



# RHEL 7 Arabic Notes

تلخيص الطالب : أحمد عبدالمنعم

فيديوهات المهندس : مصطفى حموده

رابط ال Play List

[https://www.youtube.com/playlist?list=PLy1Fx2HfcmWBpD\\_PI4AQpjeDK5-5q6TG7](https://www.youtube.com/playlist?list=PLy1Fx2HfcmWBpD_PI4AQpjeDK5-5q6TG7)

# ملاحظات قبل ما تبدأ تقرأ

1- هذه الملخصات هي عبارة عن مجرد تجميع للمعلومات وتمت ازالة كل الكلام الذي ليس له علاقة بالكورس ، من الاخر كده اللي هنا بس هو المختصر المفيد والكلام مكتوب باللهجة المصرية مع مراعاة التوضيح لاقصى درجة

2- انا كنت بستخدم توزيعه فيدورا ك Main OS وكنت مسطب ال Virtualbox ومسطب جواه توزيعه CentOS علشان هتلاقى ملاحظات خاصة بالتوزيعه دي فى بداية لما كتبت التلخيص ده

2- التلخيص ده خاص بفيديوهات البشمةهندس مصطفى حموده بس ، يعنى حضرتك هتتفرج على الفيديوهات وتقرأ من التلخيص

3- كمان هتلاقى ملاحظات انا جبتها من النت علشان تساعدك انك تفهم اكثر وبالمناسبة كمان انا حطيت لينكات لمواضيع كتيرة متعلقة بالتراك ده ممكن انت تقرأها على النت هتفيدك جدا

4- ملحوظة اخيرة وهى ان التراك ده بدايته من الفيديو رقم 28 اليوم ال 13  
وده عنوان الفيديو

## **28-Day-13\_Understanding\_Boot\_Process**

لحد الفيديو رقم 65 اليوم ال 34 وده عنوان الفيديو

65-Day-34 Source \_ Code \_ Installation

**التراك الثانى**

**Admin 2**

## Table of Contents

1-Part One : Boot Process + Systemd + reset root password.....	10
2-Network Basics.....	34
2.1-Network Troubleshooting.....	41
2.2-NIC Teaming.....	45
2.3-Network Bridging.....	50
2.4-Interface Naming.....	51
3-Setting GRUB Password + Getting Help.....	56
4-RAID.....	68
5-LVM.....	79
6- Swap.....	103
7-Quota Management.....	108
8-Access Control List.....	126
9-YUM.....	135
10-RPM.....	145
11-Source Code Installation.....	152

# Table Of Contents

1- اول جزء ده هيتكلم عن ال **Boot Process** وعن **Systemd** وازاى تعمل **reset** لل **Root Password** وده بدايته من الفيديو رقم 28 لحد الفيديو رقم 33 ( والجزء ده بالتحديد غالبية عبارة عن **Basics** بسيطة جدا المفروض تكون عند اى واحد بيستخدم ال **GNU/Linux** )

2- تانى جزء عندنا وهو خاص بال **Networking** وده متقسم لشوية مواضيع كده ، وللأمانة الشديدة انا معرفتش الخص غير شوية معلومات لان من وجهة نظرى موضوع الشبكات ده غالبية عملى فانا بصراحة لخصت اللى قدرت عليه وطبعا حضرتك لازم تدرس **CCNA** علشان تفهم الدنيا اكر شوية ، بالنسبة للمواضيع فاللى موجود هنا هو موضوع ال **Network Profiles** وموضوع ال **IPv6** وموضوع ال **Network Troubleshooting** وموضوع ال **NIC Teaming** وموضوع ال **Network Bridging** واخيرا موضوع ال **Interface Naming** **المهم الجزء ده بدايته من الفيديو رقم 34 لحد الفيديو رقم 45**

3- تالت جزء عندنا وده بيتكلم عن ازاي تعمل **Password** لل **GRUB** وازاي برضو  
تعمل **reinstall** لل **First Stage** واخيرا ازاي تقدر انك تتعامل مع ال **Man Pages**  
وال **history** وغيرهم , الجزء ده بقى بدايته من الفيديو رقم 46 لحد الفيديو رقم 49

4- رابع جزء عندنا وده بيتكلم عن ال **Basics** بتاعت ال **RAID** وشوية حاجات كده  
خاصة بال **Storage** , الجزء ده بقى بدايته من الفيديو رقم 49 لحد الفيديو رقم 53

5- خامس جزء بيتكلم عن ال **LVM** وده بدايته من الفيديو رقم 54 لحد الفيديو رقم  
56

6- الجزء السادس وده صغير خالص , بيتكلم عن موضوع ال **Swap** وهو عبارة عن  
الفيديو رقم 57

7-الجزء السابع وده بيتكلم عن ال **Quota Managment** وده عبارة عن الفيديو رقم  
58



8-الجزء الثامن وده بيتكلم عن ال **Access Control List** وده عبارة عن الفيديو رقم 62

9-الجزء التاسع وده بسيط جدا لانه بيتكلم عن ال **YUM** وهو عبارة عن الفيديو رقم 63

10-الجزء العاشر وده برضو بسيط جدا وبيتكلم عن ال **RPM** وده عبارة عن الفيديو رقم 64

11-الجزء الاخير وده بيتكلم عن ال **Source Code Installation** وده اخر موضوع فى **Admin 2** , الجزء ده مشروح فى الفيديو رقم 65

## 1-Part One : Boot Process + Systemd + reset root password

فى البداية ياريت حضرتك تنزل الكتب دى لانها بتشرح ازاي عملية ال boot  
بتم فى اللينكس

1- لينك اول مقال

<https://www.ibm.com/developerworks/library/l-linuxboot/index.html>

2- تانى كتاب وده حلو اوى وهو عبارة عن Slides بتشرح Systemd

[https://www.google.com.eg/url?](https://www.google.com.eg/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKEwjcnuCm0LfdAhUtyoUKHcvPDu0QFjAAegQIBRAC&url=https%3A%2F%2Fpeople.redhat.com%2Fpladd%2Fsystemd_NYRHUG_2016-03.pdf&usg=AOvVaw0XoZZVErrketHQu3UQ4cYe)

[sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKEwjcnuCm0LfdAhUtyoUKHcvPDu0QFjAAegQIBRAC&url=https%3A%2F%2Fpeople.redhat.com%2Fpladd%2Fsystemd\\_NYRHUG\\_2016-03.pdf&usg=AOvVaw0XoZZVErrketHQu3UQ4cYe](https://www.google.com.eg/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKEwjcnuCm0LfdAhUtyoUKHcvPDu0QFjAAegQIBRAC&url=https%3A%2F%2Fpeople.redhat.com%2Fpladd%2Fsystemd_NYRHUG_2016-03.pdf&usg=AOvVaw0XoZZVErrketHQu3UQ4cYe)

او من خلال اللينك ده

[https://people.redhat.com/pladd/systemd\\_NYRHUG\\_2016-03.pdf](https://people.redhat.com/pladd/systemd_NYRHUG_2016-03.pdf)

نبدأ بقى ، مبدئيا كلمة boot معناها الحذاء ، ومعناها هنا عملية اقلاع وتشغيل  
الجهاز

طيب اول حاجة بيعملها ال **Bios** هى POST ودى اختصار

ل Power On Self Test ودى معناها انه بيروح يعمل Check لكل الهارد وير  
اللى موجودين عندك فى الجهاز ، مثلا لو مفيش رامات ، او ان الرامات

مش Compatible مع ال Mother Board دى ، طيب لو فى مشكلة هتلاقيه بيصفر ، وكل صفارة ليها معنى ، يعنى مثلا صفارتين متقطعتين معناهم ان فى مشكلة فى الهارد وهكذا ، وبالمناسبة ال Bios مش بيعمل Test على كل حاجة موجودة ، هو بيعمل Test لل minimum هاردوير علشان يقدر يشغل الجهاز ، الحاجات اللى الجهاز مينفعش يقوم من غيرها

طيب الخطوة اللى بعد كده وهى انه بعد ما ال Bios بيعمل Power On Self Test فى اول ما يشتغل ، هبدأ يدور على ال Bootable Devices ويقارنها بالاعدادات اللى عنده ، زى مثلا انه يلاقى **اسطوانة** bootable او **فلاشة** او Network ممكن تبوت منها ، او عندك مثلا operating system على ال disk بتاعك ممكن تبوت منه

والاعدادات دى بتكون محفوظة على ال Mother Board عندك وهى اللى بتقوله ان اول bootable device مثلا هى الاسطوانة وهكذا بقى ولو ملقاش حاجة ييبوت منها هتلاقيه طلعك رسالة error ويقولك no bootable device found وطبعا رسالة ال error او طريقة عرضها بتختلف من جهاز للتانى ، مثلا فى **dell** بتيجى رسالة ال error فى شكل صفحة زرقاء انما مثلا فى hp بتكون شاشة سوداء ، وطبعا يقولك

press any key to reboot

الخطوة الثالثة وهى ان ال Bios هيروح ي load حاجة اسمها ال **IPL** ، ودى اختصار ل Initial Program Loader ، طيب ال **IPL** اصلا عبارة عن code برمجى صغير جدا جدا موجود على ال Mother Board بتاعت الجهاز ، وكل وظيفته انه يروح يعمل execute لحاجة اسمها ال Bootloader ، يعنى يروح يشغل ال Bootloader

طيب اى بقى ال boot loader ده ؟؟؟ هو عبارة عن software صغير جدا موجود عندك على ال bootable device لو عندك hard disk وبت boot منه فهيكون موجود على ال **mbr**

اذا ال **IPL** هو اللى بينادى على ال boot loader علشان ي load ال os

طيب زى ما انت عارف ان ال mbr بيتكون من 3 اجزاء هما

ال Partition Table وده حجمه **64 بايت** ، وال Boot Loader وده حجمه

**446 بايت** ، وال magic number وده حجمه 2 بايت

هنا بقى اللى هيروح ينادى على ال boot loader اللى هو جزء من ال mbr هو ال **IPL**

الخطوة اللى بعد كده وهى ان ال Boot Loader هيبداً يعمل load لل

Kernel ، وبعدين بعد ما ال Kernel يقوم يروح ينادى على systemd ، واخر

خطوة بقى وهى ان systemd هتروح هتنادى كل ال services او ال Daemons على حسب بقى انت بتسميها اي .

السؤال هنا بقى ، بما ان ال bootloader اللى هو جزء من ال mbr ومساحته 446 بايت ، هل المساحة دى كافية ان ال bootloader يتخزن فيها؟؟  
طيب انت عندك اصلا فى توزيعه زى fedora او Ubuntu ال boot loader بيكون ليه اصلا background لما الجهاز بيدأ ، واصلا مساحة ال background دى او الصورة بتكون بالكيلو

طب ازاي برضو؟؟؟ ، واذا كانت اصلا ملفات ال bootloader بتكون اكبر بكثير من مساحة ال boot loader ذات نفسه اللى هى **446 بايت** ، واذا كان ملف ال configuration بتاع ال bootloader اصلا مساحته بالكيلو ، يبقى ازاي هيحط كل ده فى ال bootloader اللى مساحته 446 بايت ؟

قالك بس يا معلم ال boot loader ده انا هقسمه لمرحلتين هما  
ال first stage وال second stage و ال first stage ده هو اللى هيكون حرفيا فى ال mbr وال second stage بما انها هتحتوى على الملفات الكبيرة يبقى خليهالى على الهارد ، وهيحطها لك فى المجلد اللى هو

/boot

طيب ده اصلا مش عيب فى ال **bootloader** ذات نفسه ، ده اصلا قصور  
فى ال Design بتاع ال mbr ذات نفسه ، واللى صمم ال mbr هم المهندسين  
بتوع Microsoft

وطبعا ال First Stage اللى هو موجود فى ال mbr كل وظيفته ان ينادى على  
ال second stage اللى موجودة على الهارد

نىجى بقى للنقطة الاكثر الاهمية فيما يخص موضوع تسطيب الويندوز  
واللينكس جمب بعض ، طيب لو مثلا حبيت تنزل ويندوز جديد ، يعنى تشيل  
الويندوز القديم اللى موجود جمب اللينكس وتحط ويندوز جديد ، اي بقى اللى  
هيحصل هنا ؟؟؟ مبدئيا كده الويندوز عنده boot loader خاص بيه ، اسمه

**NTLDR**

وده اختصار ل

### **New Technology Loader**

هيعمل اى بقى ؟؟ هنا بقى هو هيروح تحديدا لما يسمى بال first stage اللى  
موجود فى ال mbr ويمسحها تماما يعنى هيعملها overwrite ويحط نفسه  
تماما وللأسف ال **NTLDR** مش بيْفهم اذا كان في لينكس ولا لأ

طبعا لما تعمل reinstall للويندوز ، بيكون اللينكس طبعا لسه موجود وال  
second stage وملفاتها لسه موجودة ، بس اللى حصل فعليا هو ان  
ال first stage اتمسحت

بالمناسبة الكلام ده الويندوز بيعمله فى حالة ال gpt برضو ، طبعا اوعى  
تتهبل وتروح تعمل reinstall بالكامل للينكس ، روح اعمل reinstall بس لل  
first stage

طبعا لما تعمل reinstall لل first stage هيروح هو كمان يعمل overwrite لل  
NTLDR وساعتها بقى ال First Stage هتفهم اوتوماتيك ان انت عندك ويندوز  
ولينكس

وطبعا زى ما قولنا ان ال bootloader بيعمل load على مرحلتين الاولى هى  
ال first stage وبعدها ال second stage لحد هنا تمام اوى

النقطة اللى بعد كده وهى ان الكرنال مقسوم لجزئين الاول هو الكرنال  
والثانى ملف اسمه **initramfs** والثانى ده بقى وظيفته انه بيشيل ال drivers  
وبقية الحاجات اللى انت ممكن تعملها access

زى مثلا انت عايز تعمل access ل filesystem زى ال NTFS او مثلا  
ال Network فهو بيشيل ال drivers دى

طيب اى وجه الاستفادة من انه يحطلك ملف ال kernel لوحده وملف ال init لوحده؟؟ بص ده هيسمحك اولا ان ال Foot Print بتاع الكرنال تكون صغيرة جدا يعنى الكرنال هيستهلك memory صغيرة جدا وكمان ده هيسمح للكرنال انها تعمل load بطريقة اسرع

انما العكس بقى فى ويندوز ، حاطط الكرنال كله هو وال modules وال drivers فى ملف واحد علشان كده اى غلطة فى الكرنال بتوقع النظام كله

ملحوظة جانبية وهى ان الحالات اللى انت بتحتاج انك تعمل فيها restart للجهاز هما اتنين لما تكون ال SELinux معمول ليها disabled وانت هتعملها enable والحالة الثانية لما تعمل upgrade للكرنال ودلوقتى اصلا فى حاجة اسمها ال Kernel Patch ممكن انك ت apply ال Patches للكرنال من غير ماتعملها restart

معلومة جانبية وهى ان ال system call معناها ان النظام او او systemd بالتحديد هى اللى هتبدأ تنادى على بقية ال services زى الاباتشى وال ssh وهكذا بقى

وبالتالى فانت عندك systemd هتنادى على كل ال service بالتوازي

In Parallel



طبعا init علشان تنادى على كل Process كانت او بشكل ادق بتحتاج Script لكل Process يعنى مثلا ال ssh ليها init script علشان تبدأ ، انما فى systemd فدى بتحتاج ما يسمى بال unit file ، ودى عبارة عن ملفات كل اللي فيها variable بيساوى value معينة

حلو خالص طب لو عندك init script ازاي هتشغلها على systemd ؟؟ ، طبعا systemd هى Compatible مع ال init فاللى هيجصل ان systemd هتنادى على ال init وال init هتروح تنادى على ال init script وخذ عندك بقى ، ان systemd لما اتكتبت اتكتب معاها حاجة اسمها journald ، وممكن تعتبر الاثنين انهم توأم

وزمان كان فى مشكلة متعلقة بال logs وهى ان ال init كانت بتنادى على حاجة اسمها rsyslog بعد ما بتنادى على كذا service ، وده فى حد ذاته مشكلة وهى ازاي انت هتشوف ال logs بتاعت ال services اذا كان المكان اللي بيتجمع فيه ال logs لسه مبدأش اصلا ، وبالتالي انت مش هتقدر تشوف المشكلة كانت فين بالظبط هل فى ال configuration file ولا فين بالظبط ، كل ده بسبب ان ال rsyslog بدأت بعد مال services بدأت وبالتالي هما لما بدأوا يكتبوا systemd حاولوا يتلافوا المشكلة بتاعت ال init بانهم يكتبوا حاجة جنبها اسمها journald

طيب الفرق بين enabled وبين active فيما يخص ال services وهى ان enabled معناها ان ال service يقوم لما الجهاز يشتغل وطبعاً active معناها انها شغالة دلوقتى وكل service بياجى معاها Default Configuration هو اللى بيقولها اذا كانت هتكون enabled ولا disabled اللى هو ال **vendor preset**

فى ال init لو مثلاً فى service معينة وعاييزها تبقى disabled والجهاز بيتوت استخدم الامر ده

### **chkconfig sshd off**

انما بقى فى Systemd لو عايز اعرف هل ال service معمول ليها enable ولا لأ عن طريق الامر ده

### **systemctl is-enabled sshd**

ملحوظة لو كتبت systemctl وضعت double tab هيجبك كل ال options اللى بتيجى مع systemctl

بالنسبة لعملية ال mask لل service فده معناها انك هتمنع ان اى حد يشغلها

عندنا بقى فى ال run levels ، ال run level رقم واحد انت بتستخدمه فى ال Trouble Shooting يعنى مثلا لو نسيت ال root passwd فانت اوتوماتيك بتروح لل run level ده

طيب لو انت شغال init اسمه run level ولو انت شغال systemd فاسمه target وفى الويندوز اسمه mode زى ال **safe mode** كده

عندك بقى بالنسبة لل run level رقم 2 ورقم 3 فانت عندك انهم انضموا وبقوا الاتنين فى target واحد اسمه

## **multi-user.target**

وال graphical فى systemd بقى اسمه graphical.target لو خدت بقالك والجهاز بيفتح بيقولك

## **reached target graphical user interface**

طبعا ال run level رقم 4 ده اصلا unused

طيب هل ينفع انك تروح ل run level مختلف تماما وانت فى run level مختلف برضو ؟ ايوه طبعا عن طريق الامر ده

`systemctl isolate *.target`

ومكان علامة ال \* تكتب ال target او ال run level اللى انت عايز تروحله وبالمناسبة الملفات الاصلية بتاعت systemd موجودة فى

`/usr/lib/systemd`

فى ملحوظة هنا وهى ان انت لما كنت بتعمل enable ل service معينة هو بكل بساطة كان يياخد منها soft link من المكان ده

`/usr/lib/systemd`

ويحطها فى

`/etc/systemd/system`

علشان ببساطة لما الجهاز يفتح يقوم ينادى على ال service دى من المسار بتاع etc علشان ده فيه كل ملفات ال configurations بتاعت ال system كله

وطبعا متنساش برضو ان ال **init** بتعمل ما يسمى بال sequential boot انما **systemd** بتعمل parallel boot

وبرضو systemd هى عبارة عن dependency based boot بمعنى انى ممكن عن طريقها اقول ال service الفلانية تشتغل بعد ال service دى وهكذا

معلومة وهى ان الامر init والامر telinit الاتنين بيعملوا نفس الحاجة

الامر `systemctl list-units` بيعرضك ال service بحالتهم ، انما الامر

`systemctl list-unit-files` ده هيعرضك ملفات ال services ، طبعا كل ملف

بيمثلك service معين والملفات بتاعت ال services دى هتلاقها هنا

`ls /usr/lib/systemd/system/`

والملفات اللى هنا هى اللى بيتعمل منها soft target بيكون موجود فى المسار

```
ls /etc/systemd/system
```

ومتنساش ان الملفات اللى systemd بتتعامل معاها مع ال system اسمها unit files

وملفات ال units موجوده هنا

```
cat /usr/lib/systemd/system/*
```

وبالنسبة بقى لحكاية ال service لما تكون **static** دى بتكون عبارة عن service مساعدة لل service الاصلية علشان الاصلية دى تقدر تشتغل يعنى من الاخر كده بتكون مش service حقيقية

اما بقى الفرق بين restart و reload لل service باستخدام systemctl ، هو ان restart بتقفل ال process وتشغلها مرة ثانية ، انما reload بتخلي ال process تعمل reread لملف ال configuration بتاعها

الموضوع اللى بعد كده وهو ال **grub** ، وانت كل سطر بتشوفه فى  
ال grub اسمه menuentry ، وده عبارة عن selection انت ممكن تختاره

طيب اصلا اللى بيدى اسماء ال disks وال partition هو ال Kernel يعنى  
الاسم sda و sda5 وهكذا اللى بيديه للبارتيشنز هو ال kernel ، طب اصلا  
قبل ال kernel ما يقوم انت عندك ال grub هو اللى بيقوم قبل ال kernel  
وال grub ده low level اوى يعنى بيشوف كل ال disks بالاسم ده **hd**  
وطبعا الرقم اللى بعد كده بيكون رقم الهارد ذات نفسه

يعنى مثلا hd0 معناها الهارد رقم واحد ولو عندك هاردين فالهارد التانى اسمه  
hd1 وهكذا بقى ، طب ليه الحكاية دي ؟ ، بكل بساطة لان ال grub  
بيعمله load قبل الكرنال اساسا فا مش بيقدر يشوف الاسماء اللى هى sda  
اللى بتديها ال kernel للهارد والبارتيشنز بعد كده ، فا علشان كده هتلاقه  
شايف الاسماء بتاع الديسك بالاسماء البدائية بتاعتها

