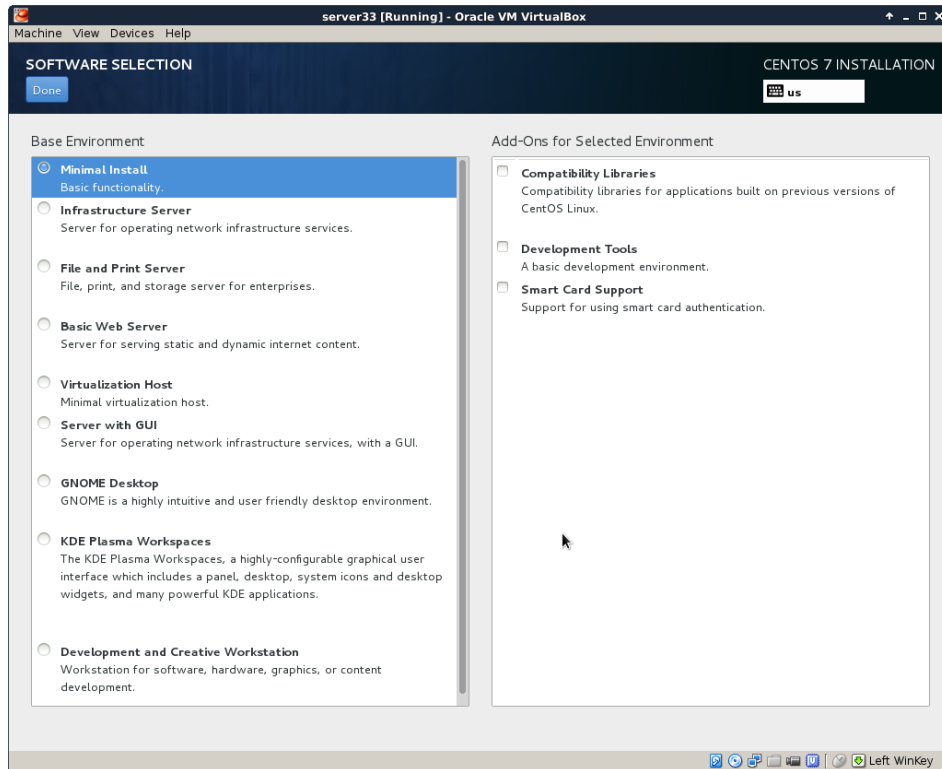
Alegeti time zone-ul corect si activati **ntp**. Alegeti un mirror apropiat. Daca nu aveti in lista un mirror local, copiat-l pe cel din screenshot (un mirror general de **CentOS**)



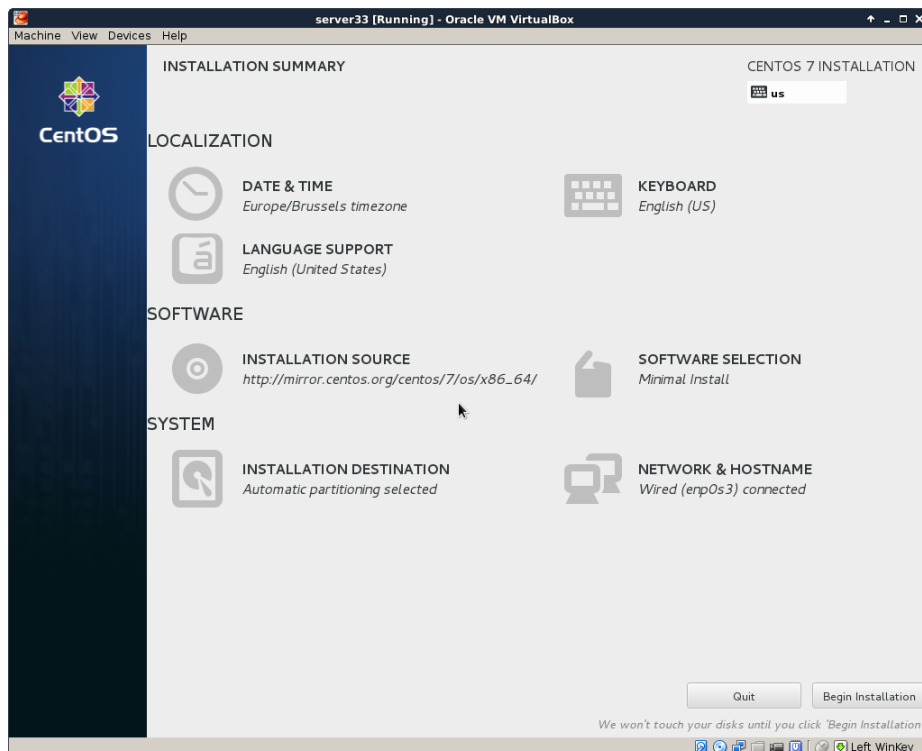
O sa dureze cateva secunde pana la validarea mirror-ului



