

جزوه آموزشی سیستم عامل لینوکس پیشرفته

برگرفته از کلاس های مهندس جلال حاجی غلامعلی در مرکز آی تی دانشگاه صنعتی شریف

نسخه 0.1

تهییه و تنظیم: نجیبیه پردیس

<http://laitec.ir/>



Linux Administration

12.....	start up and shut down
12.....	نحوه بالا آمدن سیستم
12.....	مراحل Post
27.....	مفهوم runlevel
43.....	بررسی Super Server در لینوکس (xinetd)
45.....	Extended Internet Daemon =XINETD
47.....	telnet
53	Chargen
54	iPop 3
56.....	بررسی پروتکل telnet
58.....	فعالیت های in.telnet
61.....	فعالیت های login
65.....	فعالیت های shell

66.....	بررسی گزارش عملیات Linux
66.....	سوالات لینوکس پیشرفته
71.....	فرمان shutdown چه می کند
71.....	به فایل inittab نگاهی انداده و powerfail را تشریح کنید
72.....	سوالات بخش xinetd
77.....	آیا می دانید چه تفاوت هایی بین inetd و xinetd وجود دارد؟
77.....	سوکت چیست؟ (از نوع نرم افزاری)
77.....	گزارش عملیات لینوکس
80.....	دلایل ایجاد log ها
82.....	ساختار logging
88.....	rotating Log
88.....	دلایل log فایل های rotate
91.....	فعالیت های shell
92.....	بررسی حافظه مجازی در لینوکس (swap)
94.....	اضافه نمودن حافظه موقتی swap

96.....	مراحل ایجاد حافظه موقتی swap
106.....	بررسی روش‌های محدود کردن کاربران در لینوکس (Limitation)
115.....	The Cron System
115.....	نحوه اجرای اتوماتیک برنامه‌ها در لینوکس
116.....	مولفه‌های cron
128.....	core file
132.....	امنیت در لینوکس
133.....	فاجعه و بازیابی
134.....	دسته بندی فاجعه‌ها
136.....	Information Security Management System
136.....	امنیت در چه حد
137.....	انواع تهدیدها
137.....	3 رکن اصلی امنیت
138....	لیست کلی تهدیدها
140.....	چیست ؟ ISMS

140.....	سیاست‌گذاری امنیتی
141.....	بررسی فعالیت‌های راهبر لینوکس
141.....	اتوماتیک نمودن فعالیت‌ها تا حد ممکن
142.....	نمودن نمودن
144.....	ارتباط هرچه بیشتر و بهتر با کاربران
145.....	منابع خود را شناسایی نمائید
145.....	کاربران خود را خوب شناسایی کنید
145.....	شغل و موقعیت خود را بهتر بشناسید
146.....	امنیت نمی‌تواند کم اهمیت گرفته شود
146.....	پیش‌بینی برای آینده (آینده‌نگری)
146.....	آمادگی لازم برای مواجهه با اتفاقات غیرقابل پیش‌بینی
150.....	System Accounting
157.....	Dynamic Host Communication Protocol (DHCP)
163.....	بررسی Raid و Disk Strping در لینوکس
165.....	(Disk Mirroring) RAID 1

169.....	Cooked And Raw Devices بررسی برد و دستگاه
172.....	Iptables In Linux
178.....	Linux kernel
189.....	بررسی پروتکل FTP
193.....	یک توزیع update
195.....	چیست؟ bin
197.....	کاربرد Selinux چیست؟
199.....	LDAP
200.....	LDAP Authentication
203.....	Network Information System
206.....	Pluggable Authentication Modules
208.....	web server
211.....	VLAN
214.....	پایگاه داده
214.....	ORACLE

215.....	MySQL
215.....	Postgre
216.....	Linux Performance And Tuning
216.....	linux memory architecture
216.....	درک معیارهای عملکرد لینوکس
218.....	Monitoring Tools
218.....	top
218.....	vmstat
218.....	uptime
221.....	tcpdump/ethereal
222.....	nmon
223.....	KDE system guard
223.....	ng the operating system
224.....	dmesg
224.....	ulimit

226.....	Daemons
227.....	تغییر runlevel ها
227	تغییر پارامترهای کرنل
229.....	tuning network subsystem
230.....	سامبا
232.....	Squid Proxy Server
232.....	دلایل استفاده از پروکسی
235.....	Redirectors
236.....	نصب squid
238	NFS

Linux Administration

مقدمه

در دوره LPIC2 باید به دستورات (command) لینوکس تسلط داشته باشید و در این دوره تمکز بیشتر روی سرویس‌هاست. در ابتدا تلاشمان این است که بالا آمدن سیستم را نشان دهیم. (از روشن کردن تا prompt گرفتن)

start up and shut down

بالا آمدن سیستم عامل را اصطلاحاً بوت شدن (boot) کامپیوتر می‌گویند و نرم‌افزاری که سیستم عامل را وارد حافظه می‌نماید اصطلاحاً بالا آمدن سیستم عامل را اصطلاحاً بوت شدن (boot) کامپیوتر می‌گویند و نرم‌افزاری که سیستم عامل را وارد حافظه می‌نماید اصطلاحاً bootloader یا bootstrap خوانده می‌شود.



در تکراس صبح‌ها که سرکار می‌رفتند چکمه (boot) را با کمک زبانه (strap) می‌پوشیدند.

نحوه بالا آمدن سیستم

به مجرد اینکه موبایل، کامپیوتر یا تلویزیون را روشن می‌کنید¹ post اتفاق می‌افتد؛ پردازنده به مکانی از بایوس² پرس می‌کند که در آنجا تست بسیار مقدماتی شامل تست پردازنده، حافظه و سایر قطعات انجام می‌شود.

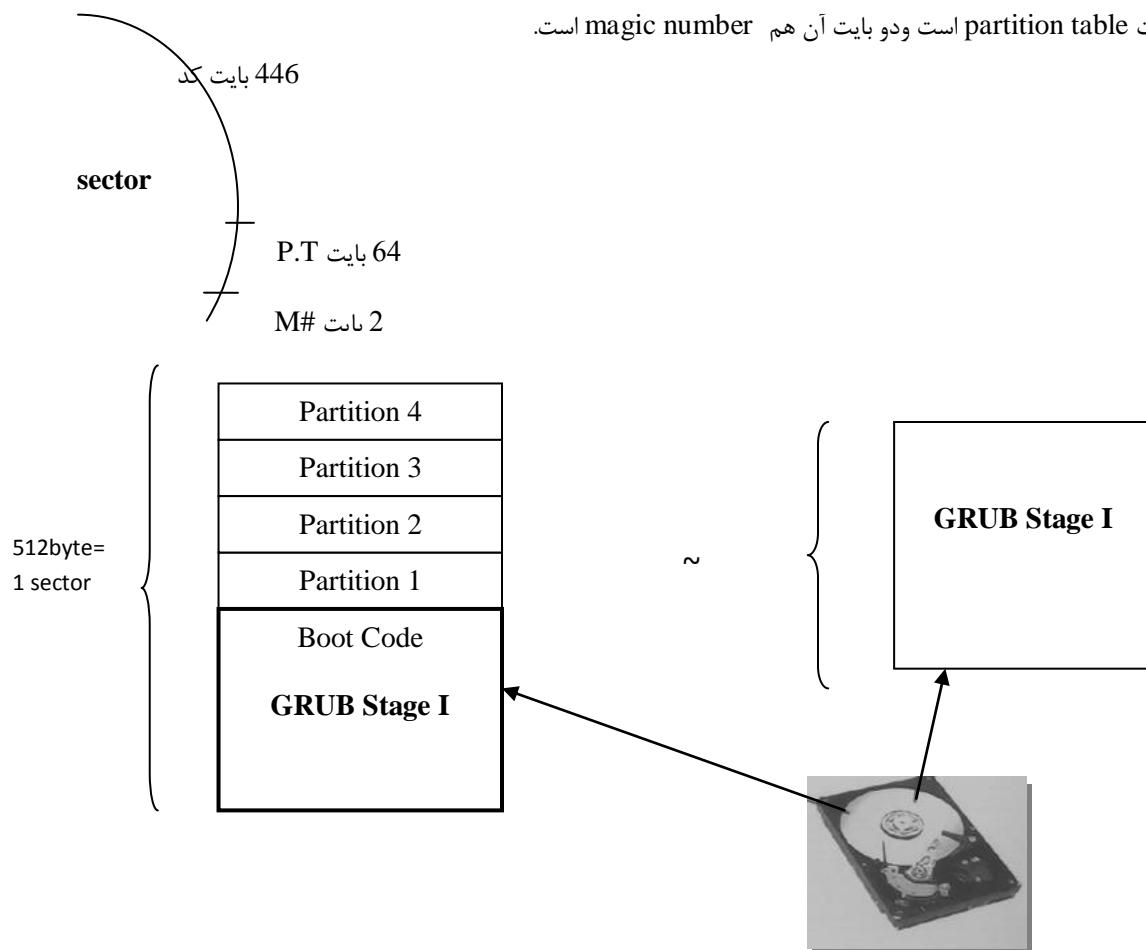
Post مراحل

- آیا برد مادر کار می‌کند؟
- آیا حافظه سالم است؟
- آیا فلاپی، دیسک سخت و CD-ROM به فرمان بایوس پاسخ می‌دهند؟