طيب الكلام ده بالنسبة للهارد ككل ، انما بقى بالنسبة للبارتيشنز هتلاقه  
بالاسم ده ، اول بارتیشن هيكون اسمه **msdos1** وتانى بارتیشن هيكون  
اسمه **msdos3** وهكذا

إذا لو عندك بارتيشن مثلا واسمه sda1 فال equivalent بتاعته يعنى اللى  
بيساويه فى ال grub هيكون **hd0,msdos1**  
يعنى من الاخر كده الجملة دى

**root='hd0,msdos1'**

معناها انه هيعمل load لل root filesystem اللى موجود فى ال disk الاول  
فى البارتيشن الاول  
طيب المعلومات دى بيحبها منها ؟؟؟ اولا نفذ بس الامر ده

**fdisk /dev/sda**

وهيظهرلك نوع ال partition table بتاعك هل هو **mbr** ولا **gpt** يعنى هل هو  
dos ولا gpt ، مثلا عندى انا فى فيدورا علشان انا gpt فا ظاهرلى بالشكل ده  
**set root='hd0,gpt3'**

معناها ان ال **root file system** موجود عندى فى الهارد الاول فى  
البارتيشن رقم 3 ، وده لانى عامل 2 بارتيشنز قبله هم ال efi وال swap  
فظهر انه البارتيشن التالت

دخلنا على موضوع الفرق بين ال mbr وال gpt ، اولاً انت عندك لو ال mbr بقى corrupted بسبب virus معين ، فانت الدتا اللي على الهارد بقت بالسلامة بقى ولو ال partition table نفسه طار يبقى الداتا بتاعتك مع السلامة ، طيب هما علشان يحلو المشكلة دي عملوا ال

### gpt partition schema

وقالك بس انا هحطلك نسخة منه فى اول الهارد ونسخة تانية فى اخر الهارد ، وبالمناسبة المهندسين بتوع Apple هما اول ناس اخترعوا حكاية ال gpt وبدأوا يتكلموا عنه فى 2004 وبعدين اتكلموا عنه فى 2004 وبعدين بدأ ينفذوا فى 2008 وبقى ال defacto standard بتاع العالم كله ، وطبعاً اكبر عدد من البارتيشنز ممكن يتعمل على ال gpt هو 128 بارتيشن

الحاجة التانية او العيب التانى اللي كان موجود فى ال mbr وهو ان كل block كان ليه address ، وكان عدد ال address المتاحة هـى

$2^{32}$

وده معناه ان مساحة اى disk مينفعش تتخطى مساحة  $2^{32}$  (2 اس 32 بت) يعنى 4 مليار sector يعنى 4 مليار block ، طب اصلاً ال sector size مساحته 512 بايت يعنى 4 مليار لو قسمتهم على مساحة ال sector هيكون تقريباً عندك 2 تيرابايت للهارد الواحد



انما ال gpt عدد ال address المتاحة هي 64 بت يعني

$2^{64}$

وده رقم كبير فشخ

وبالمناسبة ال mbr ممكن تلاقي اسمه في ال Bios كده **LBA** يعني  
**Logical Block Addressing** والاتنين مرادفين لحاجة واحدة

انما ال **GPT** ممكن تلاقي اسمه في ال BIOS بالشكل ده **GUID** او **EFI**  
او طبعا **UEFI**

وكمان انت عندك ال bootloader او ال Grub كان بيقول لل kernel ان ال  
root file system موجود في

```
set root='hd0,msdos1'
```

وكمان كان بيقوله على ال UUID بتاع البارتيشن  
يعني بيقوله روح دور هنا

```
search --no-floppy --fs-uuid --set= root --hint-bios= hd0,msdos1  
--hint-efi= hd0,msdos1 --hint-baremetal= ahci0,msdos1  
--hint= 'hd0,msdos1' 0f089c70-e20f-46a4-8a92-af7974f0c4ce
```

والسبب في ان ال bootloader يقول للكرنال دور على البارتيشن اللى ال UUID بتاعه كذا ، هو انك لو عندك 2 هارد وبدلت واحد مكان التانى فهو كده كده هيدور على ال UUID وملوش دعوة بقى باى بارتيشن دى ، وطبعاً ال UUID بيتغير فى حالة واحدة بس وهى انك لو عملت format للبارتيشن ده ، واصلا ال UUID بيت create لما تعمل format للبارتيشن

اما بقى بالنسبة لل rhgb quiet فدى اختصار ل

redhat graphical boot

وكلمة **quiet** معناها انه بيقله خبى اللى بيحصل ، خبى ال services اللى بتشتغل

النقطة اللى بعد كده وهى ان السطر اللى بيبدأ ب linux16 بيعرفه فين مكان ال kernel

linux16 /vmlinuz-3.10.0-862.3.2.el7.x86\_64

والسطر اللى بعده اللى هو بدايته initrd16 بيعرفه فين مكان ال init ذات نفسها

initrd16 /initramfs-3.10.0-862.3.2.el7.x86\_64.img

طب ليه ؟؟ لان ببساطة الكرنال وهو بي load محتاج يشوف الملف ده علشان يعمل load لل modules وال drivers

ملحوظة انت ممكن تلغى ال RedHat Graphical علشان تشوف الكرنال ذات نفسها وهى بتعمل boot

وطبعا عادى جدا انك تبوظ ملف ال configuration بتاع ال grub وزى ما انت عارف الامر اللى بيعمل regenerate للملف ده هو

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```

اه هنا بقى فى نقطة مهمة وهى ان موضوع انك تعمل rebuild لملف ال grub بيختلف من توزيعه للتانية ، فانت هتبحث الاول عن الملف grub.cfg وتشوف موجود فى انهى مجلد بالظبط وبعدها تشوف كمان اذا كانت التوزيعه دى بتستخدم ال grub2 ولا ال grub بس وتعمل regenerate بناءا على مكان الملف ده ، الموضوع ده بيختلف من توزيعه للتانية

وكل مره هتعمل reconfig لملف ال grub هترجع الاعدادات ال default زى ما هى وهيرجع الاسم وكل حاجة زى ما كانت ، انما بقى لو عايز تحفظ الاعدادات وتخليها متغيرش يبقى تروح تعدل الملف ده

**vim /etc/sysconfig/grub**

والملف ده عبارة عن soft link للملف ده

**vim /etc/default/grub**

وطبعا متنساش تعمل regenerate لملف ال grub بعد ما تعدل فيه اى تعديل

اوبالا ودى ملحوظة مهمة فشخ وهى ان ال grub ببدا يعد ال menu entries من رقم صفر ، يعنى لو عندك مثلا ويندوز ولينكس ، وانت عايزال grub يروح للويندوز الاول يبقى تعدل الملف ده

**vim /etc/sysconfig/grub**

وبعدها هتيجى للسطر ده

**GRUB\_DEFAULT= saved**

وتمسح كلمة saved وتخلي بدالها رقم الويندوز يعنى شوف الويندوز رقم كام وحطه بدال الكلمة دى وبرضو متنساش تعمل regenerate لمف ال grub

**طيب عندنا بقى موضوع جديد وهو ازاى تعمل reset لل ROOT Password**

بص يا سيدى كل ملف عندك او كل process بيكون ليها Label بمعنى ان ال SELinux هى اللى بتدى لكل ملف ال Label بتاعه يعنى اى وظيفته وكل ملف عندك هيكون ليه label فال label ده هيحدد الصلاحية بتاعت الملف ده وبالمناسبة ال recovery mode مش بي support ال SELinux اصلا وال SELinux هى عبارة عن Security System بيدى لكل ملف وكل حاجة فى النظام Label لو ال Label بتاعتها طلع مطابق لل Policy يبقى خير وبركة

طبعا المفروض انى بعد ما اغير ال root passwd المفروض اقول للنظام رجعلى ملف ال shadow وخليه مطابق لل SELinux Policy ، يعنى لازم اقول للنظام انه يرجعلى ال Label لاصله ، لانه لو مرجعش لاصله النظام لما ياجى ي boot هيلاقى ال label بتاع ملف shadow مثلا متغير فا هي Fail او ال SELinux هتمنع النظام من انه ي boot

دلوقتى بقى وانت فى ال grub لو عايز تاخد root access على الجهاز عدل فى الملف بتاع اول kernel مثلا ده لو عندك يعنى اكثر من Kernel واكتب

**rd.break**

بمعنى ان انت هتكتب الكلمة دى فى نهاية السطر اللى بدايته كلمة

**linux16**

واصلا اصلا **rd** معناها انه ال kernel بيتعمله load وبعدين ي load ملف ال initrd فا هنا بقى ال rd.break هتخلى ال kernel يقف ميكملش بعد ما يعمل load لملف ال initrd يعنى الكرنال ذات نفسه مش هي Load وبعدين تضغط

ctrl + x

هتبص تلاقى نفسك داخل لل root ع طول

انت بقى لما تدخل فى الوضع ده ، هتلاقى الدنيا اتغيرت خالص ، مبدئيا كده ال Kernel بتاعك عنده File System مصغر يشابه ال / ، بس مش هو ال file system اللى موجود على الهارد

لكن هو مشابه ليه ، والنسخة دي بقى هى ال Mounted فى ال Memory  
دلوقتى ، انما ال disk الحقيقى معمول ليه Mount فى

/sysroot

علشان كده لما تكون فى وضع ال

switch-root#

وتيجى تكتب

ls /

ده بيعرضلك النسخة المصغرة اللى معمول ليها Mount فى ال Memory  
وطبعا انت لازم تكتب mount علشان تعرف ال root file system بتاعك  
الاصلى معمول ليه mount فى انهى مكان بالظبط وطبعا هنا فى خازوق وهو  
ان ال root file system بيكون معمول ليه mount ك read only ، طب انت هنا  
اصلا عايز تغير الباسورد بتاع ال root وازاي هتغيره وهو اصلا read only  
هنا انت بقى هتضطر تعمل **remount** ك **rw** لل root file system اللى  
معمول ليه mount فى المجلد

/sysroot

عن طريق الامر ده

**mount -o remount,rw /sysroot/**

ال -o معناها option

وبعدها بقى هتغير ال root عن طريق الامر

**chroot /sysroot/**

وبعد ما تغير ال root passwd وتيجى تستخدم الامر

**ls -lZ /etc/passwd**

هتبص تلاقى ال Context او ال Label بتاع الملف بقى عبارة عن علامة استفهام ، وده لانى ال SELinux مش Active فى وضع ال Recovery Mode

يبقى كده انا المفروض اقول لل File System اعمل relabel للملف مرة تانى قبل ما تبدأ ال OS ، وهنا بقى عندك حلين

يا اما انك تعمل ال relabel بطريقة Manually ، او انك تعمله بالطريقة السهلة اللى هى Auto Relabel ، وكده هو هيعدى على الملفات ويشوف ال Label بتاعها المفروض يكون اى على حسب ال Policy ويعمله Apply

طب الموضوع ده هيثم ازاي ؟ انت هتعمل ملف مخفى فى ال / بالشكل ده

**touch /.autorelabel**

وده معناه ان ال selinux قبل ما الكرنال ي load هتعدى على كل الملفات وترجع ال label بتاع كل ملف زى ما كان وطبعا كل الهدف من الملف ده انى بقول لل SELinux معلش انا عدلت ملف شوفلى الملف ده كان ال label بتاعه اى ورجعيه لاصله

كده الموضوع ده فيه خطر ، وهو ان اى واحد ممكن يعمل reboot ويعدل فى ال grub ويخش يغير الباسورد بتاع ال root فالحل هنا بقى ان انت هتعمل باسورد لل grub ذات نفسه وفى حالة اذا كنت عايز تعمل Security اكثر للهارد فانت هتعمله full encryption

فى بقى شوية حاجات كده خاصية بيك انت ك System Admin  
زى انك تعمل hardening لل server وانك تمنع ALT + CTRL + Delete

وبالنسبة للامر ده بتاع الت كنترول ديليت ، احنا فى حاجة عندنا وهى ان لما بضغط على الكام زرار ده بيعمل حاجة شبه ال init 6 ، وطالما انى شغال Systemd يبقى نستبعد ال init  
وطبعا انت لو عدلت فى ال soft link هيتعدل تلقائيا فى الملف الاصلى ،  
طب انت دلوقتى فى ملف ال Alt + Ctrl + Delete هيعدل هو اوتوماتيك فى  
ملف ال reboot.target

طبعا انت هنا لو عايز تلغى تفعيل Service معينة فانت هتستخدم الامر  
systemctl mask ctrl-alt-del.target

كده هيخلي الملف ده يمشور على

`/dev/null`



و dev/null/ ده بقى عبارة عن special file هو عامل زى المالا نهاية كده ،  
ولو عايز تمسح الملف بشكل secure ابعته ل dev/null  
والمفروض انك طالما عدلت فى ملف target او service معينة يبقى  
المفروض تعمل

`systemctl daemon-reload`

كده انت لما بتعمل mask انت كده مش بتعدل فى الملف الاصلى ، لا لا انت  
بتعدل فى ملف ال configuration بتاعه اللى هو عبارة عن soft link وموجود  
فى

`/etc`

وطبعا كل ملفات systemd الاصلية موجودة فى `usr/lib/systemd` ، انما  
التعديلات بتتم فى `etc`

وده يوصلنا لنتيجة منطقية وهى ان systemd وهى شغالة بتبص الاول على  
الملفات اللى موجودة فى `etc` لو ملقتهاش هناك بتروح تبص فى

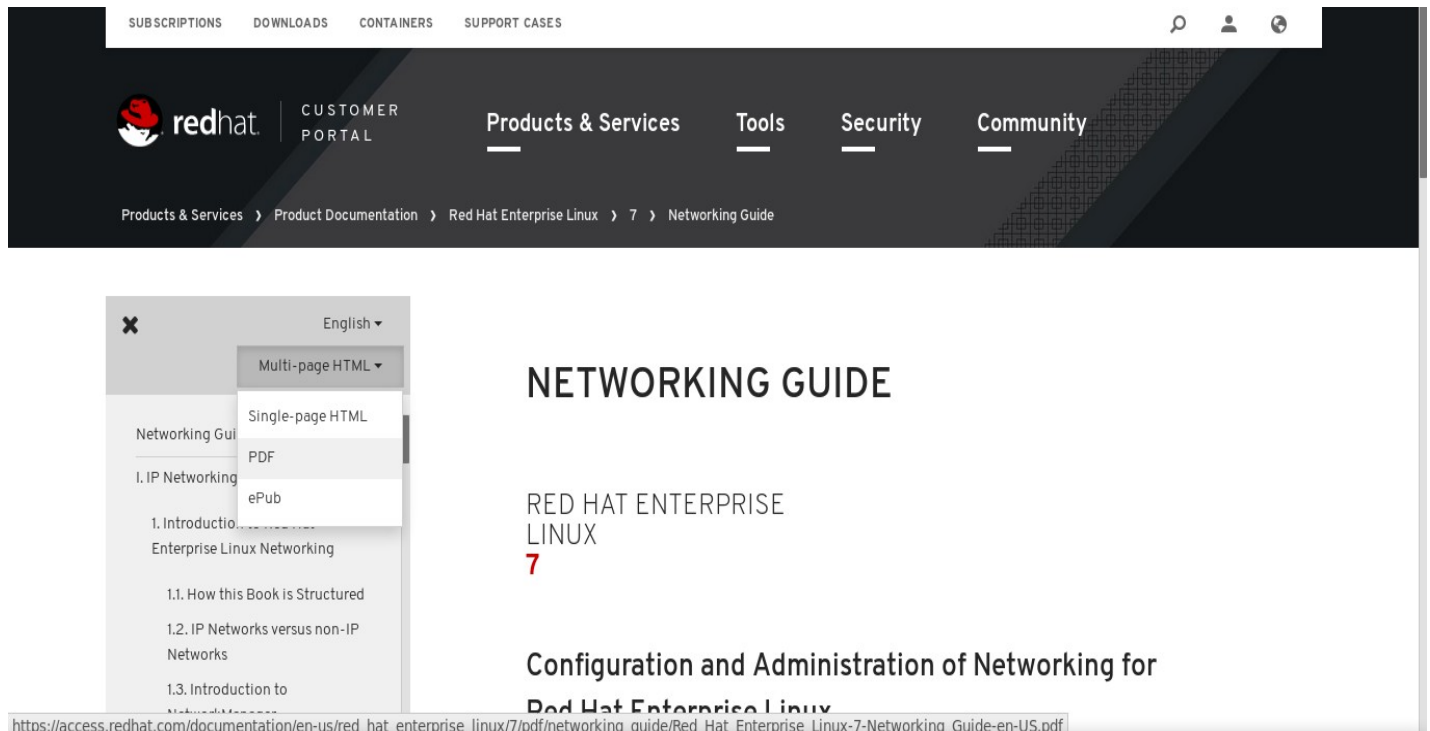
`usr/lib/systemd`

# 2–Network Basics

قبل ما نبدأ فى جزء ال Networking فى guide مجانى تبع redhat بيشرح  
المواضيع الخاصة بال Network ، هيفيدك جدا ، ود لينك ال guide

[https://access.redhat.com/documentation/en-us/  
red\\_hat\\_enterprise\\_linux/7/html/networking\\_guide/](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/)

وممكن تنزل ال guide بصيغة pdf لو حابب يعنى



SUBSCRIPTIONS DOWNLOADS CONTAINERS SUPPORT CASES

redhat CUSTOMER PORTAL

Products & Services Tools Security Community

Products & Services > Product Documentation > Red Hat Enterprise Linux > 7 > Networking Guide

English

Multi-page HTML

Single-page HTML

PDF

ePub

Networking Guide

I. IP Networking

1. Introduction to Enterprise Linux Networking

1.1. How this Book is Structured

1.2. IP Networks versus non-IP Networks

1.3. Introduction to Network Management

NETWORKING GUIDE

RED HAT ENTERPRISE  
LINUX  
7

Configuration and Administration of Networking for  
Red Hat Enterprise Linux

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/7/pdf/networking\\_guide/Red\\_Hat\\_Enterprise\\_Linux-7-Networking\\_Guide-en-US.pdf](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/pdf/networking_guide/Red_Hat_Enterprise_Linux-7-Networking_Guide-en-US.pdf)

نخش بقى فى جزء ال Network واول مصطلح عندنا هو ال NIC وده اختصار ل Network Interface Card اللى هو كارت الشبكة بتاعك

عندنا بقى فى اللينكس فى اتنين implementation بالنسبة لل Network الاول وده القديم وهو ال network العادى والثانى وهو ال Network Manager

بيقولك بقى ان الناس بتوع RedHat حبوا انهم يبنوا Modular System يعنى يبنوا system يكون عبارة عن Building Blocks جيب بعض وال Building Blocks هيسمحلنا اننا نعمل advanced tasks

طيب ال Network Manager بيسمحلك انك ت configure basic tcp ip يعنى IP Address Settings ، يعنى تحط ip address والموضوع انتهى على كده

او بيسمحلك انك تعمل advanced setting زى انك تعمل configure لحاجة اسمها ال VLAN

وبما ان systemd سهلت الدنيا شوية وبقيت بتعمل control لكل ال services عن طريق command واحد هو

systemctl

فكذلك الامر برضو بالنسبة لل Implementation بتاع ال Network Manager  
سهلت الدنيا وبقيت بتعمل control لكل حاجة عن طريق الامر

nmcli

وده اختصار ل

network manager command line interface

فى عندك بقى الامر ip address show من الاوامر اللى ممكن تختصر فيها  
يعنى ممكن تكتبه كده

ip addre show

او كده

ip a s

ال loopback هو ال virtual interface بتاعك وده كل وظيفته انه يقولك ان  
ال TCP/IP Stack شغال وطبعاً هيسمحلك انك ت run اى services بشكل  
Locally

بالنسبة لل Mac Address فانت ممكن تعتبره هو ال Physical Location بتاع  
البيت ده يعنى مش بيتغير ، انما انت لما بتحاول تغير ال mac address مثلاً  
فى حالة ال spoofing ده معناها انك كأنك بتحط عنوان وهمى على البيت  
بتاعك ، لكن هل ده غير المكان ال physical بتاع البيت ، لا طبعاً

ال 255.255.255.55 معناها كل ال Hosts اللى معاك فى الشبكة ، انما  
ip 0.0.0.0 اى

عندك قاعدة وهى ان المتغير حطه ديما بصفر والثابت حطله اقصى قيمة  
وهى 255 ، لو عايز تعرف اقصى عدد من ال ip فى subnetmask معين  
بتطرح من 256 مثلا ال subnet ده

255.255.255.192

اقصى عدد من ال ip ممكن يكون فيه هو 256 - 192 يعنى 64 ip

ال Gateway ممكن تعتبره البوابة اللى السيرفر هiestخدمها علشان يطالع ع  
النت ، طبعا عادى جدا يكون عندك اتنين راوتر متصلين ب switch واحد

طيب الطريق اللى بيطلعك لاي حاجة فى الدنيا يعنى يخليك تطالع على اى فى  
النت فى حالة لو عندك 2 راوتر متصلين ب switch واحد اسمه

Default Gateway ال

فى فرق بين المصطلح Default Gateway وبين المصطلح Default Route ،  
بالنسبة للمصطلح الاول فده بيتم استخدامه عادة مع ال Layer 2 Switches ،  
انما المصطلح التالت ده بيتم استخدامه عادة مع ال Layer 3 Switches ومع  
ال Routers ومع ال Firewalls كمان لانها بتقدر تعمل Layer3 Routing