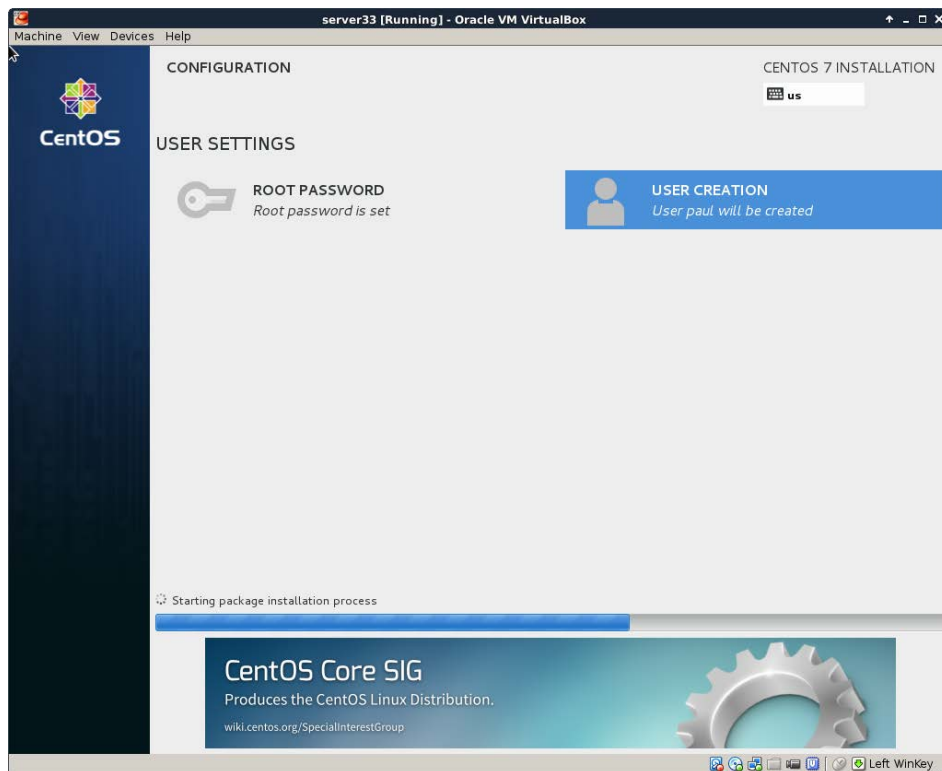
Nu selectati nici un software additional (il vom instala ulterior)