¹ Power On Self Test

² BIOS

اگر همه چیز درست بود ، بوقی را به صدا درآورده و به دنبال سکتور صفر اولین دستگاهی می‌گردد که در setup تعریف شده است. به عنوان مثال فرض کنید یک ورزشکار صحیح از خواب بیدار می‌شود در رخت خواب غلطی می‌زند (self-test) مثلاً احساس می‌کند پاپش درد می‌کندو امروز نمی‌تواند بود ؛ با مربی تماس می‌گیرد و اطلاع می‌دهد من امروز نمی‌آیم! روی مادربرد تراشه ای^۱ به نام CMOS قرار دارد که روی آن یک پایگاه اطلاعاتی کوچک (در حد چند صد بایت) تعییه شده است. پس از پایان تست سخت افزار، سیستم این پایگاه اطلاعاتی را بررسی می‌کند و سخت افزارها را به ترتیب لیست انتخاب می‌کند (موقع روشن شدن کامپیوتر با فشار کلید delete یا در لپتاپ‌ها بسته به مدل یکی از Fها وارد setup می‌شویم که در آنجا می‌توانیم انتخاب کنیم که سیستم با چه سخت افزاری بالا بیاید مثلاً فلاپی یا دیسک؟) مثلاً اگر گفتیم از فلاپی بالا بباید دستور on motor را به لایپی می‌فرستد ..فلاپی در جواب می‌گوید من آماده‌ام ولی فلاپی دیسکی داخل من نیست ، بعد به سراغ سخت افزار دوم در لیست می‌رود به عنوان مثال CD-ROM . سخت افزار پیش فرض^۲ عمدتاً دیسک سخت است سپس سعی می‌کند نرم‌افزاری را که روی سکتور 0 هارد است را بخواند و در آدرس 0x7c00 حافظه قرار دهد. هارد به قطعات مساوی 512 بایتی تقسیم شده است. در سکتور 0 446 بایت کد است که با اسمبلر نوشته شده است ، 64 بایت است و دو بایت آن هم magic number partition table است.

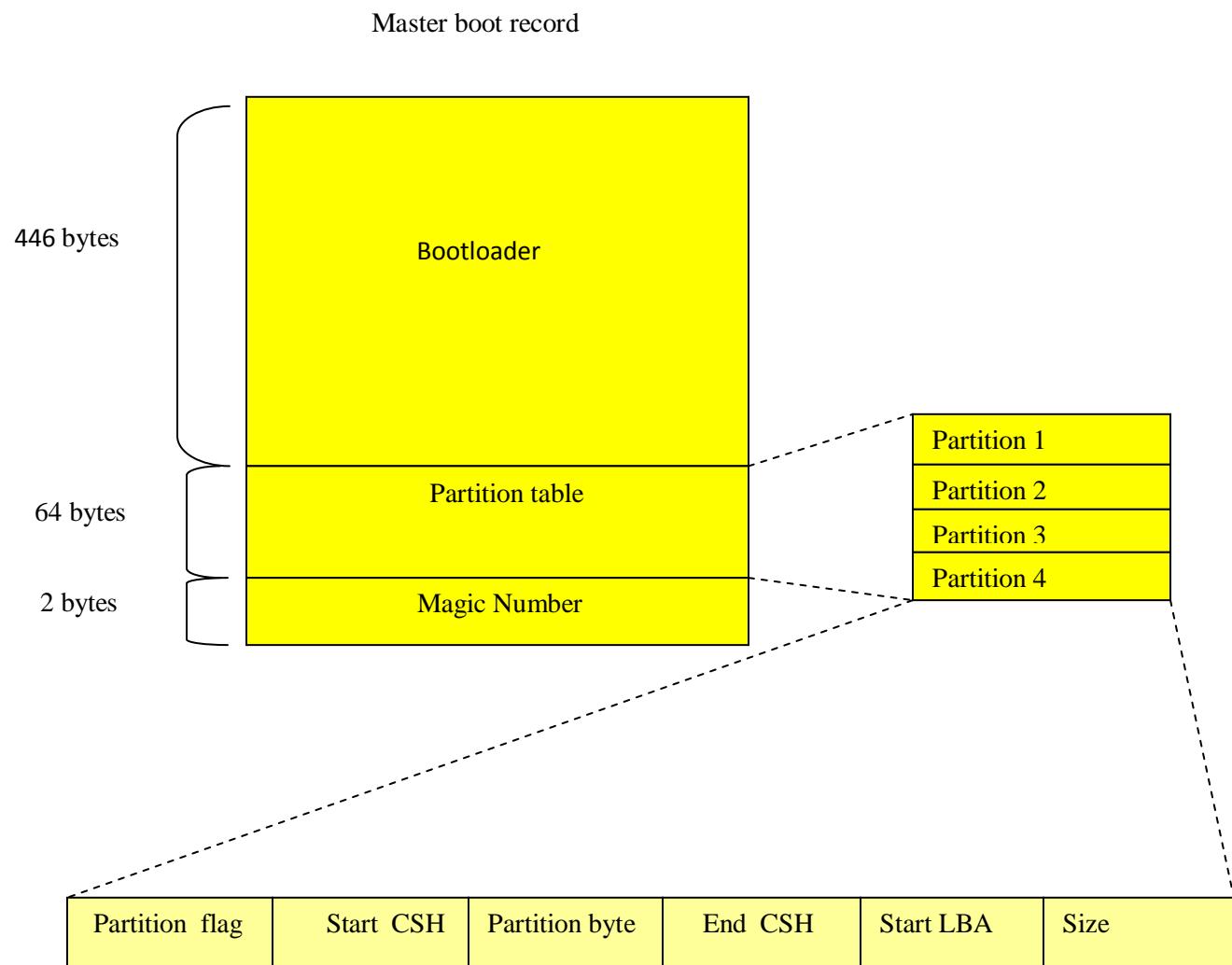


در بخش boot code دستورات اسمبلی مانند mov و shift قرار دارد.

¹ Chipset

² default

یک نرم افزار کوچک 4 الی 16 بایتی است که سعی می کند Active Partition partition table را بخواند و partition table کند. این نرم افزار خیلی کوچک است و نباید انتظار داشت که اگر partition table مشکل داشت پیغام انگلیسی چاپ کند در نهایت شماره خطای دهد: error1 یا error2.



فایل‌های شناخته شده یا مشهور هر کدام یک Magic Number دارد که سیستم عامل آنها را تشخیص دهد و بشناسدو مختص اینوکس نیست.

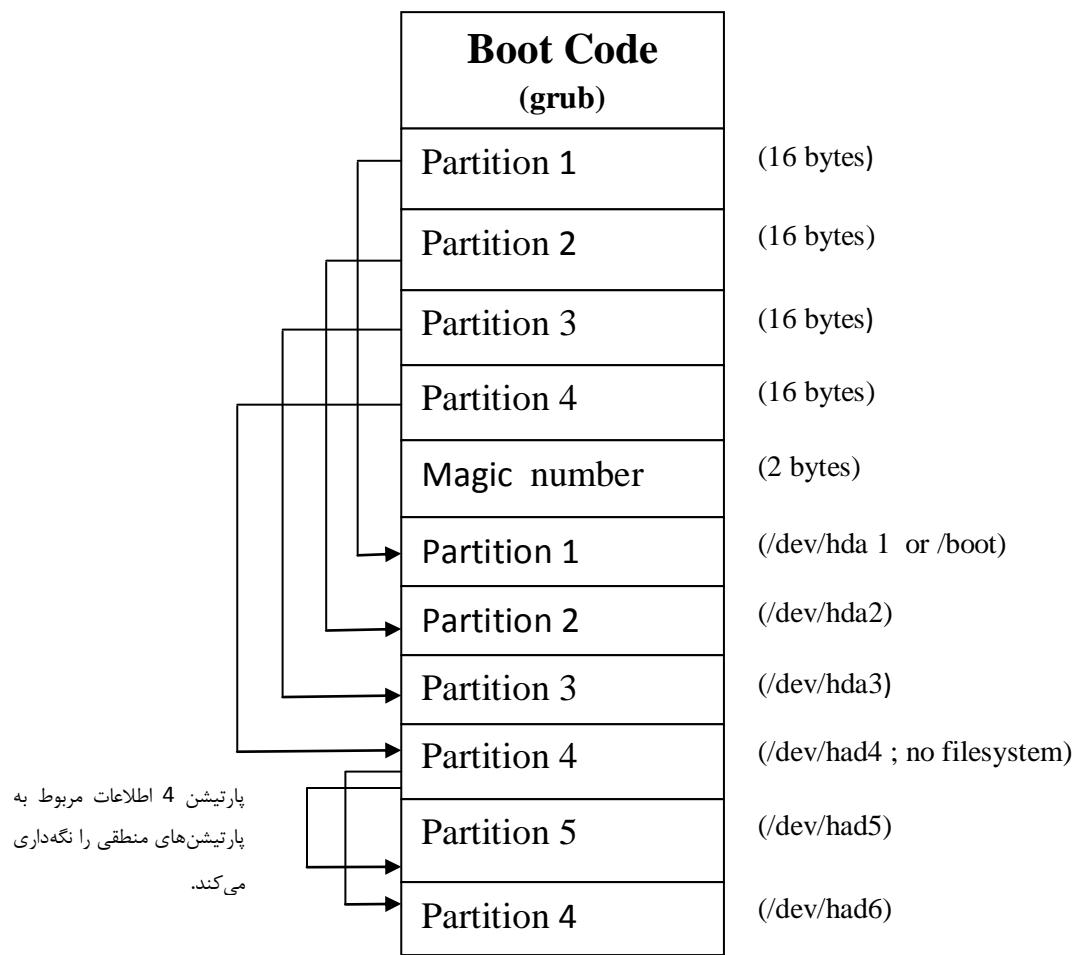
به عنوان مثال GIF8 که فرمت فایل تصویری یا mono برای فایل‌های صوتی به دستور زیر می‌توانید به فایلی که Magic number آن قرار دارند دسترسی پیدا کنید. و با AA55/ به کدهای زیر دستور می‌رسیم:

```
[n.pardis@lpi ~]$ $less /usr/share/magic
#offset is 128
>>19  ubyte  128
>>>(19.b-1)  ubyte  0x0      DOS Emulator image
>>>7  ulelong >0          \b, %u heads
>>>11  ulelong >0         \b, %d sectors/track
>>>15  ulelong >0         \b, %d cylinders

0xFE    leshort 0xAA55           x86 boot sector
>2     string  OSBS            \b, OS/BS MBR
# J\xf6rg Jenderek <joerg dot jenderek at web dot de>
>0x8C   string  Invalid\ partition\ table    \b, MS-DOS MBR
# dr-dos with some upper-, lowercase variants
>0x9D   string  Invalid\ partition\ table$ 
>>181   string  No\ Operating\ System$ 
>>>201   string  Operating\ System\ load\ error$ \b, DR-DOS MBR, Version 7.01
to 7.03
>0x9D   string  Invalid\ partition\ table$ 
>>181   string  No\ operating\ system$ 
>>>201   string  Operating\ system\ load\ error$ \b, DR-DOS MBR, Version 7.01
to 7.03
>342   string  Invalid\ partition\ table$ 
>>366   string  No\ operating\ system$ 
:
```

