

### 3.1 Using the Drac

Problem 1: I couldn't connect to the iDRAC website ( because your connection is not secured ) so I added an exception.

Problem 2: I cant Launch Virtual Console on iDRAC6 because I couldn't open the .jnlp file. so I installed icedtea-netx but also it dint work. so I went to the server room and installed ubuntu manually. After that, I enabled the SSH remote access so I can easily access it from my pc and continue through the remaining questions.

#### Question 1:

a- What is a DRAC? Drac is an abbreviation for Dell Remote Access Controller. It is a complete remote management system. It support the user with all the mandatory tools to monitor, troubleshoot and even repair the server through direct access or remotely around the world. It gave full access and control of the server hardware and OS. It reduce time to manage servers and enable faster maintenance. We can achieve a full access and control to the server from any client that has web browser.

b- How does it work? Through its browser-based or command-line interface we can access and monitor the server. When Drac enabled the administrator will login and interact with the server as if sitting in front of it. It allows the administrator to access the system BIOS over the Internet when the server is rebooted. HOW ?! this is done when the admin configure the BIOS to send its output t serial port. This serial port is redirected to the DRAC serial port. So user can then login into the DRAC via SSH and connect the SSH session to this serial, which mean that the console will be accessed through the Internet.

Sources: 1- [http://www.dell.com/content/topics/global.aspx/power/en/ps2q02\\_bell](http://www.dell.com/content/topics/global.aspx/power/en/ps2q02_bell) 2- [https://en.wikipedia.org/wiki/Dell\\_DRAC](https://en.wikipedia.org/wiki/Dell_DRAC)

---

### 3.2 Installation Instructions

Since the installation was manually as I mentioned in the previous section. I will just clarify here how I installed the SSH daemon and configured it.

```
kotaiba@bristol:~$ sudo apt install openssh-server
```

Then after that i opened the sshd\_config in order to edit it through nano.

```
kotaiba@bristol:~$ sudo apt-get update
```

```
kotaiba@bristol:~$ sudo apt-get install nano
```

```
kotaiba@bristol:~$ sudo nano /etc/ssh/sshd_config
```

I changed the port to listen on TCP port 2222 instead of the default TCP port and I only allowed my user to connect. After that, I restarted the SSH service

```
kotaiba@bristol:~$ sudo systemctl restart sshd
```

### Question 2:

a- What is UEFI PXE booting? Abbreviation for (Pre execution Environment), is a method of having the computer or server (Client) boot using only its Network Card as long as the computer connected to the network. It do exactly the same ordinary boot procedure ( Power On > BIOS > HD ).

b- How does it work?

In order to boot via UEFI PXE, first we needs to power on the computer. so the UEFI will boot. After that its possible to boot via PXE. I pressed the F12 button key. The network-card will be booted and make a connection with the network. after that the computer will get an IP-address from the DHCP server in the network. Then the computer will locate and establish a connection with the PXE server in the network to download a Network Boot Program (NBP).

Sources: 1- <https://www.howtogeek.com/57601/what-is-network-booting-pxe-and-how-can-you-use-it/>  
2- <https://www.youtube.com/watch?v=zQ-TQhmjhuc>

### Question 3:

a- What is a GPT? GPT is an abbreviation for Globally Unique Identifier Partition Table is a standard for the layout of the partition table on a physical storage device on Desktop or Server. its format used to define the hard disk partitions in computers with UEFI start-up firmware. Its new standard that replaced MBR because it doesn't have limits like MBR. GPT is much larger.

b-What is its layout? Explain each elements? GPT layout used in new computers because of GPT partition table disk support up to 9.44-ZB zettabyte volume and ability to have up to 128 primary partitions.

GPT Layout:



Layout elements:

1- Protective MBR (LBA 0):

I will talk about it in the next question (Question 4).

2- Primary GPT Header (LBA 1): GPT header defines the available disk space and the composition of the partition table item size and quantity. It records the partition ID, header size, location, and the backup partition table header and its position and size.

### Question 4:

a- What is a Protective MBR and why is it in the GPT? The Protective MBR start in sector 0, precedes the GPT partition table on the disk. It contains one type partition that spans the entire length of the disk. The Protective MBR exists inside the GPT partition Table in order to protect it from "previously-released MBR disk tools". Because these tools do not know how to access the GPT disk. Old or legacy software interprets only the Protected MBR when it accesses a GPT disk.