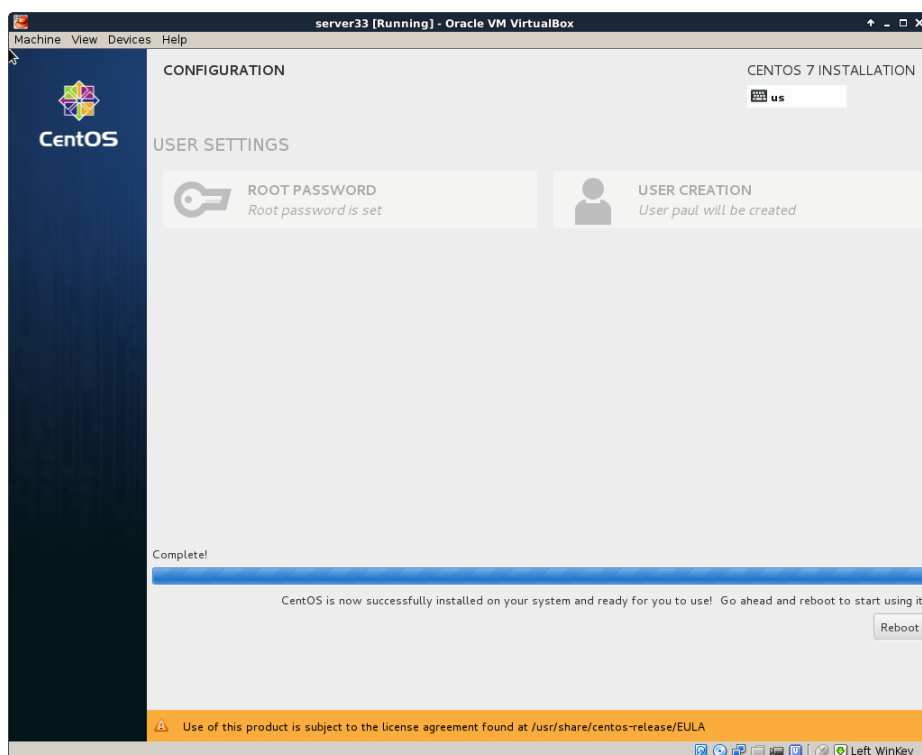
Dupa configurarea retelei, locatiei si software-ului ar trebui sa va intoarcati la pagina initiala. Asigurati-va ca nu mai sunt avertismente si ca ati respectat corect pasii pana aici.



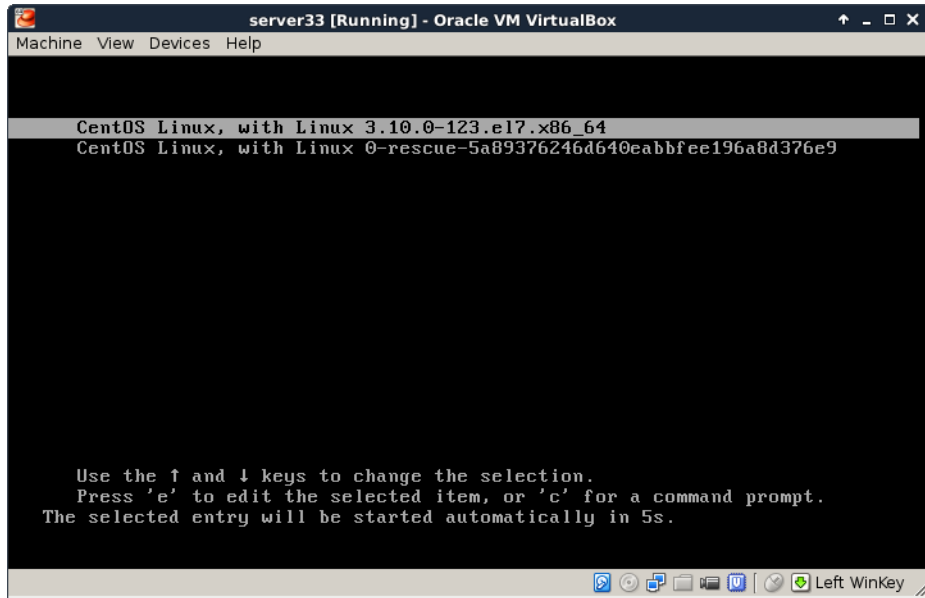
Puteti introduce **parola root** si crea un **user account** in timp ce instalarea se descarca. Acest pas este unul din cele mai lungi si poate dura de la cateva minute la o ora in cazul unei conexiuni slabe la internet.