بایوس به شرطی سکتور 0 را می‌خواند و تحلیل می‌کند که بداند آن magic number AA55H است اگر نبود می‌نویسد این هارد سیستم عامل ندارد یا مشکل دارد.

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda3	29G	17G	12G	59%	/
/dev/sda10	9.5G	556M	8.5G	7%	/tmp
/dev/sda9	9.5G	3.6G	5.5G	40%	/var
/dev/sda7	24G	17G	5.9G	74%	/opt
/dev/sda6	29G	4.3G	23G	16%	/usr
/dev/sda8	9.5G	397M	8.6G	5%	/usr/local
/dev/sda5	29G	13G	15G	46%	/home
/dev/sda1	99M	17M	78M	18%	/boot
tmpfs	1009M	0	1009M	0%	/dev/shm
/dev/sda2	38G	29G	7.6G	79%	/var/ftp/pub



در اینجا می بینیم که پارتیشن ها بیش تر از 4 قطعه است (10) و تا هم می توانیم درست کنیم ، در واقع پارتیشن چهارم خودش به پارتیشن های دیگری ارجاع می دهد و اطلاعات مربوط به پارتیشن های منطقی را نگه می دارد.

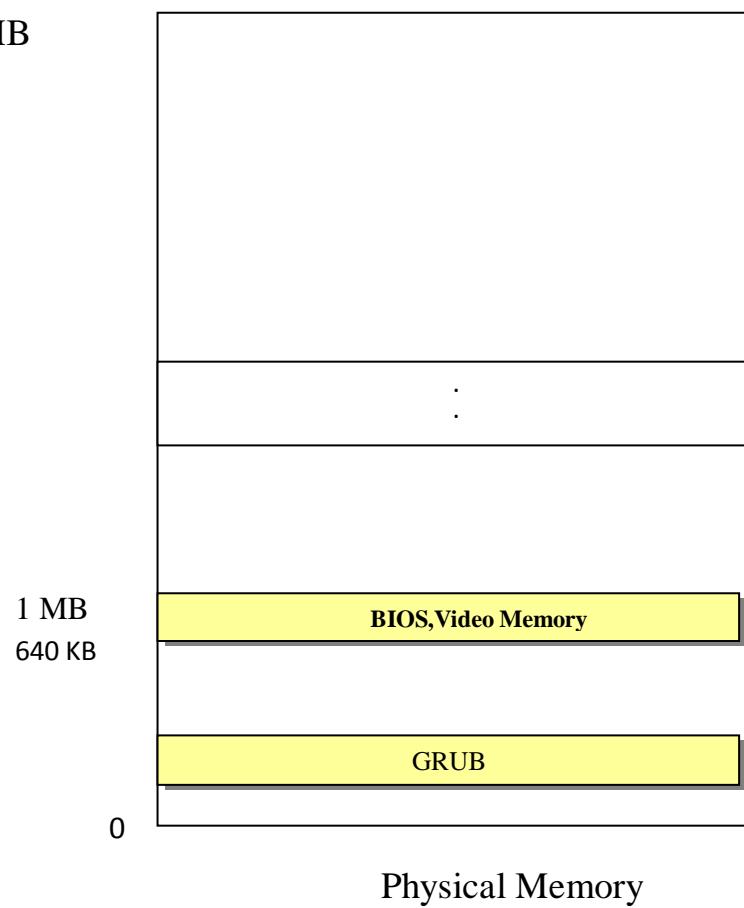
سوال چرا سکتور 0 را در اول حافظه نمی گذارد؟

چون 1024 بایت اول حافظه برای نگه داری بردارهای وقفه¹ مصرف می شود. سکتور صفر دیسک سخت خوانده شده و در آدرس 0x7C00 حافظه قرار گرفته و magic number آن کتترل می گردد که بايستی معادل 0xAA55 باشد.

¹ interrupt vectors

RAM Limit

e.g. 512MB



خیلی از جاها به این 446 بایت¹ MBR می‌گویند و بعضی از ویروس‌ها می‌خواهند ولی خیلی از بایوس‌ها هم اگر کسی بخواهد روی MBR بنویسد warning می‌دهند در واقع MBR مثل نگهبان دم در است اگر آلووده شود کل سیستم به خطر می‌افتد.

MBR پس از تست‌های بسیار مقدماتی سکتور صفر Active Partition را وارد حافظه می‌کند و در این مرحله از انتخاب سیستم عامل، لینوکس وارد حافظه می‌شود.

MBR یا Stage I، Stage II را وارد حافظه می‌کند. درست مثل وقتی که ریسیس جمهور قبل از سفر به یک کشور ابتدا نمایندگانی می‌فرستند تا بررسی‌ها و هماهنگی‌های لازم را انجام دهند در اینجا نیز سیستم عامل یکباره بالا نمی‌آید.

در دوره LPI 1 گفتیم که سیستم عامل یک نرم افزار نیست بلکه مجموعه‌ای (Σ) از نرم افزارهای است.

$$\text{operating system} = \sum m_i + \sum t_i$$

که m نشانه مازول و t به معنی جدول (table) است. به عنوان مثال یک مازول مدل cpu را چک می‌کند یکی دیگر با کارت شبکه صحبت می‌کند و هر کدام از این نرم افزارها یک جدول دارند در واقع سیستم عامل یعنی هیئت دولت که هر کدام یک کیف (جدول) دارند که اطلاعات مربوط به کارشان داخل آن است.

مازول cpu می‌داند که با چه مدل cpu کار می‌کند چقدر سرعت دارد چقدر کش دارد. در مثال وزیر خارجه‌ها با هم صحبت می‌کنند و پروتکل امضا می‌کنند یه همین ترتیب بقیه وزرا یارگیری می‌کنند.

کلی کار انجام می‌شود که نتیجه آن را با دستور زیر می‌بینیم:

```
[n.pardis@lpi ~] $ dmesg | less

Linux version 2.6.18-238.5.1.1.el5 (mockbuild@localhost) (gcc
version 4.1.2 2008 0704 (Red Hat 4.1.2-48)) #1 SMP Wed Mar 30
13:22:24 NOVST 2011
BIOS-provided physical RAM map:
BIOS-e820: 0000000000010000 - 000000000009f400 (usable)
BIOS-e820: 000000000009f400 - 00000000000a0000 (reserved)
BIOS-e820: 00000000000e0000 - 0000000000010000 (reserved)
BIOS-e820: 0000000000100000 - 0000000007f6b0000 (usable)
BIOS-e820: 0000000007f6b0000 - 0000000007f6be000 (ACPI data)
BIOS-e820: 0000000007f6be000 - 0000000007f6f0000 (ACPI NVS)
BIOS-e820: 0000000007f6f0000 - 0000000007f6fe000 (reserved)
BIOS-e820: 000000000fee0000 - 000000000fee01000 (reserved)
BIOS-e820: 000000000fff80000 - 00000000100000000 (reserved)
1142MB HIGHMEM available.
896MB LOWMEM available.
found SMP MP-table at 000ff780
Memory for crash kernel (0x0 to 0x0) notwithin permissible range
disabling kdump
```