وده معناه انه بيسمح للباكت انها تعدى تروح من شبكة وتروح للتانية ، على  
عكس ال Layer 2 Switches

طبعا انت لما تعدل فى connection عن طريق ال nmcli لازم تعمل لل  
connection ده down وبعدين تعمله up

ال ifconfig معناها Interface Config وكل ما تعمل Network Profile هتلاقى  
موجود فى

### **/etc/sysconfig/network-scripts**

وطبعا لو عايز تعرف ال Status بتاع ال Physical Interface بتاعك بيكون عن  
طريق الامر ده

```
mii-tool ens33
```

او عن طريق الامر ده

```
ethtool ens33
```

طبعا ال Full Duplex معناها انه ال Interface ده بيعت ويستقبل ، انما ال  
Half Duplex معناها انه يا اما بيعت يا اما ويستقبل

حكاية الرقم 4 فى ال IPv4 ده كان الاصدار الرابع من الايبى ، وكان فى فعلا  
اصدار خامس IPv5 بس مطلعش للناس وكان مقتصر بس على قد انهم  
يجربوه فى المعامل والابحاث

الراوتر وهو متصمم فى الاساس كان متصمم انه يعمل Forward من شبكة  
لشبكة تانية والراوتر بيعتمد على ال software فى انه يعمل ال forward ده  
علشان كده الكرنال بتاعت الراوتر كانت بتتفشخ  
انما ال switch ده optimized اصلا انه يعمل forward عن طريق ال  
Hardware ذاته نفسه يعنى يعمل ال forward من port ل port تانى

لو عندك جهاز راوتر مثلا ومش عليه ال DHCP فالاجهزة اللى على الشبكة  
هتروح تاخذ ال IP من حاجة اسمها ال APIPA ، بمعنى انت لو عندك كارت  
نت مثلا وحاول ياخذ من ال DHCP واصلا ال DHCP ده مش موجود ، ولو انت  
اصلا كمان معطتلوش IP Static فا هو اوتوماتيك هيروح ياخذ من ال APIPA كان  
ال ip اللى بياخده بيكون بدايته ديما

## 169.254

ال DHCP ده بيكون موجود فى ال Layer 3 Switches او الراوتر ، يعنى بيكون  
جوه الراوتر

ال APIPA هى اختصار ل Automatic Private IP Addressing

ولما بتضيف فى نص ال MAC Address ال fffe بيكون عبارة عن حاجة اسمها  
**EUI 64** اللى هى **Extended Unique Identifier**  
اى حاجة عندك بتبدأ ب FF00 اعرف انها بتروح لل Multi Cast

ال IPv6 عنده حاجة Equivalent لل Private IP اللى هى ال Local IPs  
الموجودة عند ال IPv4 اسمها بقى عند ال IPv6 هى **Site Local**  
وبكده ال site Local هيسمح للترافيك انه يعدى بين راوتر والتانى

عندك بقى اربع انواع من ال ips موجودة فى ال IPv6 ذات نفسه ، هم  
ال **link local** وال **site local** وال **Public IP Addressing**  
وعندك اخر واحد هو ال **MultiCast**

بيقولك بقى فى ال IPv6 مفيش حاجة اسمها DHCP وال IPv6 بينزل بالادوات  
بتاعته زى ال ping كده مثلا لو عايز تبج ipv6 اكتب ping6



## 2.1–Network Troubleshooting

خليك منطقى وانت بتعمل ال Trouble Shooting

واول حاجة فى ال Network Trouble Shooting وهى انك تعمل Check على ال Physical Connectivity ، طيب الامر ifconfig مش هيعرض كل الكروت اللى موجودة فى الجهاز ، بمعنى ان ممكن يكون عندك اكثر من كارت بس معمول ليه down او disabled فانت علشان تعرض كل الكروت لازم تكتب

### **ifconfig -a**

وطبعا حط فى اعتبارك ان ممكن يكون الكارت بايظ اصلا او بيعلق ، او ان الكارت ده مش Compitable مع ال Mother Board دى ، بس الحوار ده شبه انقرض لان كل الكروت بتتعرف اوتوماتيك

والطريقة الثانية من انك تتأكد بان الكارت متوصل بال MB وهى عن طريق الامر

**lspci**

وتشوف ال Ethernet Controller ، لو عايز تعمل check للكارت فيكون كمان عن طريق الامر ده

**mii-tool ens33**

الخطوة اللى بعد كده وهى انك تتأكد ان السيرفر بتاعتك واخد IP ، ومش مجرد انك تشوف السيرفر واخد ip ولا لأ ، دا انت تشوف اذا كان ال IP ده صحيح ولا لأ ، وهل ال IP هو Valid فى ال Subnet دى ولا لأ

نقول تانى ال subnet هو عبارة عن عدد الاجهزة المتاحة ، مثلا لو عندنا subnet mask ده اللى هو 255.255.255.0 فده معناه ان عندك 255 جهاز ممكن ياخدوا ip  
انما لو عندك ال subnet ده 255.255.255.255 فده معناه ان مفيش اجهزة ممكن تاخد IP خلاص كل الايبات اتحجرت

طيب انت هتستخدم الامر

`route -n`

علشان تشوف الاخطاء فين بالظبط زى مثلا ال Subnet Mask ، بمعنى اول ما يقولك ان ال Network Is Unreachable لازم تفكر ديما فى موضوع ال Subnet

وموضوع ال Packet Forward معناها ان ال Packet بتروح من Interface للتانى

يبقى اذا كل سيرفر هيكون عنده معلومات او خريطة عن فين الباكيت هتطلع من انهى Interface وهتروح لانهى interface بالظبط

ملحوظة جانبية وهى انك لازم تتقن التعامل مع ال Man Pages دى نقطة  
مفيهاش نقاش

ولو عايز جهاز ينادى على جهاز تانى فى نفس المكان ، ممكن تروح تعدل  
ملف ال

## **/etc/hosts**

وتكتب IP الجهاز واسمه وبالمناسبة انت ممكن تحط كذا اسم لاي جهاز

فى عملية ال resolve السيرفر بيروح الاول لملف etc/hosts وبعدين لو  
ملقاش بيروح ل

/etc/resolv.conf

والسبب فى كده ان انت عندك ملف تالت اسمه

etc/nsswitch.conf

الملف ده متعرف فيه هو هيروح يسأل انهى ملف الاول ، وخلي بالك اوى من  
الملف ده علشان بياجى منه اسئلة فى امتحان RedHat

عندك بقى اى حاجة بتعملها بره ال nmcli بيقولك انا معرفهاش ، فانت  
المفروض اى حاجة تبع ال Network تستخدم فيها ال nmcli

لو انت عملت disable لل Network Manager ساعتها بقى هتضطر تعدل فى ملف ال

etc/resolv.conf

ولما تيجى تسمى السيرفر بتاعك ، المفروض تسميه بكلمة واحدة بس مثلا server ، وبقية الاسم هتجيبوا من ال dns  
ومش لازم تعمل reboot للسيرفر لما تغير اسم الجهاز او ال hostname ،  
اعمل بس Logout وارجع تانى هتلاقيه اتغير

وبالمناسبة اى ملف مملوك للكرنال انت مش من حقك ومش هتعرف تعدل فيه ، زى مثلا الملف ده

**/proc/sys/kernel/hostname**

يعنى انت مش هتعرف تعدل اى حاجة بطريقة مباشرة فى المجلد proc ،  
وده بسبب ان المجلد ده عبارة عن virtual directory بي reflect الحاجات  
الى الكرنال بيعرفها عنك ، ولذلك لو عايز تغير حاجة يبقى تستخدم  
الامر **echo** وده الحل الوحيد ، وده مش معناه ان echo هو الى accepted ،  
لا لا زى ما انت عارف ال **redirection** هو الى accepted مش echo وبالتالى  
انت لو استخدمت اى امر تانى وعملتله redirect هيشغل برضو والسبب  
فى كده برضو ان ال developers سمحوا بكده علشان انت متضطرش انك  
تعمل reboot بسبب ان انت عدلت فى ملف ال kernel يعرفه

## 2.2–NIC Teaming

ملحوظة : متنساش انك تراجع ديما من ال documentations الخاصة ب RedHat واللى بتشرح ال NIC Teaming ، وده لينك ال documentations

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/7/html/networking\\_guide/ch-configure\\_network\\_teaming](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/ch-configure_network_teaming)

لما يكون عندك سيرفر وفيه كارت ethernet والكارت ده فى اكر من port وانت عايز تجمع البورتات دى، فا الطريقة اللى هتجمع بيها البورتات دى ليها اكر من شكل ، ولما تيجى تصمم اى solution لاي شركة startup لازم تحط فى اعتبارك العوائق المادية وغيرهم من الاعتبارات

ال slave هو ال physical port انما ال **VIF** اللى هى Virtual Interface هو ال Master بتاعهم ، هو اللى هيشيل الترافيك ويوزعه على البورتات المتصلة ببعض

فى مصطلحين وهم ال **kernel space** وال **user space** ، طيب ال kernel space معناها ان ال implementations بتاعتها يتم جوه الكرنال

عندنا بقى ال NIC Teaming هو user space كمان ال footprint بتاع ال NIC  
Teaming افضل من ال Bonding ، بمعنى ان ال Load بتاعه على السيرفر  
اقل من ال Bonding

ودول سببين مقنعين جدا ليك ك system admin انك تختار ال Teaming بعيدا  
عن ال Bonding

ملحوظة لو عملت ال up لل master interface كده ال physical interfaces مش  
هتبقى ال up

وده مثال على ال Bonding

```
nmcli con add type bond con-name bond0 ifname bond0 mode active-backup;  
nmcli con mod bond0 ipv4.addresses 192.168.40.1;  
nmcli con mod bond0 ipv6.method ignore;  
nmcli con mod bond0 ipv4.method manual;  
nmcli con add type bond-slave ifname ens4f0 master bond0;  
nmcli con add type bond-slave ifname ens4f1 master bond0;  
nmcli con mod bond0 + bond.options mii= 100  
nmcli con mod bond0 + ipv4.routes 192.168.40.0/24  
nmcli con up bond0;
```

لو عايز تعمل teaming يعنى تربط كارتين physical بكارث وهمى انت بتعمله ده المثال

```
[root@server-pc ~]# nmcli connection add type team-slave ifname ens37 master team0
```

Connection 'team-slave-ens37' (d47c8d71-81fe-4f30-ab79-fa6c35b18b54) successfully added.

```
[root@server-pc ~]# nmcli connection up team-slave-ens37
```

Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/30)

وبكده انت تقدر تستخدم ال Bonding او ال Teaming براحتك ، لكن طبعا ال Teaming احسن شوية ، ولازم تشوف حالة ال Virtual كارت عن طريق الامر ده

```
teamdctl team0 state
```

وخلى بالك ديما من ال Behavior بتاع ال VM workstation علشان بتستعبط احيانا

والامر بتاع **teamdctl** هينفعك ديما لو انت بتعمل trouble shooting او بتعمل Maintenance

وخلى بالك برضو من الامر ده **teamnl team0 ports** علشان تشوف حالة البورتات وسرعتها

ولو عايز تشوف حالة وتفاصيل كل team عندك بيكون عن طريق الامر

## **teamctl team0 config dump**

ويكون الناتج فى شكل JSON Format

```
{
  "device": "team0",
  "mcast_rejoin": {
    "count": 1
  },
  "notify_peers": {
    "count": 1
  },
  "ports": {
    "ens37": {
      "link_watch": {
        "name": "ethtool"
      }
    },
    "ens38": {
      "link_watch": {
        "name": "ethtool"
      }
    }
  },
  "runner": {
    "name": "activebackup"
  }
}
```



وده هيفيدك لو انت عندك كذا سيرفر وعايذ تعمل team ، فانت بدل ما  
هتروح على كل سيرفر وتعمل نفس الاوامر ، لا انت تروح تاخذ dump من  
السيرفر اللى انت عملته وتعمله import لكل السيرفرات

## 2.3–Network Bridging

ده لينك ال documentation بتاع RedHat

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/7/html/networking\\_guide/ch-configure\\_network\\_bridging](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/ch-configure_network_bridging)

خلى بالك ان ال switch بيطلق عليه Multi Port Bridge

طبعا مينفعش اى virtual machine هى اللى ت control ال Physical Interface ذات نفسه

وبالنسبة لطريقة عمل ال Network Bridge فهى نفس الطريقة بتاعت ال NIC Teaming وال Bonding وخلي بالك من الامر `brctl` برضو ولازم تقرأ ال documentations لان زى ما قولنا موضوع ال Network ده مفيهوش رعى كثير

## 2.4–Interface Naming

بص يا برنس اسم ال Interface هيكون معتمد على مكانه فى ال BIOS ، يعنى مثلا ال **eno** دى ال BIOS بيقرأها Ethernet on board يعنى ده كارت مدمج

مع ال Mother Board

انما ال **ens** ده كارت Ethernet PCI Express حرف ال s من اخر كلمة express

طبعا انت ممكن تجيب كارت يكون فيه port واحد مثلا او 4 بورتات مثلا ، طبعا لو فى بورت واحد يعنى interface واحد يبقى خلاص سميهولى بالاسم الطبيعى **ensX** حيث X هى رقم البورت

انما لو عندك كارت فيه اكثر من بورت ، قالك بس سميهولى بالاسم ده **ensxpy** حيث x هى رقم ال slot اللى متركب فيه الكارت ، وال y هو رقم ال port يعنى ممكن اسمه يكون كده

ens1p0

ens1p1

ens1p2

حيث s هو رقم ال slot ، وده هيفيدك بشكل كبير لانه هيكون عندك expected behavior لكل interface ، ولما تكتب الامر `lspci` هيقولك كل كارت فيه كام port ولو مقالکش يبقى اوتوماتيك تعرف ان فيه port واحد

لو عايز تعرف معلومات الهارد تبعك اكتب الامر ده

**`sudo smartctl -a /dev/sda`**

وخلى بالك ان انت عندك فى package بتحاول تخلى الكرنال يشوف الاسماء الى مبعتوعة من ال BIOS اي ويعملها Apply  
انما انت لو عايز الكرنال يشوف الاسماء القديمة اعمل reboot وبعدين اضغط حرف ال e علشان تعمل edit لل menuentry دى وتعالى على السطر اللي بدايته **linux16**

واكتب

`biosdevname= 0`

وبعدين اكتب جمبها برضو

`net.ifnames= 0`

وبعدين اضغط `ctrl + x`

طبعا انت لو عملت reboot للسيرفر اسماء ال interfaces هترجع زى ما كانت ، يبقى انت المفروض تقول لل bootloader متخليش الكرنال تاخد الاسماء اللى جايه من ال bios وخليها تاخد الاسماء القديمة ، يبقى انت هتعدل فى الملف اللى هو

**/etc/grub.fg**

او الملف

**/boot/grub2/grub.cfg**

طبعا متنساش انك بعد ما تغير ال interfaces للاسماء القديمة ، لازم ترجع تعمل reconfigure لل nmcli زى مثلا انك تعمل up للانترفيس وتديله مثلا ip او تخليه automatic

ولو عايز تجيب معلومات Interface كاملة عن طريق الامر ده

**ifconfig eth0**

طيب دلوقتى بقى لو قلتلك ترجم الاسم ده

**ens1p2**

فده معناه البورت رقم 3 فى ال slot رقم 1 لان رقم البورت بيبدأ من **صفر**

الكابلات اللى بتكون مش منظمة بين السيرفرات وبعضها او بين الراك اللى شايل السيرفرات اسمها **spider cables** انما انت بتشتري ال Patch Panel كأداة تنظيم بس علشان شكل الكابلات تكون منظمة اكثر

طبعا لو اتطلب منك انك توقف ال service بتاعت ال NetworkManager وتشغل ال service القديمة اللى هى ال network هتعرف؟  
اه بسيطة انت هتروح للمسار

**/etc/sysconfig/network-scripts**

وهتعمل ملف جديد باسم ال interface اللى عندك اللى انت هتجيبه من الامر  
**ifconfig -a**

والملف ده هيكون اسمه

**touch ifcfg-eth0**

وهتكتب جواه

Name= eth0

DEVICE= eth0

IPADDR= 192.168.60.200

NETMASK= 255.255.255.0

BOOTPROTO= static

GATEWAY= 192.168.60.1

DNS1= 8.8.8.8

وبعدین

**systemctl restart network**

طبعا وانت بتعمل ملف ال configuration بتاع ال ifcfg-eth0 ال syntax  
بتاعه سهل جدا عبارة عن **variable** ويكون equal to ل **value** وكل ال  
variables بتكون عبارة عن capital letters

**لحد هنا وكده احنا خلصنا كل المواضيع الخاصة بال Network**

**متنساش تبص على المصادر**

## 3-Setting GRUB Password + Getting Help

بص يا سيدى انت عندك ال **fdisk** دى utility بتتعامل مع  
ال mbr partition scheme ، دلوقتى بقى ال **gdisk** بقت هى البديل علشان  
هى هتتعامل مع ال gpt partition scheme

وبالمناسبة بما ان ال mbr سيكون جواه ال **first stage** من ال boot loader  
فاخد عندك كمان بقى اول sector 2047 من بداية الهارد بيكونوا محجوزين  
والسبب فى كده ان الناس اللى صمموا ال mbr قالوا خلوا المساحة فاضية  
يمكن نحتاجها ، وفعلًا بيكون عندك ما يسمى بال Stage 1.5 من ال boot  
loader ، يعنى المرحلة الاولى بتقوم وبعدين بتسلم المهمة للمرحلة اللى هى  
**واحد ونص** وبعدين المرحلة دى بقى بتروح تكمل وتقرأ ال Second Stage  
اللى موجودة فى ال boot/ ، وبكده انت فعليا عندك 3 مراحل لل boot loader

لو عايز تعرف اكر من ال 3 مراحل بتوع عملية ال boot ، اقرأ ال  
documentation ده الخاص بشركة IBM هتعرف فيه اكر عن عملية ال boot

<https://www.ibm.com/developerworks/library/l-linuxboot/index.html>



بالمناسبة كمان ، هما اصلا لما صمموا ال mbr ، راحوا على الهارد وخلوا اول sector 2048 فاضيين ، وراحوا حطوا ال mbr كله فى ال sector رقم 0 وبعدين بقى قالوا خلوا ال 2047 من رقم 1 لحد 2047 يكونو reserved

حتى كمان لو عايز تتأكد هتلاقي ان اول بارتيشن فعليا بيبدأ من ال Sector رقم 2048 ، اكتب الامر

```
fdisk -l /dev/sda
```

وده معناه ان من صفر ل 2047 مركون على جمب

طيب ازاي بقى هتعمل reinstall لل first stage ، بص انت عندك طريقين ، اول طريق انك تدخل اسطوانة وتختار منها rescue centos مثلا على حسب نوع التوزيعة

والطريقة الثانية انك وانت فى ال **syslinux** تضغط esc مرتين ، وتدخل على text mode بدل ال gui وتكتب **linux rescue** ، ده طبعا شبيه ب linux text

طيب خيلنا فى الطريقة الثانية ، دلوقتى هو هيحاول ي detect اذا كان فى اى distro موجودة ولا لأ طبعا ، لو لقيها هيعملها mount فى المسار

```
/mnt/sysimage
```

طبعا هو دلوقتى بيستخدم ال root file system بتاع الاسطوانة ، واحنا هنا عايزينه يستخدم ال root file system الحقيقى يبقى المفروض نستخدم الامر

**chroot /mnt/sysimage**

دلوقتى هتضغط enter وبعدها هتعمل check على ال root file system الحقيقى بتاعك عن طريق

**ls /mnt/sysimage**

لازم تحط فى بالك وتشوف ال

/boot

معمول ليه mount فى انهى بارتيشن ، وهنا بقى لما تيجى تعمل reinstall لل grub او ال first stage انت هتعملها على ال MBR مش هتعملها على البارتيشن بتاع ال root ، طبعا فى حالة ال efi الوضع هيفتخلف

يبقى انت هنا هتعمل install لل grub على الهارد كله ، يعنى هتعمله على ال mbr ، لان ال Mbr خارج نطاق البارتيشنز ، ولازم تخلى بالك انت هتستخدم انهى اصدار من ال grub والامر كالتالى مع الاصدار الاول

**grub-install /dev/sda**

ومع الاصدار التانى هيكون كده

**grub2-install /dev/sda**

وبعدها بقى اكتب exit واعمل reboot

والطريقة دي برضو ممكن تعمل بيها recover لل root passwd لو نسيته  
وطبعاً بعد ما تعمل reinstall لل first stage او لل grub ال SELinux هتروح  
تعمل relabel لل file system بتاعك من تانى

الموضوع اللى بعد كده وهو موضوع اننا نعمل passwd لل grub ، علشان  
اي حد يفكر انه يعدل فى ال grub يطلب منه باسورد ، طيب فى طريقتين ،  
الاولى سهلة والثانية manual  
الطريقة الاولى وهى عن طريق الامر

`grub2-setpassword`

ومتنساش تعمل **regenerate** لملف ال grub.cfg  
وخلى بالك لانه فى ال grub مش هيقبل منك غير ال root ك username  
وهيقبل منك الباسورد بتاع ال grub اللى انت عملته .