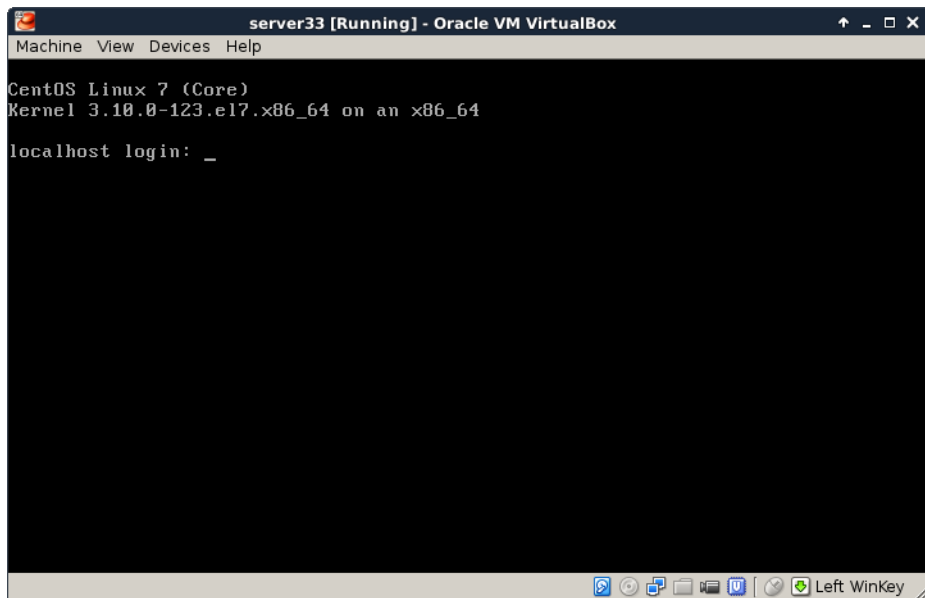
Daca ati ajuns aici, inseamna ca instalarea s-a facut cu succes. E timpul sa restartati calculatorul si sa porniti **CentOS 7** pentru prima data.



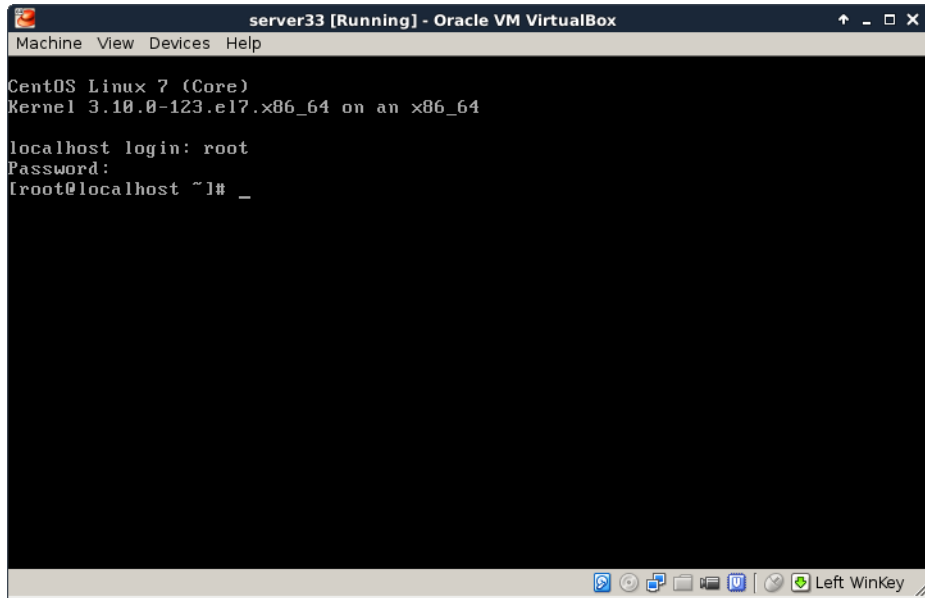
Primul ecran care apare la bootare. Nu tre sa faceti nimic, va disparea singur in cateva momente.