¹ master boot record

گزارش اینکه لینوکس وارد شده دیده چقدر حافظه داریم، کجا استفاده شده و مدل Cpu چیه! به عنوان مثال با تایپ **cpu**/گزارش **cpu** را می بینیم. چند صفحه پایین تر:

```
CPU: Trace cache: 12K uops, L1 D cache: 16K
CPU: L2 cache: 2048K
CPU: Hyper-Threading is disabled
CPU: After all inits, caps: bfefbf3ff 20000000 00000000 00000180
0000e59d 00000000 0 00000001
Intel machine check architecture supported.
Intel machine check reporting enabled on CPU#0.
CPU0: Intel P4/Xeon Extended MCE MSRs (24) available
CPU0: Thermal monitoring enabled → گرمم شد چه کار می کنی؟
Checking 'hlt' instruction... OK. ← سکته کردم چی؟
SMP alternatives: switching to UP code
Freeing SMP alternatives: 14k freed
ACPI: Core revision 20060707
CPU0: Intel(R) Pentium(R) 4 CPU 3.20GHz stepping 05
Total of 1 processors activated (6438.12 BogoMIPS).
ENABLING IO-APIC IRQs
..TIMER: vector=0x31 apic1=0 pin1=2 apic2=-1 pin2=-1
Using local APIC timer interrupts.
Brought up 1 CPUs
sizeof(vma)=84 bytes
sizeof(page)=32 bytes
sizeof(inode)=340 bytes
sizeof(dentry)=136 bytes
```

device driver و device تا موقعی که سیستم بالاست زندگی مشترک دارند باید به درستی همدیگر را بشناسند تا بتونند بهتر سرویس دهند. مثلا درایور **cpu** اول **cpu** را **initialize** می کند. در آخر درایور می نویسد که با په وسیله ای دارد زندگی می کند: **intel**

لینوکس مثل رهبر جامعه است کار اجرایی نمی کند (مدیریت روی منابع می کند) بلکه یک معاون اجرایی برای خودش می آورد (معادل نخست وزیر). مثلا رئیس جمهور هندوستان را کسی نمی شناسد نخست وزیر کار های اجرایی را انجام میدهد. کسی **login** کند یا گزارشی را پرینت بگیرد یا فایل کپی کند سیستم عامل نمی فهمد

عمده نرم افزارهایی که سیستم را بالا می آورد اینجاست:

```
[n.pardis@lpi ~]$ cd /boot/grub/  
[n.pardis@lpi grub]$ ls  
total 233  
-rw-r--r-- 1 root root      63 May 11 2011 device.map  
-rw-r--r-- 1 root root  7584 May 11 2011 e2fs_stage1_5  
-rw-r--r-- 1 root root  7456 May 11 2011 fat_stage1_5  
-rw-r--r-- 1 root root  6720 May 11 2011 ffs_stage1_5  
-rwxr-xr-x 1 root root    779 Feb 14 2012 grub.conf  
-rw-r--r-- 1 root root  6720 May 11 2011 iso9660_stage1_5  
-rw-r--r-- 1 root root  8192 May 11 2011 jfs_stage1_5  
lrwxrwxrwx 1 root root      11 May 11 2011 menu.lst -> ./grub.conf  
-rw-r--r-- 1 root root  6880 May 11 2011 minix_stage1_5  
-rw-r--r-- 1 root root  9248 May 11 2011 reiserfs_stage1_5  
-rw-r--r-- 1 root root 32428 Jan  4 2007 splash.xpm.gz  
-rw-r--r-- 1 root root   512 May 11 2011 stage1  
-rw-r--r-- 1 root root 104988 May 11 2011 stage2  
-rw-r--r-- 1 root root  7072 May 11 2011 ufs2_stage1_5  
-rw-r--r-- 1 root root  6272 May 11 2011 vstafs_stage1_5  
-rw-r--r-- 1 root root  8904 May 11 2011 xfs_stage1_5
```

دستور strings به کار ادمین های لینوکس می آید. فایل های باینری را می خواند اگر چیزی قابل نمایش بود نشان می دهد. بیشتر برای فایل های executable و با data base مناسب است و به درد فایل های txt نمی خورد.

```
[n.pardis@lpi grub]$ strings stage1  
ZRRI  
D|f1  
GRUB  
Geom  
Hard Disk  
Read  
Error  
Floppy
```

یک read error بیش تر ندارد.

کار متخصص کودکان از بقیه همکارانش سخت تر است چون نوزاد نمی تواند صحبت کند و راجع به بیماریش به او اطلاعات بدهد خروجی نوزاد حداکثر گریه است! به همین ترتیب اگر سیستم در stage 1 ، crash کند کار سخت تر است چون فقط می گوید 1.

باید شماره خطاهای را بدانید البته بازهم کار مشکل است.

```
[n.pardis@lpi grub]$ strings -15 stage2|less

/grub/grub.conf
[Linux-initrd @ 0x%x, 0x%x bytes]
[Multiboot-module @ 0x%x, 0x%x bytes]
linux 'zImage' kernel too big, try 'make bzImage'
[Linux-%s, setup=0x%x, size=0x%x]
, loadaddr=0x%x, text%s=0x%x
, <0x%x:0x%:0x>
Address 0x%x: Value 0x%x
Filesystem tracing is now off
Filesystem tracing is now on
APM BIOS information:
Version:          0x%x
32-bit CS:        0x%x
Offset:           0x%x
16-bit CS:        0x%x
16-bit DS:        0x%x
32-bit CS length: 0x%x
16-bit CS length: 0x%x
16-bit DS length: 0x%x
```

يعنى اگر 15 تا حرف دیدی که قابل نمایش است نمایش بده. به نسبت stage I اطلاعات مفصل تر است.

اگر در این مرحله سیستم به مشکل برخورد کند مثل آنکه مریض، کودک 8-9 ساله باشد کار راحت تر است؛ پیغام ها و اطلاعات به نسبت زیاد است. در لیست stage1_5 هم داریم که به ندرت به آنها برخورد می کنیم مگر اینکه لینوکس را روی فایل سیستم ذاتی^۱ خودش نصب نکرده باشیم مثلاً پارتیشن fat_stage1_5 بوده:

این امر به هر دلیلی که خیلی هم قانع کننده نیست ممکن است اتفاق بیفتد مثلاً دیسک را با ویندوز فرمت کردیم حالا به زور می خواهیم لینوکس نصب کنیم مثل این است که چون من عربی بلدم lpi را به زبان عربی درس بدhem!

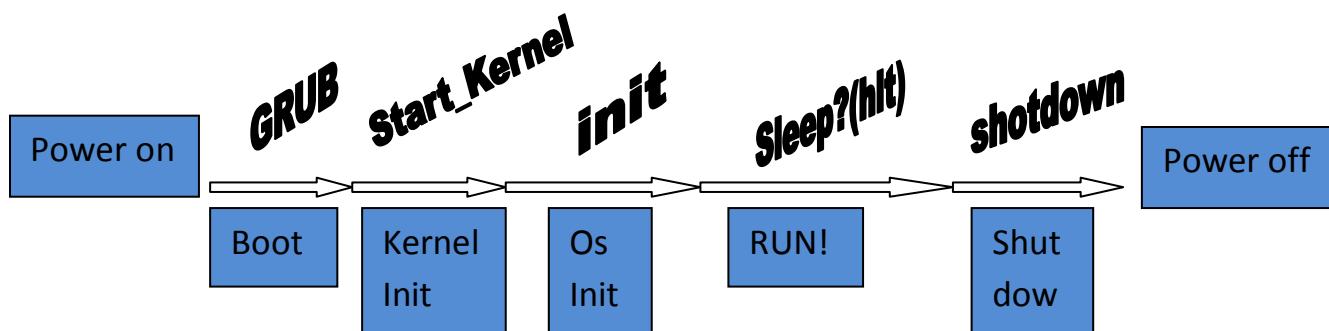
نرم افزار کوچکی است که مثل stage1 و stage2 را مهیا می کند و کارش که تمام شد در memory نمی ماند که بگوییم ویندوز و لینوکس را روی fat کنار هم نصب می کنیم که به هم دسترسی داشته باشند! به علاوه در کرنل نرم افزارهای بسیاری هست که ویندوز را می شناسند که بعداً با آنها آشنا می شویم.

فایل سیستم ذاتی لینوکس الان^۱ ext3 و ext4 است. با اینکه minix پایه لینوکس است و توروالدز لینوکس را از روی آن نوشته یا ufs^۲ و xfs فایل سیستم های لینوکسی هستند ولی اینها native default لینوکس روی pc نیستند. اکثر فایل سیستم ها ext3 هستند:

¹ Native

```
[n.pardis@lpi grub]$ mount
/dev/sda3 on / type ext3 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda10 on /tmp type ext3 (rw)
/dev/sda9 on /var type ext3 (rw)
/dev/sda7 on /opt type ext3 (rw)
/dev/sda6 on /usr type ext3 (rw)
/dev/sda8 on /usr/local type ext3 (rw)
/dev/sda5 on /home type ext3 (rw)
/dev/sda1 on /boot type ext3 (rw)
tmpfs on /dev/shm type tmpfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
/dev/sda2 on /var/ftp/pub type ext3 (ro)
```

اولین نرم افزاری است که وارد سیستم می شود و همکاره است (نخست وزیر است). اصطلاحا number one پروسس ها می باشد که فعالیت های لازمه را که توضیح آنها در ادامه می آید را انجام داده و به خواب عمیقی فرو می رود؛ در سیستم های multiuser اگر نرم افزاری کاری ندارد باید بخوابد چون cpu می گیرد.



```
[n.pardis@lpi grub]$ cd /boot
[n.pardis@lpi boot]$ ls
config-2.6.18-194.el5           symvers-2.6.18-194.el5.gz
config-2.6.18-238.5.1.1.el5      symvers-2.6.18-
238.5.1.1.el5.gz
grub                           System.map-2.6.18-194.el5
initrd-2.6.18-194.el5.img       System.map-2.6.18-
238.5.1.1.el5
initrd-2.6.18-238.5.1.1.el5.img vmlinuz-2.6.18-194.el5
lost+found                       vmlinuz-2.6.18-238.5.1.1.el5
memtest86+-1.65
```

¹ extension

² Unix File System

اسم کرنل ما vmlinuz-2.6.18¹ است ولی ما در کلاس دو تا لینومس داریم که هر کدام برای کاری استفاده می شود.