عندنا ملحوظة مهمة جداً وهى انك لما عملت regenerate لملف ال grub ، كده  
التعديلات بتاعت ال

`biosdevname= 0`

وال

`net.ifnames= 0`

اتلغت ورجعت الكرنال تاخذ الاسماء اللى ال BIOS هيبتهاها بخصوص ال  
Network Card Interfaces ، فكه ال **ssh** ممكن ميشتغلش لان ال Interface  
خد IP جديد ، وطبعال **rhgb quiet** هترجع هي كمان

طيب انت دلوقتي لو عايز تثبت الكلام اللى انت عدلته في ال grub بحيث  
انك حتى لو عملت regenerate لل grub تفضل الاعدادات بتاعتك زي ما  
هي ، كده انت هتروح تعدل الملف اللى موجود في

### **etc/sysconfig/grub**

وهتيجي على السطر ده

```
GRUB_CMDLINE_LINUX="rd.lvm.lv=centos_server-pc/root  
rd.lvm.lv=centos_server-pc/swap "
```

وهتضيف بعد كلمة swap كلمة **biosdevname= 0** وبعدها **net.ifnames= 0**

ملحوظة انت لما استخدمت الامر

**grub2-setpassword**

كل اللى حصل انه عملك ملف اسمه **user.cfg** والملف ده كل اللى موجود

فيه هو الباسورد بتاع ال grub بشكل **hashed**

والباسورد بتكون **one way hash** يعني انسى ان حد يرجعها وكمان بيستخدم

ال 160 bit في التشفير

واللى حصل كمان انك لما استخدمت الامر ده وعملت passwd لل grub ،  
راح رايح للملف بتاع ال

**/boot/grub2/grub.cfg**

وراح ضايف if statement

```
### BEGIN /etc/grub.d/01_users ###  
if [ -f ${prefix}/user.cfg ]; then  
    source ${prefix}/user.cfg  
    if [ -n "${GRUB2_PASSWORD}" ]; then  
        set superusers="root"  
        export superusers  
        password__pbkdf2 root ${GRUB2_PASSWORD}  
    fi  
fi  
### END /etc/grub.d/01_users ###
```

وكمان راح عمل ال default user هو ال root وبعدها هيحاول يعمل load  
للباسورد بتاع ال grub من الملف

**/boot/grub2/user.cfg**

معلومة ليك بقى وهى ان ال grub بي support ال hashed passwd وال plain text بس طبعا ال plain text مش recommended خالص

طبعا فى حاجة Advanced وهى انك ممكن تخلى يوزر معين هو اللى يكون ليه صلاحية على ال entry المعينة اللى انت بتحددها فى ملف ال

**/boot/grub2/grub.cfg**

وكلمة

## **--unrestricted**

معناها ان اى حد ممكن يأكسس ال entry دى ، الكلام ده فى الفيديو رقم 46 اليوم رقم 22

طيب فى حالة انك نسيت الباسورد بتاعت ال grub ، بكل بساطة دخل اسطوانة وادخل فى ال rescue mode واعمل

**chroot /mnt/sysimage**

وبعدها افتح ملف ال grub.cfg

وبعدين روح على السطور اللى فيها الباسورد بتاعت ال grub وامسح الكلام ده كله مش عايزينه واعمل exit وبعدها reboot ، وبعدها بقى لو عايز تعمل reset للباسورد بتاعت ال root خش عدل فى ال grub واكتب rd.break

## نخش بقي في ال Man Pages

قولنا ان ال option بيغير ال behavior بتاع ال command انما ال argument ده عبارة عن معلومة لل command

طيب مبدئيا كده ال man وهو بيتعمل او لما جم يعملوه قسموه من جوال sections والناس بتوع **solaries** بيسموه **chapters** الاتنين واحد يعنى وكل جزء او section بيوصف مجموعة حاجات شبه بعض او اوامر زي بعض

طيب لو عايز تعرف الاوامر موجود في انهي section ، او قصدى يعنى لو عايز تعرف ال sections اللى عندك ، فانت هتكتب

man man

اذا لو انت بتدور على configuration file هتدور ديما في section رقم 5 وبكده انت مطالب انك تعرف على الاقل ال section رقم 1 ورقم 5 ورقم 8 لان اغلب شغلك هيكون معاهم

طيب فرضا بقي انت عارف ال Command بيعمل اي ، لكن ناسى اسم ال Command ذات نفسه ، يبقى انت هتستخدم ال search ، فرضا مثلا انت عايز تجيب الاوامر المتعلقة بال date ، يعنى الاوامر اللى بتتعامل مع ال date يبقى هتستخدم الامر بالشكل ده

man -k date

وتدور بقى فى الناتج اللى طلعلك عن الامرالى ممكن ينفذلك الحاجة اللى  
انت عايزها

طيب لو عايز ال man يعرضلك النتائج واحدة بواحدة زى كأنه slides ولما  
تخلص تضغط q ، بيكون عن طريق الامر

## man -k ls

و متنساش حرف ال K بيكون capital لو عايز النتائج تتعرض فى شكل slides  
انما لو كتبته small هيعرضلك كل النتائج تحت بعض

بالنسبة لحرف ال K ال Capital هو بيعرضلك النتائج اللى جاية ، يعنى بيقولك  
اي اللى هيعرضهولك ، لو عايز تعمل skip ، اضغط ctrl + d

طبعا لو نزلت لآخر الصفحة فى ال man page وعايز تروح لاول الصفحة  
اضغط g ولو عايز تنزل لآخرها خالص اضغط G

ولو عايز تعمل search على كلمة معينة فانت بتستخدم ال / وبعدها تكتب  
الكلمة ، ولو عايز تروح للنتيجة اللى بعدها اضغط n وطبعا لو عايز تروح  
للنتيجة اللى قبلها هتضغط N

فى بعض الاوامر اللى هتحتاج فيها explanation اكتر ، ساعتها هتروح  
بقى لل info pages



يقولك بقى ان اى package بيتعملها install بينزل معاها doc file بيكون موجود فى

## **/usr/share/doc**

وطبعا هناك بقى لازم تقرأ ال **README** لانه مهم وده بيعلمك ازاي تعمل compile لكل حاجة بنفسك وحاجات تانية كتيرة

وكمان فى بعض ال services اللي انت بتسطبها ، هيديك ملف مساعد اسمه **example** علشان تقدر من خلاله انك تعمل configure لل service دى اذا خليك فاكر المكان ده كويس اوى ، علشان هتستفيد منه كتير

الطريقة اللي بعد كده وهى انك تروح لل Documentations الخاصة ب redhat عن طريق اللينك ده

**<https://www.redhat.com/docs>**

طبعا افضل documentation ممكن تذاكر منها هى ال Documentation بتاعت redhat

## آخر جزء عندنا وهو ال history

لو انت كتبت !! فى الترمنال هينفذك اخر امر انت كتبتة ، وبما ان الاوامر فى ال history بتكون مترتبة ، فانت لو عايز تنفذ الامر رقم كذا اضغط

!255

وبعدين رقم الامر ، هنا مثلا هو هينفذ ال Command رقم 255

طيب لو عايز تنفذ اخر امر كان ببدا بحرف ال i مثلا اضغط

!i

وخلى بالك ان اى حاجة بتكتبها فى ال history بتتخزن فى ال buffuer او ال ram ، طيب امتى الاوامر دى بتنتقل من ال ram وتروح للملف

.bash\_history

اول ما بتعمل logout بياخد اللى موجود فى ال ram او ال buffer ويحطه جوه الملف ده

وطبعا لو الكهرياء قطعت ، فكده الكرنال مش هتقدر تاخذ اللى موجود فى ال buffer وتكتبه فى الملف ده

انما بقى لو عايز تمسح ال history فا هيكون عن طريق الامر

**history -c**

بس خلى بالك ان ده هيمسح الاوامر اللى موجودة فى ال buffer بس

لكن لو عايز تمسح اللى موجود فى الملف `bash_history` يبقى برضو لازم  
تعمل `logout`

طيب لو مش عايز تعمل `logout` يبقى زى ما انت عارف تستخدم  
ال `redirection` علامة واحدة بس هتعمل `overwrite` لكل اللى موجود فى  
الملف ده

ولو عايز تعرف ال `History` بتاعك بيشيل لحد كام امر عن طريق الامر ده

**`echo $HISTSIZE`**

وطبعا لو مش فاكّر فاستخدم ال `tab` ديما

**لحد هنا احنا خلصنا الجزء الثالث ده الخاص بالتراك التانى**

# 4-RAID

فى البداية كلمة RAID هى اختصار ل

## **Redundant Array Of Independent Disks**

او هى اختصار ل

## **Redundant Array Of Inexpensive Disks**

السؤال هنا بقى ليه فى مصطلحين لل RAID ؟ بص اول ما تسمع كلمة

## **Independent Disks**

فده معناه انى بتكلم عن ال Hardware RAID ، انما كلمة Inexpensive فدى

معناها انى بقصد ال Software RAID

طيب كلمة **redundancy** معناها ان الداتا موجودة اكتر من مرة ، وترجمة

كلمة RAID **معناها انها مجموعة من ال disks المنفصلين عن بعض فى**

**شكل lists**

دلوقتى انت لما بتيجى تكتب الداتا بتكتبها الاول على ال cash space بتاع ال

RAID Controller ، وبعدين يبدأ ينقلها على الهارد

وطبعا كل RAID Controller يسمح لك انك تركيب مجموعة disks محددة ، ولما هتعمل raid array لازم كل ال disks تكون متصلة على نفس ال RAID Controller ، يعنى مينفعش تعمل list بين 2 هارد موجودين على RAID Controller و 2 تانيين فى RAID تانى خالص ، والسبب فى كده ان كل RAID منهم ليه cpu خاص بيه وليه cash memory خاصة بيه

وبالمناسبة يعنى ال Hardware RAID احسن وافضل بكثير من ال Software RAID

وخلى بالك ان كل RAID Controller يسمح لك انك تركيب نوع معين من الهاردات زى ال ssd مثلا

طيب عندك بقى ال RAID وهو بيشتغل ، بيشتغل بحاجة اسمها ال

#### RAID Level

عندك بقى ال RAID 0 بيكتب بشكل **Stripping** ، يعنى بيقطع الداتا وهو بيكتبها ، يكتب شوية على الهارد ده وبعدين شوية على الهارد ده وبعدين شوية على الهارد التالت وهكذا ، وبكده ال RAID 0 سريع جدا فى الكتابة وسريع جدا فى القراءة لانه بيوزع ال load على الهاردات الموجودة

وبالنسبة للعدد فانت اكثر حاجة ممكن تستخدمها مع ال RAID 0 هو unlimited طالما ال Machine بتاعتك بتسمح بكده ، وحاجة كمان وهو ان ال RAID 0 لا بيستخدم ال **Mirroring** ولا بيستخدم ال **Distributed Parity** ، دا بياخد الهاردات كلها مع بعض وبسيحها ويخليهملك كلهم هارد واحد

تانى نوع من ال RAID عندنا ، هو ال **RAID 1** وده بيعمل حاجة اسمها **Mirroring** ، يعنى الداتا اللي كتبها هنا هيروح يكتبها على الهارد التانى ، وبالتالي السرعة هتكون قليلة جدا بالنسبة للكتابة ، على عكس سرعة القراءة اللي هتكون عالية جدا لانه بيقرأ من 3 هاردات مع بعض او هاردين على حسب العدد اللي موجود ، وطبعاً اقل عدد من الهاردات اللي ممكن تستخدمها مع ال RAID 1 هو 2 واكثر عدد هو unlimited بس برضو على حسب ما ال RAID Controller يسمحك ، لازم تحط فى بالك ان لو عندك 3 بارتيشنز ، فالمساحة بتاعت ال Array هتكون مبنية على اصغر مساحة بارتيشن فيهم

النوع الثالث وهو ال **RAID 5** ، وده بيعمل حاجة اسمها **Stripping with Distributed Parity** وال parity دى عاملة زى ال check sum يعنى بيستخدمها علشان يرجع بقية الداتا ، مثلا لو عندك abc وكتب هو a و b فكده هو ممكن يستخدم ال b علشان يرجع ال c ، الموضوع وما فيه عبارة عن معادلة ، عندك قيمتين ، وبالتالي تقدر تجيب الثالثة و اقل عدد ممكن تستخدمه مع ال RAID 5 هو 3 هاردات ، وبالنسبة لل read speed هتكون معقولة ، وبالنسبة لل write speed بتاعته هتكون شغالة برضو يعنى مش حلوة اوى ومش وحشة اوى ، والسبب فى ان عملية الكتابة مش optimal فيه ، لان انت لما يكون عندك abc فا هو هيروح يكتب a وبعدها ال b والمرحلة الاخيرة اللى هيكتب فيها ال parity هتاخذ وقت لانها عملية xor ، عملية حسابية شوية ، وبالتالي هى عملية مرهقة جدا ، وهى كمان مكلفة جدا بالنسبة للجهاز بتاعك وخصوصا ال CPU ، اه حاجة اخيرة بخصوص ال RAID 5 ، وهى ان ال RAID 5 بيسمحلك ان Disk واحد هو اللى يحصله Failure ، ولو حصل ان disk حصله fail ، فالحل انك بكل بساطة تشيله من ال Array وتحط واحد جديد مكانه ، طيب اى اللى يحصل هنا بالضبط؟؟ كل اللى يحصل ان ال RAID 5 هيسخدم الهاردات اللى موجودة عنده علشان يشوف ال data ال missing ويبدأ انه يرجعها ، زى ما قولنا فى المعادلة فوق

النوع الرابع من ال RAID وهو ال **RAID 6** ، وده بيعمل حاجة اسمها

### Stripping With Dual Distributed Parity

يعنى بيعمل dual من ال check sum ، وعندك بقى العدد الاقل من الهاردات  
اللى ممكن تستخدمها مع ال RAID 6 هى 4 هاردات ، وهنا هو هيكيب نسختين  
من ال parity ، وطبعاً من اهم مميزات ال RAID 6 انه بيسمحلك ان يكون  
عندك 2 هارد هما اللى يحصلهم Parity  
طيب ملحوظة مهمة ليك لما تيجى تعمل RAID 6 ، حاول تخلى عدد الهاردات  
تكون **even** يعنى زوجية

اخر نوع عندك وهو **RAID 10** وده Complex شوية ، لانه بياخد من RAID 0  
ومن RAID 1

وهنا بقى لازم تفرق بين مساحة ال **Array** او ال **list** ، وبين

ال **usable space** بتاعت ال **Array**

بطريقة اخرى لازم تفرق بين ال **raw space** قبل ما تدخلها جوه ال **Array**

وبين ال **usable space** ب بعد ما تدخل جوه ال **Array**



طيب فرضا بقى انك عملت ال RAID Array ، قولى بقى هتبدأ تكتب على انهى هارد فيهم ؟؟ مش كده وبس طب انت اصلا لما تيجى تعمل file system هتعمل لانهى واحد فيهم بالظبط يعنى هل هتعمل لواحد الاول وبعدين تعمل للتانى ولا اى الدنيا بالظبط ؟؟

بص بكل بساطة انت هتضرب ال 3 فى الخلاط وتطلع device جديدة اسمها ال **RAID Device** ، يعنى هتعمل **Virtual RAID** وبكده انت لما تيجى تتعامل هتتعامل مع ال RAID Device دى ، وهى بقى اللى هتتولى المسؤولية انها تكتب الداتا على انهى واحد فيهم ، وبالمناسبة كمان انت لما تعمل format فانت هتعمل format لل RAID Device دى ، وبرضو كذلك الامر لما تعمل mount هتعمل mount لل RAID Device دى ، وبالتالى ال Virtual Device هتعمل represent لل 3 هاردات دول مع بعض ، وخلي بالك ان virtual يعنى مش physical مش موجودة فى الحقيقة

نيجى بقى لل implementation بتاع ال Software RAID وهى اسمها

**mdadm**

ودى اختصار ل

## **Multiple disk administration tool**

وبتسحملك انك تعمل create وتعمل manage لل RAID Device اللى عندك ، كده انا لما اعمل Implement لل RAID Device يعنى معناها انى ه create Virtual RAID وهقولها ال RAID Device بتاعها كام بالظبط

وكمان هقولها ال Devices اللى هى مسئلة عنها مين ومين بالظبط ، وبكده انا هعمل File System لل Virtual RAID دى وهى مسئلة انها تكتب الداتا على بقية الهاردات

فى امر جميل جدا وهو **watch** بيخليك تشوف ال process وهى شغالة ، انت ممكن تستخدمه زى الامر time لو عايز تعرف مثلا الحاجة دى خدت وقت قد اي زى ال compile بتاع الكرنال استخدم watch مع ال

### **watch cat /proc/mdstat**

طبعا ده هيخليك تشوف ال Array وهى بيتعملها Build وهيجهلك 3 حروف وهم UUU معناهم ان الثلاثة **UP UP UP**

طيب انت كده لو فى هارد فى ال array حصله **failure** ، وجيت انت ضفت هارد لل array مرة ثانية وجيت نفذت الامر

### **watch cat /proc/mdstat**

فاللى انت شايفه ده هو ال Virtual RAID بيعمل rebuild لل array لان انت ضفتله هارد جديد ، يعنى بيروح يشوف اى الداتا اللى اتكتبت لما الهارد الجديد كان موجود او لما كان حصله fail وبالتالي هو هيعمل create لل Parity ويعمل generate للداتا مرة ثانية ويكتبها على ال disk اللى انت لسه ضايفه ده

فرضا بقى انك جيت فى وقت وخلص مش عايز تستخدم ال array او ال Virtual RAID دى ، بكل بساطة اعمله umount وبعدين نفذ الامر ده

**mdadm --stop /dev/md0**

وبعدين

**cat /proc/mdstat**

وطبعا مش معنى انك وقفت ال RAID يبقى الداتا راحت ، لا طبعا

طيب بعد ما وقفته لو عايز ترجعه ، يبقى هتعمله حاجة اسمها assemble  
اعادة تجميع عن طريق الامر

**mdadm --assemble --scan**

كده هو هيروح يدور على ال Virtual RAID ويجمعهم تانى ويشغلهم كمان

**cat /proc/mdstat**

وبكده انت لما توقف ال Array او ال Virtual RAID ، ده معناه انك مش  
هتشوف ال device اللى بتعمل manage لل array دى ، وممكن كمان تثبت  
ال mount بتاع ال Virtual RAID فى ال fstab

ولما بتعمل stop لل md0 بتبقى بخ خلاص يعنى بتختفى ، وبيرجع كل هارد  
standalone لوحده ، كل حته لوحدها

وبالمناسبة انت اصلا لما بتعمل read array ، هو بيروح اوتوماتيك لكل هارد منهم ويقول له يلي اسمك sdb انت والاتنين اللي معاك ، انتو كده member فى ال array اللي اسمها md0

وبكده انت تخلي بالك من ال

## **--zero-superblock**

واصلا اصلا ال Software RAID هو عبارة عن Module معمول ليه Implement فى الكرنال ذات نفسها

وطبعا كل RAID Controller ليه سرعة معينة و كاش ميمورى ، ولو هتعمل مقارنة بين ال Hardware RAID وبين ال Software RAID ، فانت عندك ال Hardware RAID مينفعش تقارنه بال Software RAID ، وخط فى بالك ديما ان ال Hardware RAID هو افضل ديما من ال Software RAID

دخلنا بقى على موضوع انى اتعامل مع البارتيشنز واخلوها RAID ، اول حاجة هتعمل بارتيشن عن طريق الامر

**fdisk /dev/sdb**

وبعدين تغير ال type او تغير ال system id بتاع البارتيشن وتخليه

**linux raid autodetect**

اضغط t علشان يعرضلك كل الاوامر

فى حكاية ال software raid لما تيجى تعمل بارتيشنز ، ابقى خلى المساحة موحدة ، يعنى لو هتعمل بين بارتيشن فى هارد والبارتيشن التانى فى هارد تانى ، تخلى حجم الاتنين قد بعض

كمان ممكن تعمل label لبارتيشن ال File System بتاعه ntfs عن طريق الامر

**ntfslabel /dev/sda1 'Apple'**

بدل ما تستخدم ال Gparted

واوعى تغلط غلطة كبيرة زى انك وانت بتعمل create لل Array ، انك تستخدم بارتيشن يكون مستخدم اصلا مع Array تانية

وطبعا لو عملت reboot وجيت تشوف ال **lsblk** ، هيوريك كل بارتيشن هو member فى انهى array بالظبط

وديما اعمل mount عن طريق ال UUID ، وبالتالي لما تيجى تتعامل مع ال fstab اتعامل بال UUID ، وطبعا لو لقيت فى ال fstab اى بارتيشن النوع بتاعه linux raid يبقى تعرف على طول انه جزء من raid array

اخر حاجة لو عايز تتعامل مع بارتيشنز ال File System بتاعهم غير ال ntfs يبقى تستخدم ال **e2** وتضغط 2 tab

# 5-LVM

ملحوظة مهمة جدا ليك وهى ان ال Storage بتاعتك لازم تكون **Flexible** ولازم يكون فى Backup ديما وباستمرار ، وخلي بالك كمان ان اخطر حاجة تخص ال Storage وهى انك تعمل resize لل File System ، واصلا اصلا عملية ال resizing ل Fixed Partition دى فى حد ذاتها مصيبة ، طيب اى الحل بقى ؟؟

بيقولك الناس بتوع اللينكس جوا من فترة كده وعملوا حاجة اسمها ال **lvm** وده اختصار ل Logical Volume Manager ، اى بقى حكايته ؟؟؟

بص اول ما اقولك Logical يبقى تنسى خالص اى Physical Limitations ، بمعنى ان انت عندك مثلا هارد 500 جيجا ، طبعا ال 500 جيجا دول physical ، تمام