În curând va apărea ecranul de login. Se cheama **tty** sau **getty**. Aici puteți introduce **root** ca și username. **Procesul de login** va cere parola. (în timpul scrierii parolei nu vor apărea caractere pe ecran)



Acum sunteți logați în mașina virtuală.



De acum exemplele vor fi sub forma asta. De exemplu aici vedem 3 exemple de comenzi din CentOS7

```
[root@localhost ~]# whoami
root      pts/0      2018-11-01 22:14
[root@localhost ~]# hostname
localhost.localdomain
[root@localhost ~]# date
Sat Nov  1 22:14:37 CET 2018
```

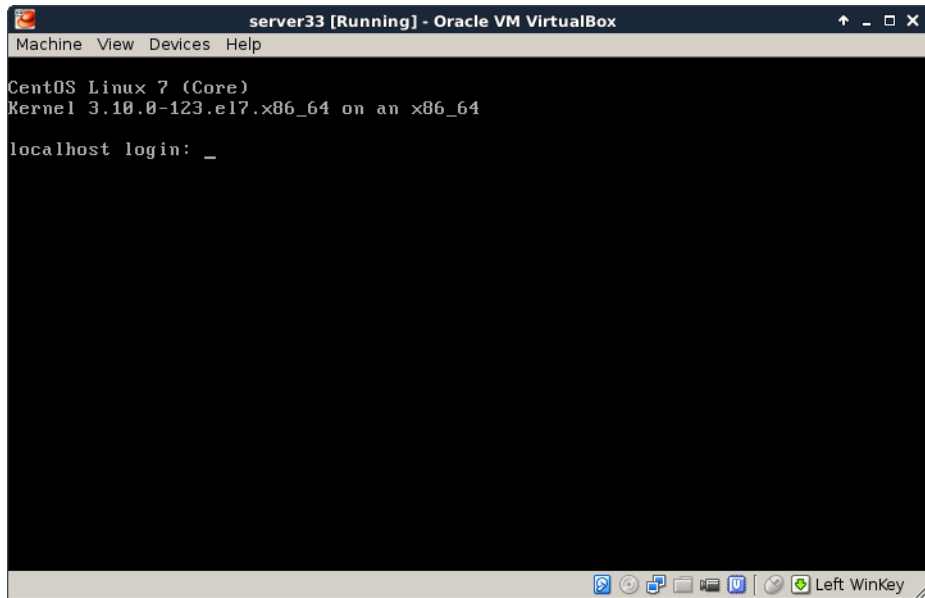
Atunci cand suntem conectati la **ssh**, aceleasi comenzi vor da urmatorul rezultat.

```
[root@localhost ~]# whoami
root      pts/0      2018-11-01 21:00 (192.168.1.35)
[root@localhost ~]# hostname
localhost.localdomain
[root@localhost ~]# date
Sat Nov  1 22:10:04 CET 2018
[root@localhost ~]#
```

If the last part is a bit too fast, take a look at the next topic **first logon**.

Pasul 4 – primul login in CentOS

Tot ce trebuie sa faci ca sa te loghezi este sa introduci credentialele in acest ecran.



Asta este una din multele variante prin care se poate accesa masina virtuala. Este nevoie de practica pentru a intelege tot, iar urmatoarele capitole vor discuta unele din metodele de conectare si niste configurari de sistem de baza.

Step 5 – setarea hostname-ului

Setarea hostname-ului implica modificarea fisierului **/etc/hostname**. Dupa cum vedeti, parametrul default este **localhost.localdomain**.

```
[root@localhost ~]# cat /etc/hostname
localhost.localdomain
```

Puteti utiliza comanda **echo server33.netsec.local > /etc/hostname** urmat de un reboot. Dar totodata, exista si o metoda specifica **CentOS 7** de a face modificarea.

```
[root@localhost ~]# nmtui
```

Comanda de mai sus va deschide un meniu unde aveti optiunea **set system hostname**. Folosind **nmtui**, acesta va modifica pentru voi fisierul **/etc/hostname**.

```
[root@localhost ~]# cat /etc/hostname
server33.netsec.local
[root@localhost ~]# hostname
server33.netsec.local
[root@localhost ~]# dnsdomainname
netsec.local
```

Documentatia de pe **centos.org** si **docs.redhat.com** mai dau ca varianta urmatoarea metoda.