برای اینکه ببینیم با چه کرنلی بالا آمده ایم:

```
[n.pardis@lpi boot]$ uname -a
Linux lpi.aictc.edu 2.6.18-238.5.1.1.el5 #1 SMP Wed Mar 30 13:22:24 NOVST 2011
i686 i686 i386 GNU/Linux
```

روی سیستم می توان لینوکس های متعدد نصب کرد و روی هر لینوکس می توانیم چند کرنل داشته باشیم. اگر دقت کنید فایل های موجود در زیر دایرکتوری /boot مشترکاتی دارن دمثلا برای هر ورژن initrd و system.map معادل stage را داریم. وقتی کرنل را رد می کند و بالا می آید یک سری فایل هایی درست می کند که یکی از آنها system.map است.

```
[n.pardis@lpi boot]$ less System.map-2.6.18-194.el5
```

```
00000400 A kernel_vsyscall
00000410 A SYSENTER_RETURN
00000420 A kernel_sigreturn
00000440 A kernel_rt_sigreturn
00400000 A phys_startup_32
c0400000 T _text
c0400000 T startup_32
c0401000 T startup_32_smp
c0401080 t checkCPUtype
c0401101 t is486
c0401108 t is386
c040116a t check_x87
c0401192 t setup_idt
c04011af t rp_sidt
c04011bc t ignore_int
c04011f0 T _stext
c04011f0 t run_init_process
c04011f0 T stext
c040122c t init_post
c04012e7 t rest_init
c0401308 t try_name
c0401485 T name_to_dev_t
c04016cc T calibrate_delay
System.map-2.6.18-194.el5
```

¹ virtual machine linux

اینها m_i ها هستند(اکثر) و نشان می دهد که روتین های کرنل در چه آدرسی نشسته اند. به عنوان مثال is386 چک می کند که i386 cpu است یا نه. وقتی لینوکس را نصب می کنیم با هسته اصلی که vmlinuz است یک system table هم می آید که نشان می دهد چه نرم افزارهایی با هم جمع شده اند تا کرنل درست شده است. اگر یک فایل t درست شد از روی این می فهمیم که کدام روتین fault را به وجود آورده است.

```
[n.pardis@lpi boot]$ ls initrd-2.6.18-194.el5.img
-rw-r--r-- 1 root root 2556825 May 11 2011 initrd-2.6.18-
194.el5.img
[n.pardis@lpi boot]$ file initrd-2.6.18-194.el5.img
initrd-2.6.18-194.el5.img: gzip compressed data, from Unix, last
modified: Wed May 11 20:49:35 2011, max compression
[n.pardis@lpi boot]$ cd
[n.pardis@lpi ~]$ mkdir initrd
[n.pardis@lpi ~]$ cd initrd/
[n.pardis@lpi initrd]$ cp /boot/initrd-2.6.18-194.el5.img .
[n.pardis@lpi initrd]$ ls
total 2504
-rw-r--r-- 1 n.pardis lpi1 2556825 Aug 24 12:14 initrd-2.6.18-
194.el5.img
[n.pardis@lpi initrd]$ mv initrd-2.6.18-194.el5.img initrd.gz
[n.pardis@lpi initrd]$ gzip -d initrd.gz
[n.pardis@lpi initrd]$ file initrd
initrd: ASCII cpio archive (SVR4 with no CRC)
```

یک سری warning می دهد که جای نگرانی ندارد

```
[n.pardis@lpi initrd]$ cpio -i < inirdtrd
cpio: dev/ttys0: Operation not permitted
cpio: dev/ptmx: Operation not permitted
cpio: dev/ram1: Operation not permitted
cpio: dev/tty4: Operation not permitted
.....
```

حالا با دستور 1 چارت سازمانی را می بینیم که برای کسانی که lpi1 را گذرانده اند آشناست.

```
[n.pardis@lpi initrd]$ ls
total 5728
drwx----- 2 n.pardis lpi1 4096 Aug 24 12:16 bin
drwx----- 3 n.pardis lpi1 4096 Aug 24 12:16 dev
drwx----- 2 n.pardis lpi1 4096 Aug 24 12:16 etc
-rwx----- 1 n.pardis lpi1 2287 Aug 24 12:16 init
-rw-r--r-- 1 n.pardis lpi1 5819904 Aug 24 12:14 initrd
drwx----- 3 n.pardis lpi1 4096 Aug 24 12:16 lib
drwx----- 2 n.pardis lpi1 4096 Aug 24 12:16 proc
lrwxrwxrwx 1 n.pardis lpi1 3 Aug 24 12:16 sbin -> bin
drwx----- 2 n.pardis lpi1 4096 Aug 24 12:16 sys
drwx----- 2 n.pardis lpi1 4096 Aug 24 12:16 sysroot
```

در سیستم عامل واقعی هم این دایرکتوری ها را داشتیم ولی مثلا زیر etc هیچ فایلی نداریم. در واقع initrd یک سیستم عامل کوچک است که کارهایی انجام می دهد تا سیستم عامل اصلی را وارد سیستم کند. با دستور recursive زیر همه زیردایرکتوری ها را می بینیم ، به جای اینکه تک تک 1 بزنیم.

```
[n.pardis@lpi initrd]$ lpg -R|less
./lib:
total 992
-rw----- 1 n.pardis lpi1 30644 Aug 24 12:16 ata_piix.ko
-rw----- 1 n.pardis lpi1 17908 Aug 24 12:16 dm-log.ko
-rw----- 1 n.pardis lpi1 11188 Aug 24 12:16 dm-mem-cache.ko
-rw----- 1 n.pardis lpi1 8808 Aug 24 12:16 dm-message.ko
-rw----- 1 n.pardis lpi1 75644 Aug 24 12:16 dm-mod.ko
-rw----- 1 n.pardis lpi1 74148 Aug 24 12:16 dm-raid45.ko
-rw----- 1 n.pardis lpi1 18512 Aug 24 12:16 dm-region_hash.ko
-rw----- 1 n.pardis lpi1 38932 Aug 24 12:16 ehci-hcd.ko
-rw----- 1 n.pardis lpi1 147376 Aug 24 12:16 ext3.ko
drwx----- 2 n.pardis lpi1 4096 Aug 24 12:16 firmware
-rw----- 1 n.pardis lpi1 72660 Aug 24 12:16 jbd.ko
-rw----- 1 n.pardis lpi1 189932 Aug 24 12:16 libata.ko
```

است یعنی روتین سیستم عاملی کرنل را باید config کرد ؛ کلی سوال از ما میپرسد که مثلا tape داری؟ دیسک مدلش چیه؟ کارت شبکه داری؟ این تمام سوال و جواب هایی است که موقعی که کرنل را ساختیم همه در این فایل قرار می گیرند:

```
[n.pardis@lpi boot]$ less config-2.6.18-194.el5
```

```
#  
# Automatically generated make config: don't edit  
# Linux kernel version: 2.6.18-194.el5  
# Tue Mar 16 21:51:38 2010  
#  
CONFIG_X86_32=y → آیا سیستم x86 32 بیتی است؟ جواب  
CONFIG_GENERIC_TIME=y  
CONFIG_LOCKDEP_SUPPORT=y  
CONFIG_STACKTRACE_SUPPORT=y  
CONFIG_SEMAPHORE_SLEEPERS=y  
CONFIG_X86=y  
CONFIG_MMU=y → آیا memory management دارد؟ بله  
CONFIG_GENERIC_ISA_DMA=y  
CONFIG_GENERIC_IOMAP=y  
CONFIG_GENERIC_HWEIGHT=y  
CONFIG_ARCH_MAY_HAVE_PC_FDC=y  
CONFIG_DMI=y  
CONFIG_DEFCONFIG_LIST="/lib/modules/$UNAME_RELEASE/.config"  
  
#  
# Code maturity level options  
#  
CONFIG_EXPERIMENTAL=y  
config-2.6.18-194.el5
```

اینها زمانی استفاده می شود که خودتان تصمیم گرفتید کرنل را کامپایل کنید و همان موقع این فایل ساخته می شود نه این که هر دفعه با مازول هایی که گفتیم load شود. اگر لینوکس را نصب کرده اید از سایت kernel.org یک فایل 60MB دانلود می کنید و کرنل را از ابتدا نصب می کنید؛ همان موقع این فایل ساخته می شود. مورد استفاده آن هم برای زمانی است که به مشکل برخوردیم که بدانیم به سوالات چه جوابهایی داده ایم.