Sources: 1- [https://nl.wikipedia.org/wiki/GUID\\_Partition\\_Table](https://nl.wikipedia.org/wiki/GUID_Partition_Table) 2-

<http://www.disk-partition.com/gpt-mbr/difference-between-mbr-and-gpt-1203.html> 3- Image source: <http://www.disk-partition.com/gpt-mbr/images/what-is-gpt/guid-partition-table-scheme.png> 4- <https://github.com/pbatard/rufus/issues/954> 5- <https://support.microsoft.com/en-us/help/302873/frequently-asked-questions-about-the-guid-partitioning-table-disk-arch> 6- <http://www.disk-partition.com/gpt-mbr/gpt-guid-partition-table-disk-1203.html>

---

## 4 Partitions

### Question 5:

a- Copy and dump the Protective MBR in hex format on your Wiki page, and fully annotate all their fields. This means you must describe the purpose of every field, and translate all fields that have a numerical value into human readable, decimal format.

I got help from my friend WIKI.

```
hexdump dump_sda.img -C -v | awk '{printf("%10d %s\n", (NR*16)-16, $0)}'
```

```
 0 00000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
16 00000010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
32 00000020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
48 00000030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
64 00000040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
80 00000050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
96 00000060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
112 00000070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
128 00000080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
144 00000090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
160 000000a0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
176 000000b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
192 000000c0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
208 000000d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
224 000000e0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
| .....|
240 000000f0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```

|.....|
256 00000100 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
272 00000110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
288 00000120 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
304 00000130 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
320 00000140 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
336 00000150 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
352 00000160 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
368 00000170 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
384 00000180 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
400 00000190 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
416 000001a0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
432 000001b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
448 000001c0 01 00 ee fe ff ff 01 00 00 00 af c2 e7 0e 00 00
|.....|
464 000001d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
480 000001e0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
496 000001f0 00 00 00 00 00 00 00 00 00 00 00 00 00 55 aa
|.....U.|

```

#### Byte(s) Description

0-440 Unused by UEFI systems

440-445 Unused and set to zero

446-509 MBR partition records that only have one entry pointing to the EFI Partition

510-511 Set to AA55

512 The rest of the logical block if any is reserved. Set to Zero

So all the records stored in the Protective MBR is in bytes 446-509.

#### Byte(s) Description Decimal format

446 Set to 0x00 to indicate a non-bootable partition 0

447-449 Set to 0x00010000, corresponding to the Starting LBA field 65536

450 Partition type set to "EE" which indicates the EFI partition 238

451-453 Set to the address of the last logical block on the disk. 0xFEFFFF  
16711679

454-457 Set to the LBA of the GPT Partition header. 0x01000000

16777216

458-461 Set to the disk size minus one. 0xAFC2E70E 2948785934

### Question 6:

a- Do the same for the GPT header.

```
kalachkar@desktop-15:~$ hexdump -C ~/Desktop/GPT_HEADER.bin 00000000 45 46 49 20 50 41 52
54 00 00 01 00 5c 00 00 00 |EFI PART....\...| 00000010 04 b9 e1 1f 00 00 00 00 01 00 00 00 00 00 00
00 |.....| 00000020 af c2 e7 0e 00 00 00 00 22 00 00 00 00 00 00 00 |.....".....| 00000030 8e
c2 e7 0e 00 00 00 00 f9 b5 60 00 a0 a6 e6 47 |.....`....G| 00000040 </code>
```

Offset, Description and Value:

Offset	Description	Value
0	Signature "EFI PART"	0x4546492050415254 EFI PART
8	Revision (version 1.0)	0x00000100 GPT version 1.0
12	Header size	0x5C000000 92
16	Header checksum (CRC32)	0x41C1528A x
20	Reserved	0x00000000 0
24	LBA of GPT header	0x0100000000000000 LBA 1
32	LBA of backup GPT header (last sector of the disk)	0xAFC2E70E00000000 LBA 250069679
40	Starting LBA for partitions	0x2200000000000000 LBA 34
48	Ending LBA for partitions	0x8EC2E70E00000000 LBA 250069646
56	Globally unique identifier (GUID) for the entire disk	0x0C1DCB133B1FE742 13CB1D0C-1F3B-42E7-
72	Starting LBA of partition table	0x0200000000000000 2
80	Number of partition entries	0x80000000 128
84	Size of each entry (bytes)	0x80000000 128
88	Partition table checksum (CRC32)	0xD7841B0F x
92	Reserved space	x