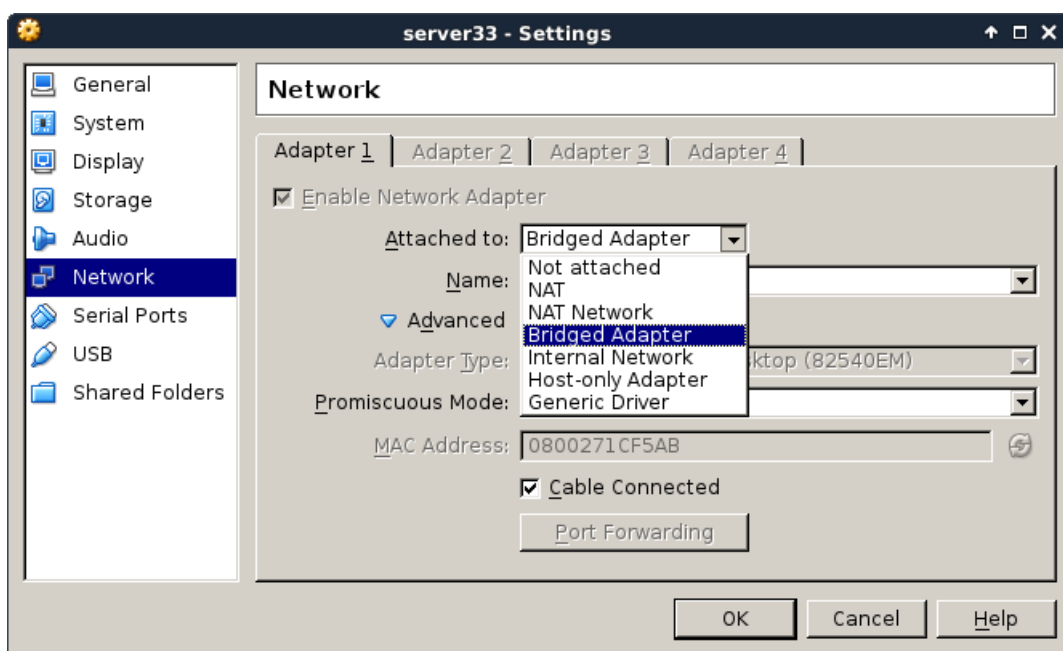
```
[root@localhost ~]# systemctl restart systemd-hostnamed
```

Pasul 7 – Virtualbox network interface

Default, VirtualBox ca conecta masina virtuala printr-o interface **nat**. Aceasta va afisa un ip similar cu 10.0.2.15. In cazul in care la configurarea VirtualBox-ului ati setat reseaua la Bridged Adapter acesti pasi nu mai sunt necesari.

```
[root@server33 ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast s\
tate UP qlen 1000
    link/ether 08:00:27:1c:f5:ab brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 86399sec preferred_lft 86399sec
    inet6 fe80::a00:27ff:fe1c:f5ab/64 scope link
        valid_lft forever preferred_lft forever
```

Puteti muta conexiunea din Settings -> Network -> attached to -> bridged adapter, si astfel se va crea o legatura ca si cum masina virtuala e conectata direct la retea locala, asadar primind un IP de la serverul DHCP real.



Daca faceti aceasta modificare in timp ce VM-ul ruleaza trebuie sa executati urmatoarele comenzi:

```
[root@server33 ~]# systemctl restart network
[root@server33 ~]# ip a s dev enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast s\
tate UP qlen 1000
    link/ether 08:00:27:1c:f5:ab brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.110/24 brd 192.168.1.255 scope global dynamic enp0s3
        valid_lft 7199sec preferred_lft 7199sec
    inet6 fe80::a00:27ff:fe1c:f5ab/64 scope link
        valid_lft forever preferred_lft forever
[root@server33 ~]#
```