چیز اضافه ای است که ربطی به لینوکس ندارد در واقع این یک بچه کرنل است که فقط memory را تست می کند:

```
[n.pardis@lpi boot]$ file memtest86+-1.65
memtest86+-1.65: Linux x86 kernel
```

دستور file با استفاده از magic number این اطلاعات را می فهمد.

```
[n.pardis@lpi boot]$ cd grub/
[n.pardis@lpi grub]$ ls
device.map      grub.conf          minix_stage1_5    stage2
e2fs_stage1_5   iso9660_stage1_5  reiserfs_stage1_5  ufs2_stage1_5
fat_stage1_5    jfs_stage1_5      splash.xpm.gz    vstafs_stage1_5
ffs_stage1_5    menu.lst         stage1           xfs_stage1_5
[n.pardis@lpi grub]$ file stage1
stage1: x86 boot sector, code offset 0x48
```

سیستم عامل می تواند قدیمی باشد ولی نرم افزارها به روز باشند.

```
[n.pardis@lpi grub]$ file --version
file-4.17
```

تعدادی error code هست که اگر سیستم بالا نیاید grub آنها را اعلام می کند مثلا 7 error stage1 ، stage1_5 ، stage2 و stage2_5 در این فایل قرار دارند. در stage1 مثلا فقط Hard Disk Error چاپ می شود و اطلاعات بیشتری نمی دهد.

stage2 با هوش تر است و پیغام هایش واضح تر. بهتر است محتویات این فایل را پرینت بگیرید و در دسترس داشته باشید چون اگر سیستم با مشکل مواجه شود به این راهنمای دسترسی نداریم. خطای bad file or directory یا no such directory مربوط به خرابی سخت افزار است یا اینکه سیستم را با ویندوز بالا آورده ایم، فایل ها را خراب کرده یا ویروسی شده که به در حالت عادی نباید این پیغام ها را مشاهده کنیم. تا اینجا کرنل بالا آمده و سیستم عامل کار دیگری ندارد بلکه فقط از بالا بر منابع مدیریت می کند init به عنوان مقام چیدمان را انجام می دهد مثلا وب سرور را بالا می آورد، میل سرور را بالا می آورد و...

حال سوال این است که init از کجا می داند وب سرور کجاست؟ مدیرها روی میزشان کارتاپل دارند که مثلا ساعت 10 جلسه دارند یا ساعت 2 سخنرانی دارند.پروسس init با شماره 1 وارد سیستم می گردد و برای اجرای بقیه نرم افزارها به فایل /etc/inittab مراجعه می نماید.فایل /etc/inittab یکی از مهم ترین فایل های سیستم عامل لینوکس است.در نسخه های جدید fedora و ubuntu فایل inittab تغییر پیدا کرده است ولی روی بقیه نسخه ها تغییر نکرده است.معمولًا fedora پیش مرگ redhat است و اوبونتو پیش مرگ debian ! تغییرات اول در اوبونتو و fedora اعمال می شود در سایت های مختلف بررسی می شود مشکلاتش پیدا می شود بعد در نسخه های بعدی ردت و دبیان اعمال ubuntu می شود.ولی چون امتحان بین المللی LPI هنوز بر آن مبنای نیست نسخه ای که ما تدریس می کنیم در 14 fedora core و 10 ubuntu کمی فرق می کند با این وجود کلیات یکسان است.

یک فایل اساسی بوت است که اگر نباشد init نمی داند وب سرور را بالا بیاورد یا نه؟ این فایل پرونده اصلی سیستم است . در خود فایل نوشته:

This file describes how the INIT process should set up.

بهترین ترجمه برای set up برپاسازی است.در این فایل مثل دیگر فایل های پیکربندی اگر خطی با # یا & شروع شود comment است.
اگر init چاک شود init بالا می آید و پیغام می دهد که چه کار کنم؟ نه crash می کند نه hang !
init فایل init چیزی را خط به خط می خواند.در سیستم های عملیاتی و جدی مجبور یید در این فایل تغییراتی ایجاد کنید ولی هر چیزی را که تغییر دادید در comment آن بنویسید که در چه زمانی و به چه دلیلی تغییر یافته است.

سوال های امتحانی: runlevel چیست؟ grub چیست؟ init چیست و چگونه active می شود؟

runlevel مفهوم

فرض کنید اعضای خانواده ساعت 4 صبح پنیر می خواهند. برای خریدن پنیر به سوپرمارکت می رویم ولی چون تعطیل است مدتی جلوی سوپرمارکت قدم می زیم. پس از مدتی کارگری می آید و درب سوپرمارکت را باز می کند ولی هنوز نمی توانیم داخل برویم چون کارگر مشغول نظافت مغازه می شود. پس از مدتی می بینیم که بقیه مشتری ها برای خرید وارد مغازه می شوند ما هم با آنها وارد می شویم و از مغازه دار پنیر می خواهیم ولی می گویید من پنیر ندارم اما مدتی صبر کن تا از مغازه بغل برایت پنیر بیاورم ..سیستم عامل لینوکس به مجرد بالا آمدن در یکی از حالات زیر قرار خواهد گرفت:

Shutdown	runlevel 0	تعطیل
Single User Mode	runlevel 1	یک نفر
Multi User Mode	runlevel 2	خرید
Full Multi User	runlevel 3	مغازه بغل
Unused	runlevel 4	
Graphical Mode	runlevel 5	
Reboot	runlevel 6	

```
[n.pardis@lpi ~]$ less /etc/inittab

#
# inittab      This file describes how the INIT process should
set up
#           the system in a certain run-level.
#
# Author:      Miquel van Smoorenburg,
<miquels@drinkel.nl.mugnet.org>
#           Modified for RHS Linux by Marc Ewing and Donnie
Barnes
#

# Default runlevel. The runlevels used by RHS are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3, if you do not have
networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:
```

اگر سیستم خاموش یاشد در runlevel 0 قرار داریم درست مثل موقعی که سوپرمارکت بسته باشد بعضی موقعیت سیستم مشکل دارد یا داریم نرم افزارهای را کم یا زیاد می کنیم که نباید در این موقعیت login کند؛ سیستم را با single user بالا می آوریم.(یک نفر نظافت می کند!)

در مثال موقعی است که مشتری ها برای خرید داخل می شوند.

(آوردن پنیر از مغازه بغل) : مثلا کامپیوتر کلاس جاوا ندارد ولی کامپیوتر اتاق بغل دارد و با ما share است ؛ برنامه روی آن اجرا می شود.

سوال: چرا در inittab نوشته که از RL-0 و RL-6 به عنوان پیش فرض init استفاده نکنید؟

خیلی موقع در انجمن ها مطرح می کنند که سیستم را روشن می کنیم ولی بلافضله خاموش می شود ، دلیلش این است که خط آخر به صورت **id:6:initdefault** یا **id:0:initdefault** تعريف کرده است.

جهت اطلاع کسانی که امتحان بین المللی lpi دارند: بحث runlevel خیلی مهم است و سوالات زیادی راجع به آن مطرح می شود چون اگر شما Admin یک سایت جدی باشید خیلی وقت ها مجبورید بین switch ها runlevel کنید.

اگر لینوکس را روی sun نصب کنید RL-4 معنی پیدا می کند و گرنه روی intel base pc معنی ندارد.

RL-5 محیط گرافیکی است. در لینوکس یعنی گرافیکی:

```
[n.pardis@lpi ~]$ man X  
X(7)  
X(7)  
  
NAME  
      X - a portable, network-transparent window system  
  
SYNOPSIS  
      The X Window System is a network transparent window system which  
runs  
      on a wide range of computing and graphics machines. It should be  
rela-  
      tively straightforward to build the X.Org Foundation software  
distribu-  
      tion on most ANSI C and POSIX compliant systems. Commercial  
implemen-  
      tations are also available for a wide range of platforms.  
  
      The X.Org Foundation requests that the following names be used  
when  
      referring to this software:  
  
      X  
      X Window System  
      X Version 11  
      X Window System, Version 11  
      X11
```

X Window System is a trademark of The Open Group.