دلوقتى بقى انت عايز تكتب داتا حجمها 600 جيجا ، طبعا مش هينفع ، والحل هنا انك تستخدم حاجة زى ال LVM ، طيب ازاي ، بيقولك هات هارد 500 تانى وحطه جنبه وضم الاتنين على بعض بشكل logical وبكده يكون عندك 1000 جيجا ، ودى اهم ميزة فى ال LVM

طيب الفكرة كلها فى ال LVM هى ان بيكون عندك اكر من هارد ديسك او بارتيشن ، واصلا ال LVM مش بيضم اى حاجة Physical ، يعنى مش بيتعامل مع اى حاجة تكون Physical ، اومال هيفهم اى؟؟ هيفهم logical فقط ، مهو لما يضمك اكر من هارد ديسك مش هيضمهم Physical ، لا هيضمهم بشكل Logical

برضو دلوقتى الحاجة ال Logical الللى عندى دى هل هى Disk؟؟  
طبعا لا بقى اسمها Volume ، يبقى انا هعمل عملية Convert واخلى ال Physical Disk اقلبه ل Logical Volume ، بعد ما حولت كل الهاردات الفيزيكال الللى عندى ، هخلى ال LVM يجمعهملى مع بعض ، وبكده لما انا جمعتهم مع بعض بقى اسم كل واحد فيهم Logical Volume انما كلهم مع بعض اسمهم Volume Group

فاكر فى ال RAID لما كنا بنقول اننا لما نيجى نعمل RAID ، المفروض ان البارتيشنز او الهاردات الللى هنعمل عليهم ال RAID يكونوا كلهم مساحة واحدة ، طبعا الكلام ده بيكون فى ال RAID ، انما بقى فى ال LVM الوضع مختلف شوية ، مش هيفرق معاه اذا كان فى بارتيشن مساحة 50 جيجا وواحد تانى 180 جيجا وواحد تالت مساحته 400 جيجا ، لانه اصلا ال LVM مش بيفرق معاه المساحات ، طب ليه المساحات مش بتفرق معاه؟؟ انت عارف اصلا ال LVM شغال ازاي اصلا؟؟ انا لما قولتله جمعلنى البارتيشنز دى وخليها لى Volume Group ، راح جى على اول بارتيشن مثلا وليكن



مساحته 5 جيجا وراح مقسمه ل تحت صغيرة ، وقسمه تحت بالتساوى  
كمان ، كل حته قد الثانية ، وراح برضو للبارتيشن التانى وقسمه برضو بنفس  
التساوى بتاع الاولانى بالظبط ، وكل حته او قطعة منهم  
اسمها **Physical Extent** يعنى من الاخر كده هو يجمع البارتیشنز اللى  
عندك كلهم ويضربهم فى الخلاط ويطلعهم لك مساحة واحدة ، وبكده انت تقدر  
انك تعيد تقسيم المساحة دى من تانى

طيب لما ياجى يكتب الداتا بقى ياريس هيكتبها ازاي؟؟ هنا بقى هو هبدأ يملأ  
من اول Extent فاضية عندك ، وده معناه ان كل Logical Volume انت عملته  
بيمثله عدد معين من ال Extents ، طبعا كل Extent انت عملتها ليه Size ،  
واصلا ال Default Size بتاع كل واحدة فيهم لو انت مقربتلهمش وانت بتعمل  
ال LVM **بيكون حجم الواحدة 4 ميغا بايت** وده معناه ان انت لو عندك بارتیشن  
مساحته 7 جيجا فانت محتاج بما ان الجيجا بتساوى 1024 ميغا فانت محتاج  
7168 / 4 بيساوى 1792

واصلا اصلا كل Physical Extent بيحتوى على مجموعة Blocks ، طيب افرض  
بقى ان جه واحد وحب انه يتذاكى وراح قبل ما يعمل اى Logical Volume او  
قبل اى Volume Group وراح وخلي مساحة البارتیشن مثلا 510 ميغا وليس  
جيجا ها ، كده انت عندك 510 / 4 طلع عندك 127.5 ، وهنا نطلع بقاعدة

وهي ان مفيش حاجة اسمها نصف Extent ، هو يا اما خدت Extent كاملة ، يا اما باقى المساحة بقت wasted

وبكده انت تحط فى الاعتبار انك لما تعمل allocate لمساحة بارتيشن معين تخلق المساحة دي تكون بتقبل القسمة على مساحة ال Physical Extent دي ايا كانت بقى هل هي 4 ميجا ولا انت عملت Custom Physical Extent ، اهم حاجة الناتج يكون صفر ، لان لو فيه كسور فالمساحة دي هتعتبر wasted يبقى كده المساحة دي هتروح على الفاضى يعنى تخلق بالك حجم البارتيشن هيبقى قد اي وحجم ال Extent Size هيبقى اى اصلا

ملحوظة كلمة تحول البارتيشن ل volume ، اصلا volume معناها حجم او مساحة ، فكلمة انك تحول البارتيشن ل volume ، يعنى بدل ما انت بتتعامل مع البارتيشن كانه Physical Component لا انت هتتعامل معاه كمساحة او حجم

وهنا بقى بعد ما انت عملت ال Volume Group ، ابدأ بقى قسمهم زى ما انت عايز ، وكل بارتيشن منهم هيبقى اسمه Logical Volume وركز كمان زى ما قولتلك ان ال LVM مش بي فهم ال Physical Partition ، لازم تحوله من Physical Partition ل Physical Volume ، ايوه زى ما بقولك كده

اتعامل على مستوى المساحات مش على مستوى ال Physical Component  
ذات نفسه  
يبقى السيناريو هيتم كالتالى

**Physical Partition -----> Convert To Physical Volume**

**Create Volume Group From -----> Physical Volume**

**Create Logical Volume From -----> Volume Group**

**Create File System For -----> Logical Volume**

**Mount The Logical Volume**

يبقى كده خليك فاكر انك من اول ما تعمل ال logical Volume ، تنسى  
البارتيشنز الاصلية اللى كانت موجودة

وكمان افكر ان ال Hardware RAID بياخد الهارد ديسك كامل ، انما ال  
Software RAID ده غيره ممكن ياخذ جزء من الهارد يعنى بارتيشن مثلا

وبعدین انت ممکن تحول بارتیشن من هارد دیسک مع هارد تانی کامل عن طریق انک تخلي الاتین Physical Volume عن طریق الامر

**pvcreate /dev/sdb1 /dev/sdc**

وبعدین تعمل create لل Volume Group عن طریق الامر

**vgcreate data ( Group Name ) /dev/sdb1 /dev/sdc**

خلى بالك برضو ان ال man page مش فى كل الاوقات بتكون human readable خصوصا لوانت بتستعمل برنامج او اداة جديدة عليك

بعدین بقى هتعمل logical Volume من ال Volume Group عن طریق الامر

ده

**lvcreate --size 22G --name oracle data**

طبعا انت لما بتعمل volume group بيت create مجلد باسمها فى

**/dev**

وبعدین راح نعمل create ل File System لل **Logical Volume** اللى احنا عملناه وسميناه باسم **oracle** عن طریق الامر ده

**mkfs.ext4 /dev/data/oracle**

بالمناسبة برضو كل Volume Group بيكون ليه مجلد backup موجود فى

## **/etc/lvm/backup**

وده فيه كل التغييرات اللى انت بتعملها على ال Volume Group سواء **ضفت**

Logical Volume او **حذفت** حتى Logical Volume او **زودت** مساحة واحد

فيهم وهكذا ، والملف ده فعلا عجيب شوية علشان بيحبلك كل حاجة اتعملت

امتى واتعملت ازاي وبعد ايه بالظبط

```
description = "Created *after* executing 'lvcreate --  
size 22G --name oracle data'"
```

طبعا لو عايز تحول ال Physical Volume وترجعه Physical Partition بيكون  
عن طريق الامر ده

```
pvremove /dev/sdb1 /dev/sdc
```

وممكن تمسح كذا هارد من ال Physical Volume عن طريق الامر

```
pvremove /dev/sd[bcd]
```

**السؤال بقى هو انهى الاحسن انى اعمل implement ل RAID يكون  
over ل LVM ولا LVM يكون over ل RAID؟؟؟**

انا اصلا اصلا بستخدم ال RAID علشان ال Redundancy وبستخدم فوقه ال  
LVM علشان ال Flexibility علشان اقدر اعمل Extend او اعمل Shrink  
للبارتيشنز

يبقى انت كده لما بتعمل RAID فوق ال LVM كده انت هتفقد ال Redundancy  
انما لما تستخدم ال LVM فوق ال RAID ، فانت كده بتستفيد من ال RAID على  
ال Low Level وتستفيد من ال LVM على ال High Level

وخلى بالك برضو فى موضوع ال RAID انت لما بتعمله stop لل Array انت  
مش بتمسح الداتا ، لان انت ممكن تعمله scan من تانى وترجع الداتا ،  
وبالتالى كان بيسيب ال Super Block اللى هى بتكون موجودة فى بداية  
البارتيشن ويكون عليها ال metadata بتاعت البارتیشن ده او الهارد ده ،  
يعنى مثلا بيتخزن فى ال super block دى هو البارتیشن ده هل هو Member  
فى Physical Group ولا لأ ، وهل البارتیشن ده كان موجود فى RAID Array  
وهكذا بقى مما يتعلق بال meta data

علشان كده انت لازم تحذف او تعمل zeroing لل super block دى ، قبل ما تعمل RAID والا هتظرك رسالة ال warning دى

**mdadm: Note: this array has metadata at the start and may not be suitable as a boot device. If you plan to store '/boot' on this device please ensure that your boot-loader understands md/v1.x metadata, or use --metadata=0.90**

**Continue creating array?**

هنا بقى بعد ما عملنا ال RAID ، هنبداً نعمل ال Physical Volume ، لكن هيكون التعامل مع ال RAID هنا ، عن طريق الامر

**pvcreate /dev/md[01]**

وبعدين هنعمل Volume Group ونسميها مثلاً data عن طريق الامر

**vgcreate data /dev/md[01]**

بعدين هنعمل ال Logical Volume من ال Volume Group عن طريق الامر

**lvcreate --size=30G --name oracle data**

وهنا بقى علشان تكون more smart وانت بتقسم البارتيشنز من الاول خالص وقبل ما تعمل ال RAID ، حاول تخلى مساحة البارتيشنز مضبوطة

يعنى مثلا لو هتعمل بارتيشن مساحته 5 جيجا ، يبقى تضرب ال 5 فى 1024  
علشان تطلعلك المساحة بالضبط ، وحتى لما تيجى تعمل ال Logical Volume  
لازم تضرب الرقم فى 1024 وبعدين تبقى تكتبه ، علشان تحافظ على  
المساحات اللى عندك وتستغلها احسن استغلال اه بالمناسبة ال **lvs** هى  
اختصار ل Logical Volume Short حرف ال S هى اختصار short view

وخلى بالك انك طالما عملت LVM فوق RAID ، يبقى انت ملكش دعوة بال  
RAID خالص ، اخر تعاملك معاه يكون على المستوى ال Physical يعنى مثلا  
تشيل هارد ديسك وهكذا

بعدين بقى هنعمل File System لل Logical Volume اللى اسمه oracle ده  
عن طريق

**mkfs.ext4 /dev/data/oracle**

وخلى بالك ان ال Meta Data بتاعت ال LVM بتتكتب لما تنفذ الامر pvcreate  
علشان كده انت عندك **pvremove** لان دى بتضمنلك انها تمسح ال Meta Data  
من ال Super Block  
طيب انت ممكن تعمل simulate ان عندك ديسك بقى failed عن طريق  
الامر

**mdadm /dev/md0 -f /dev/sdb**



**وعموما اكثر حاجة مزعجة** لای System Admin هو موضوع ال Shrink او انه  
يعمل resize ل File System والعملية دي ليها احتمالين يا اما تفشل او يا اما  
تنجح ، ويا سلام بقى لو انت وانت بتعمل resize حصل Crash لای سبب  
بقى ، وال crash بقى ممكن يكون **Power Failure** او مشكلة فى ال OS ذات  
نفسه او ان ال Disk نفسه ي Fail فى حته معينة ويا سلام بقى لو مش واخذ  
Backup ، يبقى البس يا معلم ، يبقى انت تحاول تعمل resize بشكل امن

تعالى بقى نعمل **extend** لل Logical Volume اللى انا عملناه عن طريق الامر  
**lvextend --size +5G /dev/data/oracle**

بس خلى بالك علشان فى مطب هنا ، لما بتيجى تعمل extend بيتم ، لكن لو  
جيت نفذت امر زى

**df -h**

هتلاقى المساحة مزادتش ، طب ليه ، لان هنا ال File System ذات نفسه  
محسش بحاجة ، انت لما عدلت عدلت فى ال LVM نفسه ، كده  
ال File System محسش باى تغيير حصل لل **Inode Table** ، هتقولى ما انا  
ممكن اعمل umount لل

**/oracle**

برضو مش هتلاقى ال inode table اتعمله update ، يبقى انت كده لازم  
تعمل update لل inode table مش تعمل reread لل inode Table

يبقى انت كده هتقول لل inode table شوف ال blocks الجديدة اللي  
اتضافت ، وضفلها inodes تقابلها ، يعنى مثلا لو انا ضفت 5 جيجا ، يبقى ال 5  
جيجا دول بيشاوروا على Blocks معينة ، طيب هتعمل update لل inode table  
عن طريق الامر

**resize2fs /dev/data/oracle**

خدت بالك انك لما عملت resize لل File System كان معمول ليه mount  
عادى ، يبقى اذا نستنج من كده ان لو عندك service زى ال **Apache** او ال  
**oracle DB** وعايز تعمل extend لل File System ده مش لازم توقف ال  
Services دى كلها ، لا لا عادى اعمل Extend وكل حاجة شغالة ، لان دى اهم  
واجمل حاجة فى ال LVM وهى انك ممكن تنفذ عملية ال Shrink وال Extend  
من غير ما توقف اى Service عندك

كده انت قبل ما تخش فى ال **cluster** لازم تكون فاهم RAID كويس فشخ ،  
وتكون فاهم LVM كويس فشخ برضو ، وتكون فاهم XFS برضو كويس اوى  
اوى ، بعدها هتلاقى ال Cluster سهلة ان شاء الله

نبدأ بقى فى عملية ال **Shrink** لل File System ، وخلينا نتفق على شوية قواعد كده نمشى عليها

مبدئيا كده احنا هنعمل عملية عكسية ، حاجة كده عكس عملية ال resize وطالما هنعمل عملية عكسية يبقى اول حاجة هنبص عليها هو ال file system مش ال LVM

يعنى بدل ما كنا بنعدل فى ال LVM وبعدين نروح لل file system نقوله لف كده لفة وشوف ال blocks اللي اتضافت ، لا هنا بقى هنروح لل file system الاول وهنقوله يا عم ال file system احنا هنقلل عدد ال blocks بتاعتك ، وبعد ما تقللها فعليا ، تروح تقللها بقى عن طريق ال LVM من خلال الامر **lvreduce**

ملحوظة مهمة جدا ليك وهى انك لازم قبل ما تعمل Shrink او Resize لاي File System ، لازم ولايد انك تعمل check لل File System ده ، وخلي بالك من النقطة دى كويس

والعملية دى اصلا اجبارية لان كل الهدف من انك تعمل check لل file system وهى انك تتأكد ان ال inode table بتاع البارتيشن او ال file system ده تمام ومفيهوش مشكلة

يبقى كده لازم تتأكد ان ال file system بتاعك يكون stable والا انت هتلبس

كده انت وانت بتعمل shrink او reduce لل file system لازم تعمل umount ليه الاول ، تانى حاجة اتأكد من المساحة بتاعتك قبل ما تعمل reduce

بعد كده بقى الاجراءات بتاعت ال Shrink او ال reducing ، اول حاجة هنعمل  
umount

**umount /dev/data/oracle**

وبعدين هنعمل **e2fsck** عن طريق الامر ده

**e2fsck -f /dev/data/oracle**

وبعدين هنستخدم ال **resize2fs** علشان نقوله اجهز للحجم الجديد اللى  
هنديهولك يا برنس وده عن طريق الامر

**resize2fs /dev/data/oracle 30G**

واخيرا بقى هنعمل عملية ال shrink بتاعتنا عن طريق الامر

**lvreduce -L -5G /dev/data/oracle**

بس كده

## بعد كده بقى عندنا موضوع ال LVM Mirroring وال LVM Stripping

### وال LVM Linear

وفاكر لما قولنا ان ال lvm لما بيقسم الديسك لاجزاء والاجزاء دى اسمها extent او strips ، وبيكون ال default size بتاعها هو 4 ميغا

طيب هو اصلا ال LVM بيكتب ازاي على الديسكات ؟ مثلا لو عندك 3 هاردات ف اهو هيبداً يكتب على الهارد الاول لحد ما يتملى وهكذا لحد ما يملاهم كلهما طبعا ده فى حد ذاته عبارة عن خازوق كبير ، لان عندك مثلا لو ديستك باظ ، يبقى الداتا بتاعتك بالسلام على كده يا برنس ، يبقى هل الاحسن ان يكتب على كل strip بتاع هارد واحد بس ، ولا انه يكتب على strip بتاعت هارد ديستك ، وبعدين يروح يكتب على strip تانية بتاعت هارد ديستك تانى ، يعنى توزع ال load على الديسكات اللى عندك كلهم ودى طبعا افضل حل علشان متخليش الابرّة بتاعت هارد واحد بس تتعب

وعارف لما تكتب على مساحة من الهارد مثلا وترجع تمسح ، تكتب وتمسح ، كده الابرّة بتاعت الهارد بتتمركز منطقة واحدة وبيكون اسم المنطقة دى hotspot ، يعنى مكان فيه زحمة وبالتالي عمر الهارد بيقل بشكل كبير جدا

يبقى كده الطريقة اللى ال LVM بيكتب بيها على هارد كله لحد مايخلصه وبعدين يروح للهارد التانى ، الطريقة دى اسمها **Linear**

انما الطريقة الثانية اللى ال LVM بيروح يكتب فيها على كل strip من كل هارد اسمها **stripping** ، من اسمها كده يعنى يقطع الداتا على الهاردات اللى موجودة وبالتالي بيعمل تخفيف لل load على الهاردات ، وبالتالي بالنسبة للطريقة الثانية ، كل ما عدد الديسكات بيزيد كل ما ال performance بتاع السيرفر بيزيد وطبعاً العمر الافتراضى للديسك هيزيد من ناحية الكتابة عليه

طيب لو هتمسح Logical Volume بيكون عن طريق الامر

**lvremove /dev/data/oracle**

وبعدين هتمسح ال Volume Group عن طريق الامر

**vgremove data**

وبعدين هتمسح ال Physical Volume عن طريق الامر

**pvremove /dev/md0**

**pvremove /dev/md1**

كمان بقى ممكن توقف ال RAID لو حابب يعنى

**mdadm --stop /dev/md0**

**mdadm --stop /dev/md1**

وبعدین تمسح ال super block بتاع الهاردات عن طريق الامر

**mdadm --zero-superblock /dev/sd[bcde]**

طيب انت لو عايز ال LVM يبدأ يوزع ال data على الهاردات ، هتستخدم الامر

ده

**lvcreate --size=40G --name=data -i 4 data**

هنا ال **i** هى اختصار لعدد الهاردات الموجودة عندى فى المثال ده

وطبعاً هتضطر تشوف ال mapping بتاع ال LVM ومعناها خريطة او شكل

الهاردات فى ال LVM عامل ازاي ، عن طريق الامر

**lvdisplay -m /dev/data/oracle**

وده الناتج بتاع ال command ده

**--- Segments ---**

**Logical extents 0 to 10239:**

**Type            striped**

**Stripes        4**

**Stripe size        64.00 KiB**

### **Stripe 0:**

**Physical volume**      **/dev/sdb**

**Physical extents**      **0 to 2559**

### **Stripe 1:**

**Physical volume**      **/dev/sdc**

**Physical extents**      **0 to 2559**

### **Stripe 2:**

**Physical volume**      **/dev/sdd**

**Physical extents**      **0 to 2559**

### **Stripe 3:**

**Physical volume**      **/dev/sde**

**Physical extents**      **0 to 2559**

السطر ده

### **Logical extents 0 to 10239:**

هو عدد ال stripes الكلية الموجودة ، وهنا بقى عدد ال stripes بالنسبة لكل

هارد هو

**Physical extents**      **0 to 2559**



طبعا **stripe** دى بالنسبة لل **Physical Volume** ، بيقابلها بقى كلمة **extent** بالنسبة لل **Physical Disk** او ال **Physical Partition**

دلوقتى لو عايز تعرف عندك كام dependency بالنسبة لل Logical Volume  
بيكون عن طريق الامر

**dmsetup deps /dev/data/oracle**

يبقى خليك فاكر ان ال LVM ب **By Default** بيشتغل بال Linear ، انما بقى لو  
حببت تغير ل **stripping** ، فانت هتقوله وانت بتعمل ال Logical Volume ذات  
نفسه

الموضوع اللى بعد كده وهو موضوع ال **LVM SnapShot** ، وده موضوع مهم  
ليك ك system admin لانه هيساعدك فى احوال كتيرة اوى ، واهم حاجة منهم  
هو موضوع ال updates

بالمناسبة لما تيجى تعمل snapshot لازم يكون عندك داتا اصلا موجودة ،  
ويفضل انك لما تعمل او تاخذ snapshot من الداتا ، تروح تحطها على مكان  
يكون مساحته اكبر من مساحة ال SnapShot دى

يعنى ال SnapShot اللى انت خدتها لو مساحتها اقل ، بمعنى ان انت خدتها على مكان تانى اقل او بيساوى مساحة الداتا الفعلية ، فابقية الداتا الفعلية هيتعملها discard لما تيجى تكتبها ، وابقى بص على الفيديو رقم 56 من اول الدقيقة 40