Pasul 7 – configurarea retelei in CentOS

Cel mai nou mod de a modifica setari de retea este prin intermediul utilitarului **nmtui**. Dar, daca vrei sa modifichi fisierele de configurare din **/etc/sysconfig/network-scripts** manual atunci trebuie intai verificat si dezactivat NetworkManager de pe interfata respectiva. Pentru a verifica daca vreo interfata este controlata

de NetworkManager se foloseste comanda **nmcli**.(connected inseamna ca este controlata de NM)

```
[root@server33 ~]# nmcli dev status
DEVICE   TYPE      STATE      CONNECTION
enp0s3   ethernet  connected  enp0s3
lo       loopback  unmanaged  --
```

Dezactivati **NetworkManager** pe interfata (enp0s3 in cazul asta)

```
echo 'NM_CONTROLLED=no' >> /etc/sysconfig/network-scripts/ifcfg-enp0s3
```

Restartati reseaua fara a fi nevoie de reboot:

```
[root@server33 ~]# systemctl restart network
```

In **CentOS** nu veti mai folosi **ifconfig**, ci **ip a s** in schimb (de la ip address show)

```
[root@server33 ~]# ip a s dev enp0s3 | grep inet
    inet 192.168.1.110/24 brd 192.168.1.255 scope global dynamic enp0s3
    inet6 fe80::a00:27ff:fe1c:f5ab/64 scope link
[root@server33 ~]#
```

Pasul 7 – Adaugarea unei adrese IP statice in CentOS

Acest exemplu demonstreaza cum se poate adauga o adresa ip statica.

```
[root@server33 ~]# nmtui edit enp0s3
```

In aceasta interfata, lasati configurarea IPv4 pe <Automatic> si adaugati o adresa IP ca in exemplul de mai jos.

```
IPv4 CONFIGURATION <Automatic> <Hide>
Addresses 10.104.33.32/16 <Remove>
```

La iesirea din **nmtui** restartati reseaua si verificati ip-ul cu comanda **ip a s**

```
[root@server33 ~]# systemctl restart network
```

```
[root@server33 ~]# ip a s dev enp0s3 | grep inet
    inet 192.168.1.110/24 brd 192.168.1.255 scope global dynamic enp0s3
    inet 10.104.33.32/16 brd 10.104.255.255 scope global enp0s3
    inet6 fe80::a00:27ff:fe1c:f5ab/64 scope link
[root@server33 ~]#
```

Pasul 8 – CentOS package management

Chiar si cu instalarea din retea, **CentOS 7** nu a instalat ultima varianta a unora din pachete. Din fericire exista o comanda care se poate executa ca si root. Puteti folosi **yum install** pentru a instala unul sau mai multe pachete. Nu uitati sa folositi **yum update** din cand in cand.

```
[root@server33 ~]# yum update
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
* base: centos.weepeetelecom.be
* extras: centos.weepeetelecom.be
* updates: centos.weepeetelecom.be
Resolving Dependencies
--> Running transaction check
---> Package NetworkManager.x86_64 1:0.9.9.1-13.git20140326.4dba720.el7 \
will be updated
(output truncated)
```

```
[root@server33 ~]# yum update -y && yum install vim -y
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
* base: centos.weepeetelecom.be
(output truncated)
```