سه دستور برای اینکه بفهمیم سیستم کلاس با چه runlevel بی بالا آمده است:

```
[n.pardis@lpi ~]$ runlevel → 1  
N 3  
[n.pardis@lpi ~]$ who -r → 2  
run-level 3 2012-08-04 13:55  
[n.pardis@lpi ~]$ ps -aef |grep init → 3  
root 1 0 0 Aug04 ? 00:00:01 init [3]  
n.pardis 32479 32230 0 15:35 pts/3 00:00:00 grep init
```

```
[n.pardis@lpi ~]$ ps -aef |less
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	Aug04	?	00:00:01	init [3]
root	2	1	0	Aug04	?	00:00:00	[migration/0]
root	3	1	0	Aug04	?	00:00:00	[ksoftirqd/0]
root	4	1	0	Aug04	?	00:00:00	[watchdog/0]
root	5	1	0	Aug04	?	00:00:00	[events/0]
root	6	1	0	Aug04	?	00:00:00	[khelper]
root	7	1	0	Aug04	?	00:00:00	[kthread]
root	10	7	0	Aug04	?	00:00:01	[kblockd/0]
root	11	7	0	Aug04	?	00:00:00	[kacpid]
root	112	7	0	Aug04	?	00:00:00	[cqueue/0]
root	115	7	0	Aug04	?	00:00:00	[khubd]
root	117	7	0	Aug04	?	00:00:00	[kseriod]
root	179	7	0	Aug04	?	00:00:00	[khungtaskd]
root	182	7	0	Aug04	?	00:00:34	[kswapd0]
root	183	7	0	Aug04	?	00:00:00	[aio/0]
root	339	7	0	Aug04	?	00:00:00	[kpsmoused]
root	362	7	0	Aug04	?	00:00:00	[ata/0]
root	363	7	0	Aug04	?	00:00:00	[ata_aux]
root	366	7	0	Aug04	?	00:00:00	[scsi_eh_0]
root	367	7	0	Aug04	?	00:00:00	[scsi_eh_1]
root	372	7	0	Aug04	?	00:00:00	[kstriped]

دستور ps وضعیت سیستم را نمایش می دهد و pid مخفف parent process id و ppid process id است. pid کرنل صفر است و حضور ندارد ، وجود ندارد pid مربوط به init 1 است و پدر خیلی از نرم افزارهاست. حتی پدر 7 است که 7 پدر نرم افزارهای زیادی است.

سوال امتحانی: پروسس number one کدام پروسس است؟ runlevel 3 چه؟

در فایل inittab جز خط اول که default runlevel را تعیین می کند در بقیه خط ها قانون مندی وجود دارد و خیلی وقت ها در سیستم های عملیاتی باید این پارامترها را تغییر دهید. فایل inittab شهرداری منطقه 4، 500 خط است در حالی که در سیستم کلاس ما 20 خط است.

id:runlevels:action:process

فورمات فایل inittab :

1:2345:respawn:/sbin/minetty ttym1

label : این فیلد شناسه نام داشته و حداکثر 4 حرفی می باشد. البته در بعضی نسخه های لینوکس و یونیکس تا 8 حرف جا دارد. id ، است کاربردش در goto نیست (در debuging trouble shouting و trouble) بعضی از آنها عددی است، بعضی x و بعضی pr یا pf .

runlevel می تواند مجموعه ای از مقادیر بین 0 الی 6 باشد. یعنی نرم افزار در چه runlevel بگذاریم (default) در همه runlevel ها کار می کند.

تمرین: دستگاه رادار را به سیستم وصل کنید که همیشه کار کند در ضمن نام نرم افزار /HOME/R/RADAR است!

الان که login هستید اگر دستوری را اجرا کنید انجام می شود ولی اگر logout کنید آن دستور در حافظه از بین می رود (kill می شود) اگر وسط کار kill شد دوباره باید بزنید؛ چرا این کار را بکنیم نرم افزاری که همیشه در حافظه است و سرپرست اصلی است آن را اجرا می کند. عبارت: inittab را در rdr:: می نویسیم (با دستور vi)

تعدادی از Action ها

respawn همواره یک کپی از پروسس در حافظه باشد.

off پروسس هیچ گاه اجرا نگردد.

wait init منتظر پایان این سرویس گردد

once پروسس فقط یک بار اجرا گردد

sysinit پروسس در ابتدای بالا آمدن سیستم اجرا گردد

initdefault runlevel پیش فرض

spin یعنی فرفره و spawn یعنی این برنامه همیشه در حافظه می ماند و تا نابود شد یک کپی دیگر از آن ساخته می شود.

Process: برنامه ای که تحت این runlevel اجرا خواهد شد.

```

pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"
# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"

# Run gettys in standard runlevels
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
rdr:respawn:/home/r/radar
#4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
#9:2345:respawn:/sbin/mingetty tty9

# Run xdm in runlevel 5
x:5:respawn:/etc/X11/prefdm -nodaemon

```

اگر این فایل را save کنیم سیستم که بالا آمد init را می خواند به این خط می رسد و می فهمد همیشه باید این را در حافظه داشته باشیم و اگر kill شد دوباره قرار دهیم.

رادار سوخت؛ فعلاً اجرا نکنید: off

هر روز می خواهیم backup بگیریم (یک بار): once در ویندوز هم اینها هست البته به صورت گرافیکی!

لیست کامل Action‌ها در manual فایل inittab هست:

```

[n.pardis@lpi ~]$ man inittab
...
Valid actions for the action field are:

respawn
    The process will be restarted whenever it terminates (e.g.
    getty).

wait    The process will be started once when the specified runlevel is
            entered and init will wait for its termination.

once   The process will be executed once when the specified runlevel is
            entered.

boot   The process will be executed during system boot. The runlevels
            field is ignored.

bootwait
    The process will be executed during system boot, while init
    waits for its termination (e.g. /etc/rc). The runlevels field
    is ignored.

off    This does nothing.

...

```

توصیه: بهترین منبع آموزشی همین manual‌های لینوکس است البته مثال‌های کمی دارد و از انگلیسی روان استفاده نکرده است. جزو و اسلامیدهای مورد استفاده در این کلاس از این manual‌ها استخراج شده است با این وجود از لحاظ جامعیت قابل مقایسه با حجم عظیم manual‌های لینوکس نیست.

```
[n.pardis@lpi ~]$ psa |grep tty5
n.pardis 2487 2397 0 19:20 pts/3    00:00:00 grep tty5
root    1908   1  0 16:11          tty5    00:00:00 /sbin/mingetty tty5
[n.pardis@lpi ~]$ kill 1908
[n.pardis@lpi ~]$ psa |grep tty5
n.pardis 2487 2397 0 19:21 pts/3    00:00:00 grep tty5
root    32356   1  0 19:21          tty5    00:00:00 /sbin/mingetty tty5
[n.pardis@lpi ~]$ kill 32356
[n.pardis@lpi ~]$ psa |grep tty5
n.pardis 2487 2397 0 19:21 pts/3    00:00:00 grep tty5
root    32356   1  0 19:21          tty5    00:00:00 /sbin/mingetty tty5
```

پدر respawn است و آن action init ، mingetty است در نتیجه اولا در زمان up شدن سیستم بالا آمده است(16:11) و ثانیا هر وقت آن را kill کنیم دوباره ساخته می شود. init به سیستم عامل اعلام می کند که هر موقع سیگنال 17 به من رسید (death of child) خبرم کن سپس دوباره فایل inittab را خوانده فرزند kill شده را دوباره می سازد.

اگر در عرض 5 ثانیه 7 بار یک نرم افزار فرزند init را kill کنید init در کنسول می نویسد فایده ای ندارد و لآن دوباره آن را respawn نمی کنم و چند دقیقه دقیقه دیگر respawn می کنم.. در غیر این صورت همه وقت init و cpu صرف نرم افزارها می شود.

بخشی از فایل inittab و در نتیجه init به مجرد اجرا شدن فایل rc.sysinit را اجرا خواهد کرد که شامل نرم افزارهای مربوط به کنترل و آماده سازی سیستم است.

```
id:3:initdefault:
# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit
```

```
[n.pardis@lpi ~]$ less /etc/rc.d/rc.sysinit

#!/bin/bash
#
# /etc/rc.d/rc.sysinit - run once at boot time
#
# Taken in part from Miquel van Smoorenburg's bcheckrc.
#
HOSTNAME=`/bin/hostname`
HOSTTYPE=`uname -m`
unamer=`uname -r`

set -m

if [ -f /etc/sysconfig/network ]; then
    . /etc/sysconfig/network
fi
if [ -z "$HOSTNAME" -o "$HOSTNAME" = "(none)" ]; then
    HOSTNAME=localhost
fi

if [ ! -e /proc/mounts ]; then
    mount -n -t proc /proc /proc
    mount -n -t sysfs /sys /sys >/dev/null 2>&1
/etc/rc.d/rc.sysinit
```

بالا آمدن لینوکس کلا با shell script است و به کسانی که می خواهند در این مسیر کار کنند توصیه می شود که را به خوبی آموزش ببینند.

تفاوتوی که بین ویندوز و لینوکس هست در همین چیزهاست ؛ ویندوز که بالا می آید فقط صفحه روشن و خاموش می شود و موس تکان می خورد در حالی که لینوکس در shell script اجرا می شود (چند صد خط پیغام!)

اول اسم کامپیوتر را پیدا می کند :

```
[n.pardis@lpi ~]$ hostname
lpi.aictc.edu
```

بعد سخت افزارها را پیدا می کند:

```
[n.pardis@lpi ~]$ uname -an
Linux lpi.aictc.edu 2.6.18-238.5.1.1.el5 #1 SMP Wed Mar 30 13:22:24
NOVST 2011 i686 i686 i386 GNU/Linux
```

بعد چک می کند که شبکه داریم ، اگر نداشتیم اسم کامپیوتر را localhost می گذاردو به همین ترتیب مقدار زیادی warning و

پیغام می دهد:

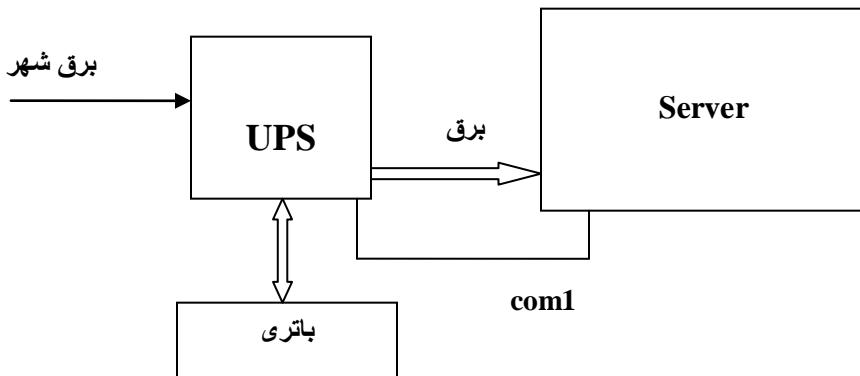
```
[n.pardis@lpi ~]$ less /etc/rc.d/rc.sysinit |wc -l
980 → comment با

[n.pardis@lpi ~]$ less /etc/rc.d/rc.sysinit |wc -l |wc -l |grep -v "#"
871 → comment بدون
```

rc در لینوکس مخفف دو چیز است : run command یا همان release candidate که در اینجا منظور همین است و نسخه های آزمایشی نرم افزارها که به منظور عیب یابی منتشر می شوند اطلاق می شود. در بخشی از فایل inittab همانگونه که ملاحظه می گردد در صورت فشار دادن کلیدهای shutdownctl+alt+del تحت هر runlevel، فرمان shutdown اجرا خواهد شد.