بص بكل بساطة انت عندك هارد حجمه 500 جيجا والهارد ده مكتوب عليه داتا حجمها مثلا 50 جيجا ، فانت لما تيجى تاخذ SnapShot من الداتا دى وتحطها على هارد تانى ، لازم تكون مساحة الهارد التانى كافية انها تشيل الداتا بتاعت الهارد الاول وتشيل زيادة كمان علشان لو انت زودت الداتا بتاعت الهارد الاول

يبقى انت من المفضل جدا ان حجم ال snapshot يكون قد حجم ال logical volume وليس قد حجم الداتا المكتوبة على ال Logical Volume ده

تعالى بقى نعمل SnapShot ، وده عبارة عن اننا بنعمل Logical Partition ، بيشاور على داتا موجودة فى

**/dev/data/oracle**

ده هيثم عن طريق الامر

**lvcreate --size=2G --name=oracle-snap --snapshot /dev/data/oracle**

هنا بقى بما اننا خلىنا مساحة ال snapshot تكون 2 جيجا ، يبقى مينفش نكتب  
اكثر من 2 جيجا فى ال snapshot ، عارف برضو ده معناه اي ، انك هتكتب  
بس 2 جيجا وضيغهم على المساحة القديمة ، ابقى ارجع برضو للفيديو 56 من  
اول الدقيقة 58 تقريبا ، من اول ما بدأ يعيد شرح ال snapshot

وصلنا بقى لحالة انك عايز تعمل **revert** لل snapshot دى ، اولاً هتعمل  
umount لل logical volume الاساسى ، وبعدين بقى هتعمل **merge** لل  
snapshot يعنى هترجع نسختها دى للمكان اللى اتاخذت منه ، وده عن طريق  
الامر

## **lvconvert --merge /dev/data/oracle-snap**

وخلى بالك ال snapshot بيستخدم حاجة اسمها **COW** ،  
يعنى **copy on write** ، كده عملية ال snapshot هيسيب النسخة الاصلية مش  
هيكرب عليها ، هيقف الكتابة عليها ، تقدر تقول كده انه هيسيبها read only

**بص** فكك من الكلام اللى فوق وركز هنا بقى ، ملخص الحوار كله ان ال  
snapshot مش حجم الداتا اللى انت هتاخذ منها صورة ، ال snapshot هى انه  
بيوقف الكتابة على الداتا اللى كانت مكتوبة قبل كده ، بس ال snapshot  
بتحددلك حجم الكتابة اللى هتكتبها قد اي ، يبقى اذا ال snapshot هو حجم  
الداتا اللى هتكتب على الداتا اللى كانت مكتوبة قبل كده

قصدي يعنى زائد النسخة الاصلية ، يعنى مثلا لو حجم الداتا الاصلية 5 جيجا  
وال snapshot حجمها 2 جيجا ، يبقى انت بعد ما تخلص كتابة وتملى ال 2  
جيجا دول ، هيكون مجموع الداتا عندك 7 جيجا

مثال تانى لو عندك داتا حجمها 3 جيجا وعملت 6 snap ، يبقى الحجم الكلى  
للداتا هيكون 9 جيجا ، وهنا فى Trick صغيرة ، وهى ان الداتا الجديدة تشمل  
برضو الداتا اللى هتتمسح

**يعنى ابي الداتا الجديدة هتشملى الداتا اللى هتتمسح؟؟؟** بكل بساطة لو  
عندك داتا حجمها 3 جيجا وجيت عملت snapshot حجمها 5 جيجا ، وجيت انت  
مسحت 1 جيجا من ال 3 جيجا دى فال snapshot هتاخد الواحد جيجا اللى  
اتسمح وتضيفه عندها ويبقى كده مساحة ال snapshot المتاحة للكتابة هى 4  
جيجا بس ، طيب افرض بقى ان انت كتبت على ال snapshot داتا اكر من  
حجم ال snapshot ، يبقى اذا طيعى ان الداتا الزيادة يحصلها lose زى كوباية  
المياه اللى بتملها على الاخر

**يعنى الخلاصة بقى ان اى عملية اضافة او حذف للداتا من على الداتا  
الاصلية ، النسبة المئوية بتاع ال snapshot هتزيد ، وبرضو هيفضل محتفظ  
بالداتا الاصلية ، لان دى فكرة ال snap بصفة عامة**

كده السؤال بتاع هل المفروض حجم ال snap يكون كبير ، ده سؤال مطاط اوى ، لان انت هنا مش هتعرف حجم الداتا اللى انت هتكتبها قد اي ، وبالتالي لازم تكون عارف هتكتب داتا قد اي

يبقى انت هتستخدم ال snapshot فى حالات معينة ، مش وخلص ، هتستخدمها مثلا فى حالة ال updates للتوزيعة كلها ، او مثلا فى حالة ال Backup وهكذا

طيب افرض بقى انك عايز تزود مساحة ال SnapShot دى ، فانت عادى ممكن تعملها Extend بشكل اوتوماتيك او بشكل Manually

فى حالة الطريقة ال Manually عن طريق الامر

```
lvextend --size=5G /dev/data/oracle-snap
```

وبالنسبة بقى للطريقة ال automatic وهى انك هتعدل فى ملف ال configuration بتاع ال LVM عن طريق

```
vim /etc/lvm/lvm.conf
```

وتعدل السطر ده

```
snapshot_autoextend_threshold = 100
```

السطر ده معناه انه هبدأ يعمل extend اول ما مساحة ال SnapShot توصل ل 100% ، طبعا انت مينفعش تستناه لحد ما المساحة دى توصل ل 100% ممكن تغيرها وتخليها 80 مثلا

**snapshot\_autoextend\_threshold = 80**

والسطر ده بقى معناه هو هيزود مساحة ال snapshot بنسبة كام فى الميه

**snapshot\_autoextend\_percent = 20**

بنسبة 20 دى نسبة حلوة مش وحشة

وطبعا هو by default مش بيعمل extend ، ال 100 معناها انه مش هيعمل  
اي extend

كده انتهى موضوع ال LVM وياريت برضو تبص على المصادر الللى قولنا  
عليها فوق

# 6- Swap

اولا ال **swap** معناها التبدیل ، وفكرتها انهم عملوها علشان تكون ك **virtual memory** یعنی تكمل الرام، بمعنى ان انت عندك applications كثيرة فى الرام ، هل بقى طول ما انت شغال هل كل ال applications بتكون active و running فى نفس الوقت ، اكيد لا طبعا ومستحيل تكون كل ال applications اکتيف و running فى نفس الوقت ، قالك خلاص طالما انت عندك ابلیکیشنز مش شغالة او بمعنى ادق suspended او واقفة بشكل مؤقت ، او مستتية لاي I/O ( یعنی مستتية اى Input او Output ) طالما انت مش محتاجها دلوقتى ، بيقى تشيلها وتحطها فى ال swap

قالك بقى افرض ان اليوزر حب يرجع يتعامل مع ال apps دى تانى او انه يبدأ ي **interact** معاها تانى ، يبقى خلاص شيلها من ال swap ورجعها لل RAM تانى ، وده سبب تسميتها بال swap ، لانه شغال بيشيلها من الرام ويحطها على الهارد ديسك ، ولما بيحتاجها بيشيلها من على الديسك ويرجعها للرام ، عملية تبديل یعنی ، طب عملية النقل دى مين بيقوم بيها؟؟  
اولا انت مش بتقوم بيها خالص ، ال os او الكرنال بشكل ادق هو اللى بيقوم بيها

انت كل اللي عليك انك هتبلغ الكرنال بان فيه swap space موجودة على الهارد ابقى استخدمها لما مساحة الرام تقل ، طبعا لما عملية ال swap تزيد ، الجهاز هيبقى ابطأ ، لان طبعا سرعة الرامات اعلى بكثير من الهارد ، وبالتالي ال Performance هيكون ابطأ ، طبعا متحاولش تقارن سرعة الرامات بسرعة ال **Mechanical Hard Disk** ، وال **SSD** مش بديل دايمًا للرامات وبرضو الرامات اسرع بكثير من ال SSD ، وانت اصلاً مش ممكن تحتاج لل swap اصلاً لان طالما الرامات عالية عندك يبقى اشطاً

طبعا بالنسبة لمساحة ال swap اللي المفروض تعملها ، مفيش حاجة standard تمشى عليها ، الموضوع بيعتمد على كم ال apps اللي انت هتشغلها يبقى لازم يكون عندك planning محترم فيما يخص عملية المساحة اللي هتجزها من الهارد لل Swap ، وكلمة Planning معناها انك تشوف ال application اللي هيشغل عندك قد اي وال Memory Requirements قد اي ، يعنى تشوف ال Minimum اي وال Recommended اي وال Best Use قد اي ، طبعا متنساش بقية ال apps

وخليك فاكرا انه كل ما ال swap ما كانت قليلة ، كل ما كان ال performance بتاعك افضل ، طيب خلى بالك برضو لما تيجى تنفذ الامر

## free -m

متبصليش على ال free بس وتسكت



خد فى اعتبارك ال cash برضو ، ليه بقى ، لان انت ممكن يكون ال free قليلة اوى ، بس معظم ال apps موجودة فى ال cache بتاع ال Memory ، طب هو بيعمل ال cache ليه ؟

لانه مش عايز يستخدم ال swap ، فانت بقى تبدأ تقلق لو مساحة الكاش قليلة اوى ومساحة ال free برضو قليلة اوى ، انت تبص على ال used لو لقيتها مليانة ابدأ اقلق فعلا

خلى بالك ان مجلد ال **proc** فيه كل حاجة الكرنال يعرفها عنك سواء بقى services او physical component او كرنال ذات نفسها ، بص اى حاجة خاصة بالجهاز موجودة فى ال proc

طيب لو عايز تزود مساحة ال swap ، فانت عند طريقين ، يا اما انك تعمل بارتيشن وتخليه ك swap وتضيفه على المساحة الموجودة بتاعت ال swap الثانية ، او الطريقة الثانية عن طريق file

ومتنساش بعد ما تعمل ال swap ، انك تحطها فى ال fstab علشان تثبت الكلام ده ، وطبعا علشان تتأكد انه كل مرة هيتعمله mount ، ابقى نفذ الامر ده

**mount -a**

وطالما مطلعش اى error يبقى الدنيا عندك تمام ، طيب لو عايز تعمل **deactivate** لل swap ، يبقى اول حاجة تتأكد انك تشيلها من ال **fstab**

معلومة على الماشى ، كلمة format معناها **zero fill** ، يعنى تملأ ال **blocks** باصفار

طيب الطريقة الثانية فى عملية ال **swap** وهى انك تعمل ملف باستخدام الامر **dd** ويكون الملف ده عبارة عن blocks كلها مليانة اصفار ، عن طريق الامر ده

**dd if=/dev/zero of=/root/myswap bs=1M count=2048**

بعد ما عملنا الملف ، هنعمله بقى file system ، ازاي يا جدع ؟ ، ما احنا قولنا خلاص ان الملف ده عبارة عن اصفار زيه زى ال file system اللى معمول ليه format او ممكن تقول عليه raw file system ، يعنى بارتيشن لسه مش عليه اى حاجة وبعدين تكتب بقى

**mkswap /root/myswap**

وبعدين نعمل activate لل swap دى عن طريق

**swapon /dev/myswap**

طيب لو الرسالة دي ظهرتلك

**swapon: /root/myswap: insecure permissions 0644, 0600 suggested**

معناها انه بيحذرك انه بيقولك ال others ليهم **read access** على ال file ده عن طريق طبعا رقم 0644 ، وبيقترح عليك انك تمنع ال others من انهم يكون ليهم اى access على الملف ده ، يبقى انت هترجع توقف ال swap من تانى

**swapoff /root/myswap**

وبعدين تغير ال Permissions بتاعت الملف ده

**chmod 600 /root/myswap**

وبعدها بقى ترجع تعمل activate لل swap من تانى

وطبعا الكلام لاننا بمنع ان ال users العاديين يكون ليهم اى direct access لل Memory بتاعت الجهاز ، ومتنساش تثبت الكلام ده كله فى ال fstab وبعدها

**mount -a**

وبكده انت ممكن تعمل ال swap على File او Partition

# 7-Quota Management

كلمة Quota معناها فى اللغة العربية الحصة او النصيب ، طيب وهنا بقى معناها انى ادى لكل مستخدم نصيب معين من ال Storage ، وبالمناسبة ال Quota مهمة بس بال Storage يعنى المساحة اللى هياخذها ال users ، لكن مش مهمة بال Performanace بتاعت ال Storage ذات نفسها ، بمعنى انها مهمة بانها مثلا تحدد 10 جيجا لمستخدم معين ، لكن مش مهمة انها تحدد هو هيكرب على الديسك بسرعة كام بقى ، يبقى نطلع بقى بخلاصة القول هنا وهى ان ال quota مهمة بالمساحة انما ال **Cgroup** هى اللى بتحدد ال user ده هيكرب بسرعة كام بالظبط

طيب هنا بقى ال Quota علشان تطبقها ، لازم وانت بتعمل mount لل file system تقوله انك هتستخدم ال quota ، يبقى كده ال File System وهو بيتعمله mount يكون بي support ال quota ، الحاجة الثانية وهى انك لازم تعمل enable لل quota ، طيب هنا بقى فى سؤال المفروض تسأله ، هى ال Quota دى هتعرف هو ال user ده مستخدم 1 جيجا ولا 2 جيجا ولا اى بالظبط قالك بقى ان ال quota وهى شغالة بيكون عندها data base بيكون فيها معلومات ال users وهما مستخدمين مساحة قد اى بالظبط

طبيب افرض ان انت عندك File System كان Created ، بس ال Quota مكانتش enabled عليه من الاساس ، معنى كده ان ال Quota DataBase مش موجودة ، وده معناه برضو انك لما تيجى تعمل enable لل Quota دى ، لازم تروح تعمل create لل data base دى

فى حالة تانية برضو ، افرض بقى ان ال Quota دى كانت شغالة على ال File System ده ، يعنى كانت **ON** ، جيب انت خليتها **OFF** ، راح جه users وكتبوا ملفات على ال File System ده ، هل ال Quota دى بقت up to date دلوقتى ، لا طبعا وساعتها بقى هتضطر تعمل recreate لل Quota

وبكده انت عندك حالتين هتضطر تعمل create لل Quota فيهم ، اولهم وهى ان ال file system يكون مش بي support ال Quota ، وجيت انت عملتها enable ، والحالة التانية انها تكون شغالة وتيجى انت تعملها OFF وياجى مستخدمين تانيين يكتبوا ملفات على ال file system ده ، يبقى كده انت المفروض هتعمل recreate لل Quota دى

**Isosf** الامر بيعرضك الملفات المفتوحة من مكان معين ، مثلا

**Isosf /usr/**

بمعنى اخر اى اللى مستخدم من المكان الفلانى ده ومسبلك انك مش عارف تعمله unmount

طيب هل ال Quota بتكون **enabled** وانت بتعمل **mount** لل **file system** ؟؟

**الجواب لا طبعا**

نيجى بقى لموضوع تفعيل ال Quota وانت بتعمل **mount** ، عن طريق الامر

ده

**mount -o usrquota,grpquota /dev/data/mssql /google**

او عن طريق ده لو نفع يعنى

**udisksctl mount --block-device /dev/data/mssql -o usrquota,grpquota**

طب احنا ليه عملنا **usrquota** و **grpquota** ، علشان ببساطة ال quota ممكن تطبق على user لوحده ، او على مجموعة users موجودين فى جروب معين ، بمعنى ان انت عندك directory مساحتها 50 جيجا ، يبقى كل اللى فى الجروب المعين ده هيكون ليه انه يكتب فى ال directory دى ، طب ولو كلهم بيكتبوا ، يبقى كلهم ليهم 50 جيجا بس ، يعنى هيوصلوا لنفس ال limit مع بعض

فانت كده وانت بتطبق ال **quota** ممكن تقوله انك هت implement  
ال quota على ال user بس ، او على ال group بس ، او على ال user  
والجروب مع بعض

وبالمناسبة انت لو عملت reboot للجهاز ، يبقى كده ال mount اللى انت  
عملته لل file system ده راح ، يبقى لازم تثبت الكلام ده فى ال fstab بالشكل  
ده

```
/dev/data/mssql          /google          ext4 defaults,usrquota,grpquota 0 0
```

وبعدين متنساش تعمل

**mount -a**

علشان تتأكد ان كله سليم اثناء عملية ال boot ، وبعدين

**df -h**

هتلاقيه اتعمله mount اوتوماتيك

تعالی بقى نقول لل kernel انها تعمل enable لل quota دى ، وده عن طريق الامر

**quotaon /google**

بص لرسالة ال **Warning** دى

**quotaon: cannot find /google/aquota.group on**

**/dev/mapper/data-mssql [/google]**

**quotaon: cannot find /google/aquota.user on /dev/mapper/data-mssql [/google]**

دى معناها ان ال quota مش لاقية ال data base بتاعت ال user ولا ال group ، وده معناها انك لازم تعمل build لل data base الاول بتاعت ال quota عن طريق الامر ده

**quotacheck -cvug /google**

**c ----> create**



**v ----> verbose**

**u ----> user**

**g ----> group**

وبعدین بقى ترجع تعمل

**quotaon /google**

طیب تعالى بقى نعمل implement لل quota ، مبدئيا كده انت ممكن تعمل  
implement لل quota عن طريق ال Size او عن طريق ال Number Of Files ،  
بمعنى انى ممكن اقوله ان ال user اللى اسمه ali ده من حقه انه يكتب 5  
جيجا بس ، او من حقه انه يكتب 5 ملفات بس ، طب وهو انا محتاج اقوله ان  
ال user ده يكتب عدد معين من الملفات ؟؟

ايوه بالظبط ، طب ليه؟ علشان ببساطة الموضوع متعلق بال inodes ، وارجع وافكر ان انت فى كل file system عندى عدد معين من ال inodes لو خلصت ، يبقى انت كده مساحة البارتيشن بقت بالسلامة على كده بمعنى اي برضو ، بص بكل ببساطة لو انت عندك user وعطتله مساحة 1 جيجا ، وراح هو عمل بتاع مليون ملف فاضى ، كده بيخلصلك عدد ال inodes اللى عندك لان متنساش ان كل inode بيشار على ملف ، وده السبب ان بيكون عندك مساحة فاضية لكن مينفعش انك تكتب داتا ، وده بسبب ان عدد ال inode خلص ، وكده انت خسرت ال storage اللى عندك

النقطة الثانية وانت بتعمل implement لل quota ، بيكون عندك 2 limits ، يا اما يكون عندك **soft limit** او يا اما يكون عندك **hard limit** ، طيب اي بقى الاتنين دول ؟

بص ال **soft limit** هى ال limit اللى لما ال user يوصلها هيبدأ يطلع له warning ، انما بقى ال Hard Limit هى ال limit اللى مستحيل ال user يعديها باى حال من الاحوال

مثال : مثلاً عندك ال account بتاع ال gmail لو ليك مثلاً 10 جيجا بس ، فال warning هبدأ يطلعك لما توصل لاستخدام 9 جيجا وده كده ال soft limit ،  
انما ال 10 جيجا هي ال Hard Limit ، طيب هل ممكن ان انت تتخطى  
ال soft limit ، ايوه هتخطاه لحد ما توصل لل hard limit

طيب تعالى بقى نعمل implement لل quota لل user اللي اسمه ali ، عن  
طريق الامر

**edquota -u ali**

تعالى بص كده للكام سطر ده

**Disk quotas for user rhadmin (uid 1000):**

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/mapper/data-mssql	0	0	0	0	0	0

طبعا هتسأل هو ليه مش جايب بقية البارتيشنز اللي معمول ليهم mount ،  
الجواب ببساطة لان ال quota مش متفعلة عليهم

طيب ال blocks هي عدد ال blocks الفعلية اللي بيستخدمها ال user ده ، الرقم 0 معناه انه مش بيستخدم اى حاجة لحد دلوقتى ، طيب السؤال برضو هل انت لما تعمل implement لل quota هتعدل فى ال blocks دى ، طبعا لا لان ال blocks دى بتمثل المساحة اللي الراجل مستخدمها حاليا ، كده انت المفروض تعدل فى ال soft limit وال hard limit بتاعته

يعنى انت ممكن تحدد ال soft limit برقم مثلا 100000 وده معناه 100 ميغا او معناه 100 الف كيلو بايت اللي هي برضو 100 ميغا ، وخلي برضو على سبيل المثال ال hard limit خليه 200000 ، يعنى كده احنا هنعمل implement لل quota بال size

طيب بعد ما عملت implement لل quota لل user اللي اسمه ali ، وجيت تنفذ الامر

**repquota -a**

هتلاقى الناتج بالشكل ده

### \*\*\* Report for user quotas on device /dev/mapper/data-mssql

Block grace time: 7days; Inode grace time: 7days

User	Block limits				File limits			
	used	soft	hard	grace	used	soft	hard	grace
root	--	20	0	0	2	0	0	

طيب ليه ال user اللى اسمه ali ، مش ظاهر ، هقولك ببساطة ، لان ال directory اللى اسمه google او بشكل ادق ال lv اللى اسمه

### /dev/data/mssql

اللى اتعملها mount فى المجلد اللى اسمه /google ، كان ال others مش معاهم اى write permissions عليه ، تعالى بقى نفذ الامر ده علشان تعرف اذا كان ال others ليهم Permissions ولا لا