b- And for the first partition table entry.

```
kalachkar@desktop-15:/$ sudo dd if=/dev/sda
of=/sne/home/kalachkar/Desktop/First_Entry.bin bs=1 count=128 skip=1024
128+0 records in
128+0 records out
128 bytes copied, 0.000607398 s, 211 kB/s
```

```
kalachkar@desktop-15:/$ hexdump -C ~/Desktop/First_Entry.bin
00000000 28 73 2a c1 1f f8 d2 11 ba 4b 00 a0 c9 3e c9 3b
| (s*.....K...>. ; |
00000010 9a fd 7b 7f a8 bf b5 4d a6 9a 68 50 78 04 8c 99
| ..{....M..hPx...|
00000020 00 08 00 00 00 00 00 00 ff 07 10 00 00 00 00 00
| .....|
00000030 00 00 00 00 00 00 00 00 45 00 46 00 49 00 20 00 | .....E.F.I.
. |
```

```

00000040  53 00 79 00 73 00 74 00 65 00 6d 00 20 00 50 00 |S.y.s.t.e.m.
.P.|
00000050  61 00 72 00 74 00 69 00 74 00 69 00 6f 00 6e 00
|a.r.t.i.t.i.o.n.|
00000060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
|.....|
*
00000080

```

Offset, Description and Value

Offset	Description	Value
0	Partition type GUID	0x28732AC11FF8D211BA4B00A0C93EC93B
	EFI System partition	C12A7328-F81F-11D2-BA4B-00A0C93EC93B
16	Unique partition GUID	0xBE56C76486F61142A248012807BB0DCA
		64C756BE-F686-4211-A248-012807BB0DCA
32	Starting LBA of partition	0x0008000000000000 2048
40	Ending LBA of partition	0xFF07100000000000 1050623
48	Partition attributes	0x0000000000000000 None
56	Partition name in Unicode	0x004500460049002000530079007300740065006D00200050006100720074006900740069006F006E
	EFI System Partition	

c- At what byte index from the start of the disk do the real partition table entries start?

It start at byte 1024 .

Sources: 1- <http://thestarman.narod.ru/asm/mbr/GPT.htm> 2- <https://lxr.missinglinkelectronics.com/uboot/doc/README.gpt> 3- <http://thestarman.pcministry.com/asm/mbr/PartTables.htm> 4- PDF that you sent to us "http://www.jonrajewski.com/data/Presentations/CEIC2013/Partition\_Table\_Documentation\_Compressed.pdf". 5- Converter: <http://www.binaryhexconverter.com/hex-to-decimal-converter> 6- <https://cryptii.com/hexadecimal/text>

## Question 7:

a- If you wanted to add a (1 + your table number) GiB FreeBSD ZFS partition, called ØŠ3 (U+00D8 U+015A U+0033) to the table by hand, what values would you have to use for the entry (including the name) in the raw table on disk? Assume this partition comes after the first partition on the disk.

## TO ANSWER THIS QUESTION I USED MY COLLEAGUE TIM WIKI

Table number is 24 so we need 25 GiB = 26 843 545 600 bytes.

1 LBA = 512 bytes

I need in total 26843545600/512=52428800 LBA's

The Partition type GUID is 516E7CBA-6ECF-11D6-8FF8-00022D09712B

The Unique partition GUID is a random generated GUID. I use the following GUID

Sources: 1- [https://en.wikipedia.org/wiki/Unified\\_Extensible\\_Firmware\\_Interface](https://en.wikipedia.org/wiki/Unified_Extensible_Firmware_Interface) 2- PDF file from UEFI specification: [http://www.uefi.org/sites/default/files/Sources/UEFI\\_Spec\\_2\\_6\\_Errata\\_B.pdf](http://www.uefi.org/sites/default/files/Sources/UEFI_Spec_2_6_Errata_B.pdf) 3- <https://wiki.ubuntu.com/EFIBootLoaders> 4- A system and Network Engineering Student asked this

question on

["https://askubuntu.com/questions/519175/what-is-the-ubuntu-os-loader-and-where-does-it-reside"](https://askubuntu.com/questions/519175/what-is-the-ubuntu-os-loader-and-where-does-it-reside).