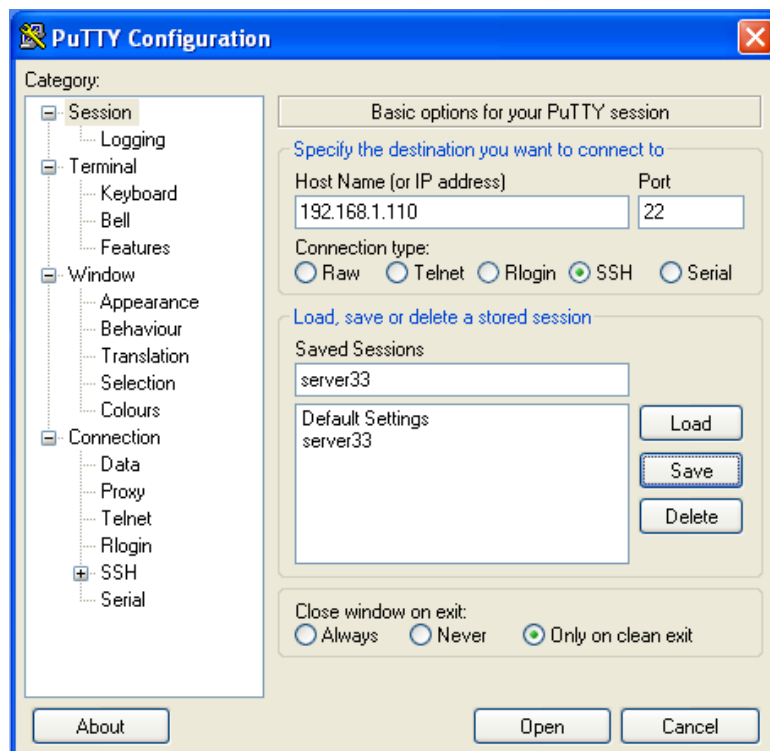
Pasul 9 – Conectarea la vm de pe sistemul gazda (Linux & MacOS X)

Deschideti un terminal nou pe Linux sau pe MacOSX si folositi comanda **ssh** pentru a va putea loga pe masina virtuala. In cazul Mac, este necesara si optiunea -p 22 (pentru a selecta portul 22). Comanda va fi de forma `ssh username@ipaddress -p 22`.

```
paul@debian8:~$ ssh root@192.168.1.110
root@192.168.1.110's password:
Last login: Sun Nov  2 11:53:57 2014
[root@server33 ~]# hostname
server33.netsec.local
[root@server33 ~]#
```

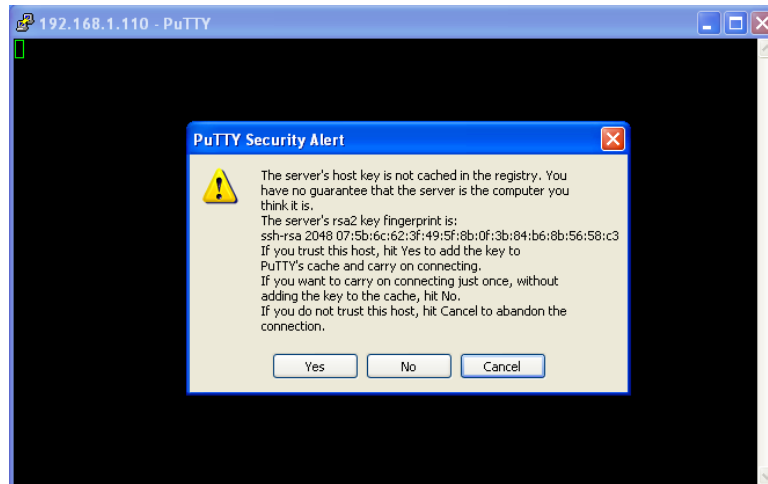
Pasul 10 – Conectarea la vm de pe sistemul gazda (MS Windows)

In Windows nu exista nativ ssh instalat, dar puteti descarca [putty.exe](#) apasand pe numele lui cu ctrl sau cu o cautare pe google. Folositi **putty** ca in exemplu de mai jos. (adresa IP a fost salvata dandu-i numele 'server33' si apasand butonul 'save').



La prima conectare veti primi un mesaj daca acceptati inregistrarea cheii de securitate sau nu. Apasati da

pentru a o salva in cache.



Introduceti userul (sau root) si parola corecta. Nu va aparea nici un caracter pe ecran in timpul scrierii parolei.

```
root@server33:~  
login as: root  
root@192.168.1.110's password:  
Last login: Sun Nov  2 12:07:49 2014 from 192.168.1.35  
[root@server33 ~]#  
[root@server33 ~]# hostname  
server33.netsec.local  
[root@server33 ~]#
```

Acum: ONLINE TEST din acest material