### ls -ld /google/

**drwxr-xr-x. 3 root root 4096 Jun 13 22:33 /google/**

كده انت هتضطر تدي لل others ال write permissions ، طبعا زي ما انت عارف عن طريق

**chmod o+w /google/**

لوجيت تنفذ الامر repquota ده تاني هيظهرلك ال report بتاع ال quota الخاص بال user اللى اسمه ali ، يبقى كده هو هيظهر لما ال user يبدأ يستخدم اى ملفات على ال File System

(خلى بالك من الملحوظة دي ، لو انت وانت بتسطب التوزيعة جيت تعمل **user** اسمه مثلا **mostafa** ، وجيت علمت على

**make this user as administrator** ، كده ال **user** ده بقى زيه زي ال **root** يعنى لما تيجى تعمله **quota** هتلاقى اللى بيزيد هو ال **quota** بتاعت ال **root** مش بتاعته ، فخلى بالك بقى انما انت المفروض تعمل ال **quota** لل **users** العاديين اللى ملهمش صلاحيات ال **administrations** )

تعالى بقى بص بقى كده على السطر ده

**ali +- 122880 100000 200000 6days 2 0 0**

اولا علامة ال + دي بتشيرالى ال blocks ولما بتظهر بيكون معناها ان ال user تخطى عدد ال block limit

انما بقى علامة ال - فدى معناها انه لسه متخطاش عدد ال file limit اللى مسموح بيها ، وهنا هى ظاهرة - لانى محددتلوش اى limit لعدد الملفات يبقى كده بشكل اوضح ، ال + معناها انه تخطى ال soft limit بتاعه

تعالى بقى نعمل limit للملفات اللى ال user من حقه ان ينشأها عن طريق برضو الامر

### **edquota -u timon**

وهتعدل اخر soft واخر hard بعد كلمة inode وممكن طبعا تستخدم watch مع **repquota**

طبعا لو عايز تعمل implement لل quota على **group** ، بيكون عن طريق

### **edquota -g dcadmins**

عادى زيها زى اللى فوق ، شوف اسم الجروب وعدله ، وبرضو

### **repquota -g /google**

ودلوقتى كمان ممكن تعمل quota ل user معين داخل جروب معين ، بس لازم تعمل quota للجروب ، وبعدين ترجع تعمل quota لل user ده

طبعا لو عايز تجيب ال quota بتاعت user معين ، ممكن تستخدم الامر ده

**quota -u ali**

طب اي بقى حكاية ال **grace period** دي ، بص باختصار شديد هى عبارة عن الفترة اللى هيبدأ يتم التعامل مع ال user اللى متحدد ليه quota كانه وصل لل hard limit من عندها

مثلا انت وصلت لل soft limit ، ومثلا ال grace period ، اللى هى فترة السماح هنا بقى هو يبدأ يعاملك على اساس انك وصلت لل soft limit بعد 6 ايام مثلا وده فى حالة لو انت مبدأتش تمسح من ملفاتك

طيب افرض انت عايز تعدل ال **grace period** ، بسيطة خالص ، اكتب

**edquota -t grace**

، وفى الملف ده بقى انت يا اما تعدل ال grace period لل Size بتاعت الملفات اللى هى ال **Blocks** ، او يا اما هتعدل لل **file numbers** ، وطبعا التعديل هيكون يا اما بالايام او الساعات او الدقائق



بس خلى بالك ان انت كده محدثش اى user ، فانت فى موضوع ال grace  
period لازم تحدد انهى user بالظبط ، وده بيتم عن طريق الامر

## **edquota -T ali**

واخيرا بقى لو عايز توقف ال quota ، فالموضوع سهل جدا ، نفذ بس الامر

ده

## **quotaoff /google**

وبكده اى user يرجع يقدر انه يكتب تانى

طيب فى سؤال حلو تانى ، دلوقتى لو انا وقفت ال quota وجه ال user كتب  
ملفات ، وجيت انت شغلت ال quota من تانى ، هل ال quota هتقدر تقرأ  
اللى اتكتب جديد ؟

اكيد لا طبعا ، لان ببساطة ال Data Base بتاعت ال Quota متعملهاش اى

Update

**ملحوظة** مهمة قبل ما تعمل update لل Quota اتأكد ان ال Quota مقفولة

الاول

## **quotaoff /google**

وبعدين

## **quotacheck -vugu**

انما لو عملت update لل data base بتاعت ال quota هيطلع يدك بالقلم  
على وشك ويقولك انك لو عملت ده ممكن ال data base بتاعت ال Quota  
دى تضرب  
عامل بالضبط زى اللى بغير الكوتش بتاع العربية وهى ماشية ، مش منطقى  
خالص 😊😊😊

بص بقى الحقة دى ، انت عمرك مش هتشوف ان ال **used** اكبر من  
ال **hard limit** ، الا فى حالة واحدة وهى ان ال **quota** تكون شغالة  
وبعدين توقفها وتيجى تكتب على البارتيشن اللى معمول ليه **mount** ،  
وهوب بعدين تروح تعمل **update** لل **data base** بتاعت ال **quota** وبعدين  
تشغل ال **quota** ، ساعتها بس هتلاقى ان ال **used** اكبر من ال **hard limit**

حاجة كده مهمة وهى انك لو جيت عملت mount لاي بارتيشن ، وجيت تنفذ الامر **lsblk** ولقيت ان البارتيشن متعملوش mount برضو ، يبقى تشوف ال logs فى

**cat /var/log/message**

هتلاقى هناك رسالة ال error دى

**mounted filesystem with ordered data mode opts[null]**

دى معناها ان ممكن ال system يكون مثلا حصله crash من نوع ما ، فانت تجرب تعمل **fsck**

او امسك عندك الجملة الحلوة دى

**EXT3 and EXT4 has two main modes to handle journaling:**

**Ordered data mode and Data writeback mode**

**Data writeback mode is generally considered to be much faster and better than ordered data mode**

**To change the default:**

**sudo tune2fs -o journal\_data\_writeback /dev/sdX**

**Also--to force a fsck on boot:**

**sudo touch /forcefsck && sudo reboot**

موقع redhat بيتكلم عن المشكلة دي

**<https://access.redhat.com/discussions/3218241>**

**<https://www.redhat.com/archives/ext3-users/2000-December/msg00047.html>**

لكن كمان ممكن يكون سبب المشكلة دي وهى ان ال file system ده معمول  
ليه mount فى ال fstab ، وبالتالي systemd كانت لما بتبدأ بتبص على ال  
entries اللى موجودة فى ال **fstab** وبما ان ال file system معمول ليه  
mount ، فامش هينفع ان اتنين file system يكون معمول ليهم mount فى  
نفس المكان فى نفس الوقت  
يبقى الحل انك هتخلي systemd تعمل reload لل daemons كلها عن طريق  
الامر

**systemd daemon-reload**

لحد هنا احنا خلصنا ال **Basics** بتاعت ال **Quota**

## 8-Access Control List

تذكر ان ال Primary Group مهمة علشان ال File Owner Ship  
بيقولك بقى علشان تتعامل مع ال Access Control List ، لازم تبص فى الاول  
على ال File System بتاعك بتاع ال root ، عن طريق الامر مثلا blkid ، لو  
لقيته xfs ، يبقى تعرف على طول انه بي Support ال Access Control List  
بشكل Automatic ، طب انت هتسأل هى ال Access Control List مش  
supported اصلا؟؟ لا طبعا هى supported وكل حاجة ، بس لازم يكون  
عندك حاجتين علشان ال Access Control List تشتغل ، اولاً ان الكرنال ذات  
نفسه يكون بي support ال xfs وبنسبة 99 كل الكرنال بيدعم ال Access  
Control List

والحاجة الثانية وهى ان لازم ال File System وهو بيتعمله mount يكون  
بي support ال Access Control List ، ودى تحتها خطين ، اولاً ال XFS بي  
support ال Access Control List يعنى مش محتاج تقوله كده وانت بتعمله  
mount ، انما بقى لو انت شغال File System غير ال XFS زى ال EXT4  
مثلاً ، فانت لازم تقوله وانت بتعمل mount انه يعمل ال enable لل Access Control  
List على ال File System ده

طب ازای اعرف ان الكرنال بتاعى انه Compiled بشكل انه ي Support ال  
Access Control List ؟؟؟

بسيطة ، روح على المكان اللى فيه ال kernel ، وهتلاقى هناك ملف اسمه  
كده

## **config-3.10.0-862.3.3.el7.x86\_64**

الملف ده فيه كل ال Options اللى ال Kernel بتاعك بي Support ، وتقريبا  
فيه بتاع حوالى 6000 سطر ، يعنى كبير فشخ

ممکن تستخدم الامر ده

## **grep -i acl config-3.10.0-862.3.3.el7.x86\_64**

علشان تبحث فى الملف على كلمة acl ، وتعرف اذا كان الكرنال بي support  
ال acl ولا لأ

طيب لو انت عايز تخلق ال File System اللى هو EXT4 او غيره يكون بي  
support ال ACL يبقى تستخدم الامر ده وانت بتعمله mount

## **mount -o acl /dev/sdb1 /work**

طيب افرض اصلا ان ال EXT4 كان معمول ليه mount ، هل لازم تعمله  
umount وبعدين ترجع تعمله mount ؟ اكيد طبعا لأ ، لو كان ال File System  
معمول ليه mount بالفعل فانت ممكن تنفذ الامر ده على طول وهو هرجع  
يعمله mount من تانى

## **mount -o remount,acl /work**

ممكن انت برضو بعد ما عملته mount وخليته ي support ال ACL ، ممكن  
تيجى تعمل check عليه وتتأكد اذا كان فعلا بي support ولا لأ عن طريق  
الامر

## **mount**

وتتفاجا انه مش ظاهرلك انه مش بي support ال acl ، مع انك انت عملته  
mount وقولته اعمله support لل ACL ، فعلشان تتأكد انه فعلا بي support  
ال ACL ممكن تستخدم الامر

## **dumpe2fs /dev/sdb1 | less**

وتبحث عن كلمة acl هتلاقى السطر ده

## **Default mount options: user\_xattr acl**

وده معناه انه فعلا بي support ال ACL



خلی بالك لو انت شغال على **RHEL 7.3** فانت فى الغالب مش هتحتاج تقوله  
ادعم ال ACL ، لان ال Developers بتوع RedHat خلوه ال Default Options  
بتاعت ال ACL تكون موجودة مع ال File System حتى وانت بتعمله mount ،  
حتى مع ال EXT4 وغيرهم

تعالى بقى ننفذ ال ACL ، مثلا لو انت عندك يوزرمعين عمل ملف ، وانت عايز  
يوزر تانى يكون ليه صلاحيات على الملف ده ، طبعا هتولى طب ما نخلى ال  
others يكون ليهم صلاحيات  
طبعا مينفعش انا عايز اليوزر المحدد ده هو اللى يكون ليه صلاحيات بس ، ده  
هيتم عن طريق الامر

**setfacl -m u:ahmed:rw file1**

**setfacl ----> set file access control list**

**-m -----> modify**

**u -----> user**

**rw -----> read وال write يعنى هديله ال**

طيب لو انت عايز تعرف ال Access Control Lists المطبقة على الملف ده ،  
يعنى تعرف ال Permissions ، عن طريق الامر

**getfacl file1**

هتظهرلك النتائج دى

**# file: file1**

**# owner: timon**

**# group: wheel**

**user::rw-**

**user:hassan:rw-**

**group::rw-**

**mask::rw-**

**other::r--**

طيب لو عايز تدى جروب معينة صلاحيات لمجلد باللى فيه ، هيكون عن طريق  
الامر

**setfacl -R -m g:dcadmins:rwx /work/**

**-R -----> recursive**

ولو عايز تتأكد كمان ، اكتب الامر

**getfacl /work**

ملحوظة مهمة جدا وهى لازم اليوزر اللى عمل المجلد ، لازم هو اللى ىدى ال **ACL** لليوزرز التانيين او الجروب ، يعنى من الاخر كده اللى عمل الملف او المجلد هو اللاحق بانه ىدى ال **Permissions** لبقية ال **Users**

اه ملحوظة مهمة ، لو انت مثلا ادبت ل user معين ال Permissions على مجلد كامل باللى فيه ، فكداه ال user ده من حقه انه يعدل ويعمل اللى هو عايزه فى الملفات طالما مسموح ليه بكداه ، طيب افرض انت بقى عملت ملف جديد جو المجلد ده بس بعد ما عطيت ال Permissions لل user ده ، فكداه اليوزر ده مش من حقه انه يعدل اى حاجة فى الملف الجديد

طب افرض انى عايز ال user ده ي **inherit** كل مره ، يعنى لو عملت ملف جديد ، كل مرة اليوزر ده هيوثر ال Permissions بتاعت المجلد اللى بيحوى الملف ده ، بمعنى اصح انا عايز ا set ال Default Permissions

بس خلى بالك انك لما تضيف ال Default Permissions ، فهى هتتضاف للحاجات اللى هتتضاف جديدة للمجلد

طب انت هتضيف ال defaults عن طريق الامر

**setfacl -m d:u:ali:rw file1**

او للجروب

**setfacl -m d:g:dcadmins:rw file1**

او للمجلد

**setfacl -R -m d:u:timon:rw /work**

**setfacl -R -m d:g:dcadmins:rw /work**

طب لو عايز تحذف ال Permissions بتاعت user معين ، عن طريق الامر

**setfacl -x u:ali file**

او

**setfacl -x u:ali /work**

طیب لو عایز بقى تحذف كل ال Permissions بتاعت اى user او group من  
ال Directory كله

**setfacl -b /work**

ومتنساش ال man page بتاعت ال setfacl

**-b =====> remove all**

طیب فى شوية errors حلوين ممكن يقابلوك لما تعمل umount لای File  
System اولال error ده

**umount: /work: target is busy.**

**(In some cases useful info about processes that use  
the device is found by lsof(8) or fuser(1))**

كده انت هتتفد الامر ده علشان تعرف انهى PID هى اللى بتستخدم  
ال File System ال busy ده

**lsof | grep '/work'**

وبعدها هتعمل kill لل Process اللى بتستخدم ال File System ده عن طريق الامر

## **kill -9 PID**

وبعدها تحاول تعمل umount لل File System ، ولو قابلتك الرسالة دى

**Isof: WARNING: can't stat() fuse.gvfsd-fuse file system  
/run/user/1000/gvfs**

فده معناه انك لازم تعمل umount للمسار ده الاول

**/run/user/1000/gvfs**

وبعديها ترجع تنفذ الامر ده

**fuser -kim /work/**

وبعدها تنفذ ال mount بتاعك بطريقة عادية خالص ، وهتلاقيها اتنفذ

## **انتهى موضوع ال Access Control List**

# 9-YUM

مبدئيا كده ال Package هي ناتج عملية ال Compile للكود بالاضافة لبعض الملفات زي ال Libraries او ال Fonts او ال Backgrounds ، يعنى ملف ال binary بتاع ال source code زائد بقية الملفات

مايكروسوفت عندها Package Manager 2 ، اللي هما ال exe وال msi ، علشان كده بتلاقى الملفات او البرامج يعنى الامتداد بتاعها بيكون .exe ، وال msi بدأ ينتشر بداية من 2012

وهتلاقى بعد كده البرامج امتدادها .msi وهو اختصار لـ **Microsoft Installer**

طبعا RedHat ال Package Manager بتاعها هو rpm ، انما Debian هو deb

بالرغم من مشاكل ال Dependencies ، اللي موجودة عند معظم ال Package Managers ، الا انه رحمك برضو من شوية حاجات ، عندك مثلا لو جيت تعمل Software ل install معين فانت الاول هتعمله Compile وناتج ال Compile ده اللى هو ال binaries هتروح تحطها فى مجلد ال bin/ او ال user/bin/ او هتخطها فى usr/sbin/

وبعدها لو فى Fonts هتروح تنقلها فى مجلد ال Fonts ولو فى Backgrounds  
يبقى هتنقلها فى /usr/share/backgrounds/ ولو فى Icons يبقى برضو  
هتنقلها فى مكانها ولو فى Configuration Files هتنقلها برضو طب لو ال App  
ده ليه Data Base ، يبقى برضو هتنقلها فى مكانها

بما ان ال Package Mangers زى ال rpm وال deb وال msi وال exe وال  
dmg ، بما انهم حلوا جزء من المشكلة اللى كانت عندنا وهى عملية ال  
software installation

طلعنا بقى ما يسمى بال YUM وال APT وال Pacman وغيرهم ، اى بقى  
حكايتهم ، قالك بس انا هخلى عندى سيرفر كده خاص وهسميه بال Yum  
Server ، وهخط على اجهزة المستخدمين ما يسمى بال yum client ، ولما اى  
مستخدم يحب انه يعمل install ل application معين ، فال yum client ده  
هروح يسال ال yum server ، وبعدها ال yum server هيرد عليه ويقول ا اذا  
كان ال app ده موجود ولا لأ وكمان هيقوله اى ال dependencies اللى  
محتاجها ال application ده ، من الاخر ال yum client بيعمل نفس فكرة ال  
browsers اللى هى عملية ال resolve



بالإضافة لكده وهو ان ال Yum Server هيكون عنده ما يسمى بال Meta Data ،  
ودى عبارة عن XML Files فيها تفاصيل كل ال Packages اللى موجودة وال  
Dependencies بتاعت كل app فيهم

طبعا متفتكرش ان ال Yum هيغنيك عن ال rpm ، لا لا خالص ال Yum  
بيستخدم ال rpm ك Backend ليه ، كل فكرة ال Yum انه هيسخدم ال meta  
data اللى موجودة على السيرفر علشان يعرف ال Dependencies بتاعت كل  
App انت هتعمله install

ندخل بقى فى العمل ، اولا انت عارف لما بتعمل

## yum search codeblocks

وتجيك النتائج دى

<b>epel/x86_64/metalink</b>	<b>  29 kB 00:00:04</b>
<b>base</b>	<b>  3.6 kB 00:00:00</b>
<b>epel</b>	<b>  3.2 kB 00:00:00</b>
<b>extras</b>	<b>  3.4 kB 00:00:00</b>
<b>rpmfusion-free-updates</b>	<b>  3.0 kB 00:00:00</b>
<b>rpmfusion-nonfree-updates</b>	<b>  3.0 kB 00:00:00</b>
<b>ruussianfedora-free</b>	<b>  4.2 kB 00:00:00</b>
<b>ruussianfedora-free-updates</b>	<b>  3.8 kB 00:00:00</b>
<b>updates</b>	

كل دى معناها انه بيروح يعمل download لل meta data اللى موجودة على السيرفر ، بس خلى بالك انت مش فى كل مره هتستخدم فيها ال yum هيروح يعمل download لل meta data دى  
لا لا ، هو هيعمل download ليها فى حالتين وهما انك فى حالة ان دى كانت اول انت تعمل connect من جهازك على جهاز ال Yum Server والحالة الثانية ان لو السيرفر بتاع ال Yum ذات نفسه عمل update لل Packages اللى عليه وبالتالي هيعمل update لل meta data وبالتالى انت كمان لما تيجى تستخدم ال yum client هيروح يعمل check على ال meta data الاول اللى موجودة على السيرفر ، لو لقيها اتغيرت هيضطر يعملها download عندك على الجهاز ، لذلك عملية ال installation او اول update انت هتعمله بتكون بطيئة شويتين لانه بيروح يعمل update لل meta data الاول

وبرضو فى عملية ال installation هو بيبدأ بال dependencies الاول طبعا ، وبالمناسبة ال repository هو عبارة عن ال Yum Server

طيب تعالى بقى نعرف احنا عندنا كام Yum Server الجهاز بتاعتنا معمول ليه configured automatic انه يروح يجيب منهم ، عن طريق الامر

**yum repolist**

وده الناتج

**Loaded plugins: fastestmirror, langpacks**

**Loading mirror speeds from cached hostfile**

**\* base: mirror.airenetworks.es**

**\* epel: mirror.airenetworks.es**

**\* extras: mirror.airenetworks.es**

**\* rpmfusion-free-updates: www.mirror-service.org**

**\* rpmfusion-nonfree-updates: www.mirror-service.org**

**\* russianfedora-free: ftp.russianfedora.pro**

**\* russianfedora-free-updates: ftp.russianfedora.pro**

**\* updates: mirror.airenetworks.es**

<b>repo id</b>	<b>repo name</b>	<b>status</b>
<b>base/7/x86_64</b>	<b>CentOS-7 - Base</b>	<b>9,911</b>
<b>epel/x86_64</b>	<b>Extra Packages for Enterprise Linux 7 - x86_64</b>	<b>12,602</b>
<b>extras/7/x86_64</b>	<b>CentOS-7 - Extras</b>	<b>313</b>
<b>rpmfusion-free-updates/x86_64</b>	<b>RPM Fusion for EL 7 - Free - Updates</b>	<b>237</b>
<b>rpmfusion-nonfree-updates/x86_64</b>	<b>RPM Fusion for EL 7 - Nonfree - Updates</b>	<b>48</b>
<b>russianfedora-free/7/x86_64</b>	<b>Russian Fedora for EL 7 - Free</b>	<b>9</b>
<b>russianfedora-free-updates/7/x86_64</b>	<b>Russian Fedora for EL 7 - Free - Updates</b>	<b>35</b>
<b>updates/7/x86_64</b>	<b>CentOS-7 - Updates</b>	<b>711</b>
<b>repolist: 23,875</b>		

نمسكهم واحدة واحدة بقى ، اولا انت عندك repository اسمها base وعندك واحدة تانية اسمها epel وهكذا بقى ، كده انت بالمنظر كده الجهاز بتاعك معمول ليه configured انه يجيب من 8 repository ، وعدد ال Packages اللي موجودة فى كل ال repositories او موجودة على كل ال Yum Servers هو

**repolist: 23,875**

يعنى بالظبط 23 الف و 875

طيب ازاي بقى ال OS عندي عرف ان فى repository بالاسم الفلانى ده ؟؟  
هتلاقى الكلام ده كله Configured فى المجلد

**/etc/yum.repo.d/**

كل الملفات اللي موجودة فى المسار ده الامتداد بتاعها بيكون .repo.