### Question 9:

a- What is the purpose of the GRUB boot loader in a UEFI system?

The purpose of a GRUB boot loader in a UEFI system is that GRUB is able to use ext2 or ext3 or ext4 file systems that Ubuntu is build on. But we need GRUB in order to boot into Ubuntu.

In addition to that GRUB makes it possible to give the option to the user to load multiple operating systems that are on the same disk. So during boot it gives the option to boot the operating systems that are available on the disk.

Sources:

1- [https://fedoraproject.org/wiki/GRUB\\_2](https://fedoraproject.org/wiki/GRUB_2)

2- <https://ubuntuforums.org/showthread.php?t=2267312>

3- <https://docs.oracle.com/cd/E19120-01/open.solaris/819-2379/fxjig/index.html>

**Question 10:** HERE a- How does the Ubuntu OS loader load the GRUB boot loader?

Via secure boot. The process of the secure boot here:

<https://wiki.ubuntu.com/SecurityTeam/SecureBoot>)

After feedback you said " Almost, but the question is how ?"

According to the source:

- 1- Microsoft signs Canonical's 'shim' 1st stage bootloader with their 'Microsoft Corporation UEFI CA'. When the system boots and Secure Boot is enabled, firmware verifies that this 1st stage bootloader (from the 'shim-signed' package) is signed with a key in DB (in this case 'Microsoft Corporation UEFI CA')
- 2- The second stage bootloader (grub-efi-amd64-signed) is signed with Canonical's 'Canonical Ltd. Secure Boot Signing' key. The shim 1st stage bootloader verifies that the second stage grub2 bootloader is properly signed.
- 3- The second stage grub2 bootloader boots an Ubuntu kernel (as of 2012/11, if the kernel (linux-signed) is signed with the 'Canonical Ltd. Secure Boot Signing' key, then grub2 will boot the kernel which will in turn apply quirks and call ExitBootServices. If the kernel is unsigned, grub2 will call ExitBootServices before booting the unsigned kernel)

Source:

1- <https://wiki.ubuntu.com/SecurityTeam/SecureBoot>)

**Question 11:** Explain how the GRUB boot loader, in turn, loads and run the kernel by answering



these 3 questions:

a- What type of filesystem is the kernel on? *SOMEONE HELP ME WITH THIS* Since the kernel located in `/boot/` we just use `mount` and find where our partition located. `kalachkar@desktop-15:/boot$ mount /dev/sda2 on / type ext4 (rw,relatime,errors=remount-ro,data=ordered)` which means that the kernel is on ext4 file system. b- What type(s) of filesystem does UEFI support? The file system supported by the Extensible Firmware Interface is based on the FAT filesystem. The EFI use FAT32 for system partition and FAT12 or FAT16 for removable media. c- What does the GRUB boot loader therefore have to do to load the kernel? *SOMEONE HELP ME WITH THIS* Mount the boot partition on the OS boot partition.

Sources: 1- [http://www.uefi.org/sites/default/files/Sources/UEFI\\_Spec\\_2\\_7.pdf](http://www.uefi.org/sites/default/files/Sources/UEFI_Spec_2_7.pdf)

**Question 12:** a- Do you need an OS loader and/or boot loader to load a Linux kernel with UEFI? Explain why or why not.

No, Because the EFI boot manager can use a stub kernel which allow the system to boot without an additional bootloader (in this case OS loader or Boot loader).

Source: 1- [https://wiki.gentoo.org/wiki/EFI\\_stub\\_kernel](https://wiki.gentoo.org/wiki/EFI_stub_kernel)

---

## 6 Initializing the OS

### Question 13:

a- What is the first process started by the kernel?