CentOS-Base.repo	CentOS-Vault.repo	rpmfusion-free-updates-testing.repo
CentOS-CR.repo	epel.repo	rpmfusion-nonfree-updates.repo
CentOS-Debuginfo.repo	epel-testing.repo	rpmfusion-nonfree-updates-testing.repo
CentOS-fasttrack.repo	google-chrome.repo	russianfedora-free.repo
CentOS-Media.repo	opera.repo	russianfedora-free-updates.repo
CentOS-Sources.repo	rpmfusion-free-updates.repo	russianfedora-free-updates-testing.repo

معلومات ال Package هتلاقيها موجودة فى

## yum info vim-enhanced

كمثال يعنى

وكمان هتعرف ال Package دى موجودة فى انهى repository بالضبط

طيب لو عندك ملف Configuration معين وعايز تعرف الملف ده انهى Package هى اللى عملته install ، ممكن تنفذ الامر ده

## yum provides /etc/sysctl.conf

هيجبك اسم ال Package اللى عملت create او نزلت الملف ده

**initscripts-9.49.41-1.el7.x86\_64 : The inittab file and the /etc/init.d scripts**

**Repo : base**

**Matched from:**

**Filename : /etc/sysctl.conf**

**initscripts-9.49.41-1.el7.x86\_64 : The inittab file and the /etc/init.d scripts**

**Repo : @base**

**Matched from:**

**Filename : /etc/sysctl.conf**

فانت عندك الملف ده ممكن يكون جايلك من ال Repository اللى هى ال Base ، او ممكن يكون جايلك من ال Anaconda اثناء عملية التسطيب

طيب افرض بقى ان ال meta data اللى عندك حصلها Corruption او مثلا انت عايز تنصفها ، فانت عندك الامر

## **yum clean**

ومن خلاله بقى ممكن تعمل clean لل **dbcache** او لل meta data او لل Packages

بالنسبة للملفات اللى انت بتنزلها عن طريق ال yum install ، مبدئيا كده انت عندك ملف ال Configuration بتاع ال yum موجود فى

## **/etc/yum.conf**

فى الملف ده فى variable اسمه KeepCashe وقيمته ال default هى zero  
يعنى بتقوله لما تنزل Package معينة ابقى امسح ملف ال rpm. بتاع ال Package وده اصلا وظيفة الامر yum clean packages يعنى انت هتنصف ملفات ال rpm. اللى نزلت ، والملفات دى هتنزل فى المسار ده

## **/var/cache/yum/x86\_64/7**

وبالتالى دى فكرة انك تخلق ال /var فى بارتيشن لوحده وذلك فى حالة انك هتخلق ملفات ال rpm. متمسحش وكذلك برضو ال Cashe ومتنساش ال updatedb

لو تفكر فى عملية ال update لما كنت بتوقف ال update وكنت تيجى ترجعه كنت بتلاقى بعض ال Packages لونه ابيض ، ده معناه ان ال Package دى موجودة فى الكاش والحد الكافى لانك تسبب الكاش بتاع ملف ال rpm. هو اسبوعين

وطبعا اى Transaction بتعمله عن طريق ال yum ، هتلاقيه موجود فى

## **/var/log/var.log**

طيب لو عايز تعمل list لكل ال Transactions اللى حصلت لل Yum ، نفذ الامر ده

## **yum history**

ID	Login user	Date and time	Action(s)	Altered
14	ahmed <ahmed>	2018-06-22 21:14	Install	10 EE
13	ahmed <ahmed>	2018-06-22 21:06	Install	1
12	ahmed <ahmed>	2018-06-22 21:05	Install	1
11	ahmed <ahmed>	2018-06-22 21:02	Reinstall	1
10	ahmed <ahmed>	2018-06-22 20:31	Install	1

طبعا لو حبيت تعرف اى اللى حصل فى Transaction رقم 11 مثلا ، عن طريق الامر

## **yum history info 11**

يمكن برضو uninstall ل Transaction معينة عن طريق الامر

## **yum history undo 5**

وطبعا ممكن تستخدمه انك تعمل remove ل Package وكل ال Dependencies بتاعتها

من الاخر الامر yum history كده كبير فشخ ، خد بالك منه

ودى الميزة اللى بتميز ال Yum عن ال rpm وهى ان ال Yum لما بيتعامل مع كل ال Packages مثلا فى عملية ال installation بيتعامل عن طريق ال Transactions بمعنى انك لو سطبت مثلا الاباتشى وراح منزل معاه مجموعة packages تانية ، فانت عندك ان الاباتشى واى حاجة هتنزل معاه ال Yum بيعاملهم على انهم transaction على بعض ، يعنى كأنهم حاجة واحدة ، وبالتالي لو حبيت تشيلهم كلهم ، يبقى بكل بساطة اعملهم undo



# 10-RPM

موضوع ال RPM

طيب زى ما قولنا ان ال Packages بتاعتنا او بمعنى اصح اللى هى ملفات ال **rpm**. ما هى الا عبارة عن Container لل Binary وشوية Fonts وشوية scripts لو موجود يعنى

تعالى نبص بصة سريعة على تفاصيل اى Package ، هناخد ال Package دى كمثال

**vim-enhanced-7.4.160-4.el7.x86\_64**

اولا بقى

**vim-enhanced =====> Package Name**

**7.4.160 =====> Package Version**

**4 =====> Minor Package Version**

**el7 =====> It Means That This Package Is Compitable with  
RedHat Enterprize Linux Version 7**

**x86\_64 =====> Means That This Package will run on a x86\_64**

**Processor**

**noarch =====> Means That This Package Is Compitable with any**

**Type Of Processor**

طيب اى الفرق بين ال Package Version وال Minor Package Version ؟ بكل  
بساطة ال Package Version دى بيكون عبارة عن Update كبير لل Package  
زى ان ال Package بيتم اعادة كتابتها من تانى  
انما ال Minor Package Version دى بيكون عبارة عن release بسيط ، زى مثلا  
انهم صلحوا bug معينة فى ال Package

بالمناسبة وانت بتسطب Package عن طريق ال RPM ممكن تستخدم ال  
options الجميلة دى

**rpm -ivh**

**i =====> install**

**v =====> verbose**

**h =====> hashed** يعنى من الاخر كده ورنى انت وصلت لفين بالظبط بيحسبك بالنسبة المؤوية

انما الامر

## **rpm -qa**

هيعرضلك كل ال Packages اللى معمول ليها install على ال system عندك طبعا برضو معلومات اى Package انت نزلتها لوعايز تعرفها بيكون عن طريق الامر

## **rpm -qpi httpd-xxxx**

بس لاز تكون ال Package موجودة يعنى ال option اللى هو **pi** هيشغل بس لو ال Package موجودة ، قصدى يعنى ال Source بتاعتها اللى هو امتدادها **rpm**.

انما بقى لو ال Package معمول ليها install بالفعل ، فانت مش محتاج تكتب ال p ، اكتب على طول

## **rpm -qp**

## **rpm -qp httpd**

طيب طبعا زى ما احنا عارفين ان ال rpm مش بيعمل resolve لل Dependencies بتاعت ال Package وبالتالي كان الطبيعى اننا نستخدم ال yum

وبما ان ال Package احنا منزلينها اصلا ، فكل اللى احنا عايزنه ، اننا هنكتب

## **yum localinstall package.rpm**

وهو هيروح بقى يعمل **resolve** لل dependencies بتاعت ال Package دى من  
ال Yum Server

نفس الكلام برضو لو عندك ملف وعازيز تعرف انهى Package هى اللى عملته ،  
عن طريق الامر

**rpm -qf /etc/httpd**

اوبالامر ده مهم جدا بالنسبالك ، افرض بقى ان انت عندك Package.rpm  
وعازيز تعرف اى الناتج بتاع ال Package دى ، يعنى قصدى ال Package دى  
هترمى نفسها فين بالضبط عن طريق الامر

**rpm -qpl httpd.rpm**

طب لو ال Package اصلا معمول ليها installed اصلا ، وعازيز تعرف ملفاتها  
فين بالضبط ، عن طريق الامر

**rpm -ql httpd | less**

الامر الخطير التانى بقى وهو انك لو عازيز تجيب ملفات ال Configuration  
بتاعت مثلا ال httpd ، عن طريق الامر

**rpm -qc httpd**

وكذلك الامر برضو لو عايز تعرف اي ملفات ال Documentations بتاعت  
Package معينة اللي هيثم ارفاقها مع ال Package او هتنزل مع ال Package

يعنى من الاخر كده خلى بالك من الامر

## **rpm -q**

واي حرف بقى بعد حرف **q**

يبقى لو ال Package معمول ليها installed يبقى مش هتضيف ال option اللي  
هو **p** ، انما لو مش معمول ليها installed يبقى هضيفه وتكتب ال Path كامل  
بتاع ال Package

**خلى بالك وانت بتعمل force installation ل Package معينة عن طريق ال**

**RPM ، لان ده ممكن يسببك System Broken وده option غير**

**recommended خالص وياريت متعملوش**

طيب دلوقتي بقى انت عندك Package ازاي بقى تتأكد ان ال Package دي  
Trusted ؟؟؟ مبدئيا كده اي Package بتطلع من RedHat ، ريد هات بتحت  
للباكج دي مايسمى بال Signature عاملة زي الختم كده  
طيب لو عايز تعمل verify لل key بتاع Package انت منزلها ، بيكون عن  
طريق الامر

**rpm -K httpd.rpm**

وطبعاً لو عايز تعرف ال Keys اللي موجودة عندك على الجهاز ومعمول ليها Trust ، عن طريق الامر

```
rpm -qa | grep -i key
```

يبقى انت مش اى **Package** تعملها **download** تروح تعملها **install** ، لا لا ،  
اتأكد الاول من ال **Key** بتاعها ، والا ممكن يكون فيها **Backdoor** وغيره

الجماعة ال Developers بتوع RedHat و CentOS بيحطوا ال Keys بتاعهم فى  
المسار

**/etc/pki/rpm-gpg/**

يبقى انت هتروح للمسار ده وتعمل لل Keys تعملها import عن طريق الامر

```
rpm --import RPM-GPG-KEY-CentOS-7
```

وترجع بقى بعدها تعمل check لل Package اللي انت نزلتها

وديما اعمل verify قبل ما تعمل install وخليك System admin ذكى 🤖🤖

بالمناسبة لو ال Key مش موجود عندك على الجهاز ، انا قصدى ان ال Key  
اللى Developers بتوع CentOS حطوه مع التوزيعه ، علشان تقارنه ببقية ال  
Keys اللى موجودة مع كل Package

فانت هتنزله ، ولو انت بستخدم ال rpm علشان تسطب ال Package فال  
rpm مش هيهتم بحوار ال key لكن هيطلعلك warning ، انما لو هتسطب عن  
طريق ال yum ، فا هو مش هيكمل غير لما انت توافق انه يعمل import لل  
Key ده

# 11–Source Code Installation

اول حاجة هنعملها وهى اننا هنعمل Yum Server ، واكيد علشان نعمل ال Yum Server فاحنا محتاجين ل Packages ، وطبعاً محتاجين اننا نعمل host لل Packages دى عن طريق ال http او ال FTP ممكن نسطب ال vsftpd ، وطبعاً هو بي host الملفات بتاعته فى

## **/var/ftp/pub**

يبقى احنا ممكن مثلاً نجيب اسطوانة CentOS Every Thing وننقل كل ال Packages اللى عليها نوديتها فى `var/ftp/pub/` ومتنساش انك لازم تعمل ال meta data بتاعت كل ال Packages دى ، طب انت هتعمل ملفات ال xml بتاعت ال meta data دى ازاي ؟  
انت هتسطب Package اسمها **createrepo** ، وبعدين بقى هتعمل ملفات ال meta data عن طريق الامر

## **createrepo -v /var/ftp/pub**

وبعدها هتلاقى مجلد اسمه repodata ات create هناك فيه كل ملفات ال XML

طبعاً السؤال بتاع احنا ليه استخدمنا ال FTP ؟ بكل بساطة علشان ال Server بتاعتنا يقدر انه يشير الملفات بتاعته ، بس خلى بالك انت ممكن تستخدم برضو ال http ، الاتنين واحد ، اللى يعجبك فيهم استخدمه



تعالى بقى نسطب ال Package الخاصة بال FTP Client على جهاز ال Client  
علشان نعمل بيها Test على جهاز السيرفر ، عن طريق انك تسطب ال  
Command Line Utility اللى اسمها ftp

## **yum install ftp**

يبقى خلى بالك ان دى ال Client Package وليست ال Server Package

تعالى بقى على جهاز ال client وجرب انك تعمل access لجهاز ال YUM Server  
عن طريق الامر

## **ftp 192.168.1.11**

لو جاتلك الرسالة دى

## **ftp: connect: No route to host**

معناها انه مش قادر يعمل ping على الجهاز ده ، طيب ابقى جرب برضو انك  
تعمل ping على جهاز السيرفر

## **ping 192.168.1.11**

ولو قدرت تعمل ping ، يبقى معناه ان الجهاز بتاعك مش Accessible

الحاجة الثانية انك ممكن تروح تسطب ال ftp على جهاز ال yum server  
وتخليه يتصل بنفسه عن طريق ال **ftp 192.168.1.11** وتستخدم ال  
anonymous ك user وطبعا مفيش باسورد هنا

ولو لقيته اتصل يبقى تشوف ال **fire wall** بتاع جهاز ال yum server ، لو  
لقيته شغال ، يبقى تقوله لو سمحت اسمحلى ان users من بره يعموا  
connect على الجهاز عن طريق الامر التالى كده انت هتسمح ل service ال  
ftp انها تعدى

### **firewall-cmd --add-service=ftp**

وطبعا لو عايز تشوف ال Allowed Services اللى عندك ممكن تستخدم الامر  
ده

### **firewall-cmd --list-all**

طبعا خليك فاكر ان اى حاجة بتعملها مع ال **firewalld** بتعمل بشكل مؤقت ،  
يعنى اول ما تعمل reboot كل حاجة هترجع لطبيعتها تانى ، يبقى انت لازم  
تخلي الكلام ده بشكل دائم عن طريق الامر

### **firewall-cmd --add-service=ftp --permanent**

تعالى بقى نرجع نجرب فى جهاز ال client

### **ftp 192.168.1.11**

طبعا موضوع انك تسبب البورت بتاع ال ftp مفتوح لاي حد ، ده غلط كبير جدا  
بس بعد كده انت المفروض تخليه allowed ل subnet معينة بس

بعده كده احنا هنعمل Configure لجهاز ال Client انه ياخذ ملفات ال rpm من  
ال ftp سيرفر اللي احنا عملناه ، يبقى احنا هنروح للمسار

**cd /etc/yum.repo.d/**

وطبعا متنساش ان اي ملف في المجلد ده بيكون امتداده

**.repo**

علشان تختبر ال ftp سيرفر بتاعك ممكن تيجي كده تنقل كل الملفات اللي  
موجوده في المجلد ده

**mv /etc/yum.repos.d/\* /root/ status=progress**

وتعالى بقى نعمل احنا ملف configuration خاص بينا

**cd /etc/yum.repo.d/**

**vim local-server.repo**

وبعدها هتضيف فيه الكام سطر دول

## **[Local-Server]**

**name=Local Yum Server**

**baseurl=ftp://192.168.1.11/pub**

**enabled=1**

**gpgcheck=1**

**gpgkey=ftp://192.168.1.11/pub/RPM-GPG-KEY-CentOS-7**

اول سطرین دول الاسم بتاع ال repo ، وبعدها انا بحدده ال baseurl

وبعدها بقوله ان ال repo دی enabled یعنی اشتغل منها ، انما لو خلیت قیمتها بصفر ، یبقی انا كده مش مخلیها متفعلة ، ملهاش لازمة ده

وبالنسبة لل **gpgcheck** ، فالو انت عایزه یعمل check لل key بتاعت كل Package فانت ممكن تخی قیمتها بواحد ، بس لازم تقوله تحت فین مكان ال key ده ، طبعا انا نسخت ال key ده من المسار

**/etc/pki/rpm-gpg/**

وتاخذ ای Key من اللى هناك

تمام كده ، اعمل بقى **yum clean all** ، او ممكن متعملوش ،وبعدها ممكن تجرب انك تعمل search على Package معينة زى ال vim مثلا ، هتلاقيه يبيحث فى ال Local History وممكن برضو تتأكد ، هو بيحب من انهو repo ، عن طريق الامر

## **yum repolist**

على فكرة موضوع انك تعمل local yum server دا شىء جميل جدا ، لو انت مثلا شغال فى شركة ولا حاجة ، فانت مثلا ممكن تنزل كل ال Packages وتعملها sync ، وتعمل configure لكل اجهزة ال Client وتخليها انها تنزل ال Packages من ال Yum Server ، وبكده انت مش هتخلى فى load كبير على النت عندك

بالنسبة بقى لموضوع ازاي تسطب اى Package من ال Source Code فى فيديو حلو اوى على ال YouTube بيشرح الموضوع ده بالتفصيل وببساطة شديدة ، لازم تتفرج عليه ، وده لينك الفيديو

<https://www.youtube.com/watch?v=W9JcK70kThI>

## آخر شوية ملاحظات عامة

فى نوعين من الملفات الامتداد بتاعهم ممكن يكون

**.ko**

او

**.so**

ال **ko** هى اختصار ل **Kernel Object** ، وال **so** هى اختصار

ل **system object** ، عادة ملفات ال **so** بتكون عبارة عن libraries

ال system بيحتاجها علشان يعمل حاجة معينة

انما الملفات **ko** دى الكرنال هواللى بيستخدمها ، ودى ممكن تكون **drivers**

**baseurl = file:///srv/my/repo/**

دی اجابة هما ليه 3

**The 'baseurl' line is the path that machine uses to get to the repository.**

**If the machine has direct access to it or mounts it as a filesystem you can use a baseurl line like:**

**baseurl = file:///srv/my/repo/**

**There are 3 slashes (/) following the file:, not 2. That is correct.**

**If you access the file via an http or https server you would use something like:**

**baseurl = http://servername/my/repo**

## الشرح بقى بالعربى

بص انت كده كده هتحت ال **Slashes 3** فى حالة ان انت عندك اسطوانة  
وعليها Packages مثلا ومعمول ليها mount فى مكان ما ، للمرة الثانية ال 3  
Slashes دول ثابتين

يعنى مينفعش يتغيروا ، زى بالظبط ما انت بتروح تكتب فى المتصفح عندك

**http://**

انت اهو حطيت اتنين **slashes** ، كذلك الامر برضو بالنسبة لل local repo انت  
هتحت ثلاثة slashes خليك فاكر النقطة دى كويس اوى

وبرضو لو عايز تعرف ليه بنكتب 3 Slashes فدى الاجابة من موقع Digital  
Ocean ، واحد هناك سأل السؤال ده

**<https://www.digitalocean.com/community/tutorials/how-to-set-up-and-use-yum-repositories-on-a-centos-6-vps>**

وده السؤال اهو

```
etc/yum.repos.d/custom.repo
[customrepo]
name=Custom Repository
baseurl=file:///repository1/
enabled=1
gpgcheck=0
```



?what is that file:/// (what is the meaning of these 3 slashes )

وده كان رد الموقع

**The first two slashes, as seen in "file://"** represent the URL, as per [https://en.wikipedia.org/wiki/File\\_URI\\_scheme](https://en.wikipedia.org/wiki/File_URI_scheme)

**The following (third) slash denotes the location of the filesystem, i.e. /repository1.**

الرد بكل بساطة يقول ان الاتنين Slashes دول بيمثلوا ال URL يعنى مثلا سيادتكم بتروح تكتب

**http://**

فانت هنا دي من اساسيات كتابة ال url وابقى خش برضو على ويكيبيديا علشان تعرف ال URL Scheme ، وبعد كده بقى ال Slash الثالثة دي بتمثل مكان ال file system بتاعك ، او بمعنى اخر المكان بتاع ال directory اللي عليها ال repo هي معمول ليها mount فى انهى مكان بالظبط

يعنى مثلا انت عندك directory معمول ليها mount هنا

**/mnt**

فانت هنا بقى وانت بتعمل ال repo ال Slash الثالثة بقى بتمثل  
ال **root file system** او بتمثل ال **mounted file system**  
والمقال بتاع ويكيبيديا حلو اوى ، لازم تقرأه

[https://en.wikipedia.org/wiki/File\\_URI\\_scheme](https://en.wikipedia.org/wiki/File_URI_scheme)

طيب خد عندك المثل ده برضو

**file:///etc/fstab**

بص يا سيدى اول اتنين Slashes دول بيمثلوا ال Local Host على حسب ما  
ويكيبيديا بتقول ، طيب وبعد ال Local Host بقى بياجى عندنا ال root file  
system ذات نفسه  
وهنا مثلا عندك ان ال

**/etc/fstab**

موجودين بشكل مباشر تحت ال Slash الثالثة اللى هى بتمثل  
ال **root file system**

بس يا سيدى دى كل الحكاية

انتهى الجزء الثانى