After the feedback I tried to proof it manually instead of googling, so as we see the first process started by kernel is `"/lib/systemd/systemd"` with PID 1:

```
kotaiba@kotaiba:~$ ps -eaf | head
UID          PID  PPID  C  STIME TTY          TIME CMD
root          1      0  0  12:24 ?        00:00:01 /lib/systemd/systemd --
systemd --deserialize 20
root          2      0  0  12:24 ?        00:00:00 [kthreadd]
root          4      2  0  12:24 ?        00:00:00 [kworker/0:0H]
root          6      2  0  12:24 ?        00:00:00 [ksoftirqd/0]
root          7      2  0  12:24 ?        00:00:04 [rcu_sched]
root          8      2  0  12:24 ?        00:00:00 [rcu_bh]
root          9      2  0  12:24 ?        00:00:00 [migration/0]
root         10      2  0  12:24 ?        00:00:00 [lru-add-drain]
root         11      2  0  12:24 ?        00:00:00 [watchdog/0]
```

b- Where is the configuration kept for the started process?

get help from Luc

In the file `/boot/efi/EFI/ubuntu/grub.cfg` the GRUB configuration it specifies:

```
set prefix=($root)'/boot/grub'
```

```
configfile $prefix/grub.cfg
```

So, in /boot/grub/grub.cfg file, we get:

```
echo      'Loading Linux 4.10.0-33-generic ...'
linux     /boot/vmlinuz-4.10.0-33-generic.efi.signed root=UUID=6278f48f-
f9c3-4f8f-9fdd-6160f62a0280 ro quiet splash $vt_handoff init=/sbin/upstart
```

c- It starts multiple processes. How is the order of execution defined?

Systemd starts units. “Units are the objects that systemd knows how to manage. These are basically a standardized representation of system resources that can be managed by the suite of daemons and manipulated by the provided utilities. Units in some ways can be said to similar to services or jobs in other init systems. However, a unit has a much broader definition, as these can be used to abstract services, network resources, devices, filesystem mounts, and isolated resource pools.”

Those units are booted in an order. “Unit files include the feature of dependencies. Any unit may want or require one or more other units before it can run.”. The order of which these units are based on:

If unit1 has Wants=unit2 as a dependency, when unit1 is run, unit2 will be run as well. But whether unit2 starts successfully does not affect unit1 running successfully.

When unit1 has Requires=unit2, however, again both units will run, but if unit2 does not succeed, unit1 is also deactivated. This happens regardless of whether the processes of unit1 would otherwise have worked fine.

But sometimes these processes need to run in a certain order:

If unit1 has the directive Before=unit2, then if both units are run, unit1 will be executed fully before unit2 starts.

If unit1 has the directive After=unit2, then if both units are run, unit2 will be executed fully before unit1 starts.

*Sources:*

1- [https://en.wikipedia.org/wiki/Linux\\_startup\\_process](https://en.wikipedia.org/wiki/Linux_startup_process)

2- Image: <http://processors.wiki.ti.com/images/b/b2/Linux-boot-sequence.png>

3- [https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/3/html/Reference\\_Guide/s1-boot-init-shutdown-process.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/3/html/Reference_Guide/s1-boot-init-shutdown-process.html)

4- <https://wiki.debian.org/systemd>


5- <https://www.digitalocean.com/community/tutorials/understanding-systemd-units-and-unit-files>

6- <https://fedoramagazine.org/systemd-unit-dependencies-and-order/>

#### **Question 14:**


a- As a final installation step, make your experimentation server reachable over IPv6.

My IPV6 address from ISP is : 2001:610:158:1046:145:100:104:16/64

kotaiba@bristol:~ host -t AAAA bristol.studlab.os3.nl bristol.studlab.os3.nl has IPv6 address 2001:610:158:1046:145:100:104:163

After I specified my IP address, I have to specify the IP address of my Gateway. So my IP address on Network : 2001:610:158:1046::0/64 and the first available IP address on this network is my gateway which is: GATEWAY: 2001:610:158:1046::1

After that I have to edit /etc/network/interface file in order to add the static IPV6 and enable it.

kotaiba@bristol:~ ping6 2001:610:158:1046:145:100:104:163 PING  
2001:610:158:1046:145:100:104:163(2001:610:158:1046:145:100:104:163) 56 data bytes

Check my network information: kotaiba@bristol:~\$ ifconfig eno1 Link encap:Ethernet HWaddr d4:ae:52:bf:e4:da

```
    inet addr:145.100.104.163  Bcast:145.100.104.191
Mask:255.255.255.224
    inet6 addr: fe80::d6ae:52ff:febf:e4da/64 Scope:Link
    inet6 addr: 2001:610:158:1046:145:100:104:163/64 Scope:Global
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:10085 errors:0 dropped:0 overruns:0 frame:0
TX packets:3722 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:890623 (890.6 KB)  TX bytes:544627 (544.6 KB)
```

And thats it my server is reachable over IPV6.

---