```
# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
#ca::ctrlaltdel:/bin/date
```

البته چون اکثرا به ویندوز عادت کرده اند ممکن است به اشتباه سرور را خاموش کنند! می توانیم اسکریپتی بنویسیم و در قسمت پروسس کد بالا به جای فرمان shutdown مسیر اسکریپتمان را می دهیم که موقع فشار دادن کلیدهای ctl+alt+del پیغامی بیاید و بپرسد Are you sure? در قرار دارد.



اگر روی سیستم UPS های online نصب باشد اصلاً مهم نیست که برق قطع شود یا نوسان داشته باشد. اگر برق شهر قطع شود از طریق باطری برق می دهد و سیستم اصلاحی فهمد که برق قطع شده است ولی به سیستم خبر می دهد که وضع برق خوب نیست. باطری که ضعیف شد با سیگنال SIGPWR (30) به سیستم اطلاع می دهد.

```
[n.pardis@lpi ~]$ kill -1
1) SIGHUP 2) SIGINT 3) SIGQUIT 4) SIGILL 5) SIGTRAP
6) SIGABRT 7) SIGBUS 8) SIGFPE 9) SIGKILL10) SIGUSR1
11) SIGSEGV12) SIGUSR213) SIGPIPE14) SIGALRM15) SIGTERM
16) SIGSTKFLT17) SIGCHLD18) SIGCONT19) SIGSTOP20) SIGTSTP
21) SIGTTIN22) SIGTTOU23) SIGURG24) SIGXCPU25) SIGXFSZ
26) SIGVTALRM27) SIGPROF28) SIGWINCH29) SIGIO 30) SIGPWR
31) SIGSYS34) SIGRTMIN35) SIGRTMIN+136) SIGRTMIN+237) SIGRTMIN+3
38) SIGRTMIN+439) SIGRTMIN+540) SIGRTMIN+641) SIGRTMIN+742) SIGRTMIN+8
43) SIGRTMIN+944) SIGRTMIN+1045) SIGRTMIN+1146) SIGRTMIN+1247) SIGRTMIN+13
48) SIGRTMIN+1449) SIGRTMIN+1550) SIGRTMAX-1451) SIGRTMAX-1352) SIGRTMAX-12
53) SIGRTMAX-1154) SIGRTMAX-1055) SIGRTMAX-956) SIGRTMAX-857) SIGRTMAX-7
58) SIGRTMAX-659) SIGRTMAX-560) SIGRTMAX-461) SIGRTMAX-362) SIGRTMAX-2
63) SIGRTMAX-164) SIGRTMAX
```

```
# When our UPS tells us power has failed, assume we have a few minutes
# of power left. Schedule a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf:::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"

# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"
```

سیستم سیگنال را به init می دهد و init خط powerfail را در inittab اجرا می کند یعنی 2 دقیقه (+2) بعد پیغام

Power Failure; System Shutting Down را برای همه می فرستد.

شرکتی بوده که ups با تضمین 10 دقیقه کار بعد از قطع برق خریده ولی بعد از دو دقیقه سیستم خاموش شده است. چون +2 را در inittab بدون تغییر گذاشته بودند. اینها را در ویندوز هم داریم که Hibernate می کند ولی نمی دانیم چطور؟ اگر در لحظه آخر برق وصل شود دو خط بعد اجرا می شود (powerokwait)؛ -c یعنی اینکه دستور shutdown را cancle می کند.

بخشی از فایل inittab که در 5 runlevel فرمان prefmd را اجرا و کنسول دارای محیط گرافیکی خواهد شد.

```
# Run xdm in runlevel 5
x:5:respawn:/etc/X11/prefdm -nodaemon
```

prefare display manager : prefmd

در لینوکس display manager های مختلف داریم مثل GNOME یا KDE در فایل زیر می توانیم محیط گرافیکی را که ترجیح می دهیم عوض کنیم. همچنان با دستور switchdesk پنجره ای ظاهر می شود که محیط های گرافیکی را لیست می کند تا انتخاب کنیم که البته در این دوره کار با محیط گرافیکی زیاد مدنظر نیست.

```
[n.pardis@lpi etc]$ less /x11/X11/prefdm

#!/bin/sh

PATH=/sbin:/usr/sbin:/bin:/usr/bin

# shut down any graphical boot that might exist
if [ -x /usr/bin/rhgb-client ]; then
    /usr/bin/rhgb-client --quit
fi

# We need to source this so that the login screens get translated
[ -f /etc/sysconfig/i18n ] && . /etc/sysconfig/i18n

# Run preferred X display manager
preferred=
if [ -f /etc/sysconfig/desktop ]; then
    . /etc/sysconfig/desktop
    if [ "$DISPLAYMANAGER" = GNOME ]; then
        preferred=/usr/sbin/gdm
    elif [ "$DISPLAYMANAGER" = KDE ]; then
        preferred=/usr/bin/kdm
    elif [ "$DISPLAYMANAGER" = XDM ]; then
        preferred=/usr/bin/xdm
    elif [ -n "$DISPLAYMANAGER" ]; then
        preferred=$DISPLAYMANAGER
    fi
fi

X11/prefdm
```

در هر runlevel اسکریپت rc مربوط به آن اجرا می‌شود که تعداد زیادی نرم‌افزار را اجرا خواهد کرد.

```
10:0:wait:/etc/rc.d/rc 0
11:1:wait:/etc/rc.d/rc 1
12:2:wait:/etc/rc.d/rc 2
13:3:wait:/etc/rc.d/rc 3
14:4:wait:/etc/rc.d/rc 4
15:5:wait:/etc/rc.d/rc 5
16:6:wait:/etc/rc.d/rc 6
```

```
[n.pardis@lpi ~]$ cd /etc/rc.d/rc5.d/
[n.pardis@lpi rc5.d]$ ls
K00ipmiev K35cyrus-imapd K74uuidd K90bluetooth
K01dnsmasq K35dhcrelay K74ypserv K92arptables_jf
K01setroubleshoot K35dovecot K74ypxfrd K92ip6tables
K01smartd K35vncserver K75netfs K92iptables
K01togg-pega K35winbind K76ipsec K95firstboot
K02avahi-daemon K36dhcp6r K80fcose K97sysstat
K02avahi-dnsconfd K36dhcp6s K80kdump K99cpuspeed
K02NetworkManager K36elastix-firstboot K84bgpd K99ktune
K03rhnsd K36lisa K84ospf6d K99lvm2-monitor
K03yum-updatesd K44rawdevices K84ospfd K99readahead_early
K05anacron K45arpwatch K84ripd K99readahead_later
K05conman K46radvd K84ripngd S00microcode_ctl
K05saslauthd K46watchdog K85mdmonitor S05kudzu
K05wdaemon K50ibmasm K85mdmpd S10network
K09privoxy K50inet K85opensmd S12syslog
K10dc_server K50netconsole K85rpcgssd S13irqbalance
K10psacct K50snmpd K85rpcidmapd S22messagebus
K10radiusd K50snmptrapd K85zebra S26dahdi
K10tcasd K50tux K86nfsllock S55sshd
K12dc_client K50vsftpd K87mcstrans S56xinetd
K12mailman K60edac K87multipathd S58ntpd
K16rarpd K65kadmin K87named S64mysqld
K19ntop K65kprop K87portmap S65dhcpd
K20bootparamd K65krb524 K87restorecond S85atop
K20nfs K65krb5kdc K88auditd S85httpd
K20rstad K69rpvcsvcgssd K88pcscd S90asterisk
K20rusersd K72autofs K88wpa_supplicant S90crond
K20rwhod K73ldap K89dund S90xfs
K20tomcat5 K73ypbind K89hidd S91smb
K24irda K74acpid K89iscsi S95atd
K25squid K74apmd K89iscsid S99local
K28amd K74haldaemon K89netplugd S99webmin
K30sendmail K74ipmi K89openibd
K30spamassassin K74lm_sensors K89pand
K34yppasswdd K74nscd K89rdisc
```

اسم سرویس‌ها یا با s (start) شروع می‌شود یا با k (kill) خاتمه می‌شود. و به همین ترتیب هر سرویسی را خواستیم حذف یا اضافه می‌کنیم.. مخزن تمام سرویس‌ها زیر دایرکتوری init.d است .

```
[n.pardis@lpi rc5.d]$ cd /etc/init.d
[n.pardis@lpi init.d]$ ls
acpid          generic-cloexec  netconsole      rwhod
amd            gpm              netfs           saslauthd
anacron        haldaemon       netplugd       sendmail
apmd           halt             network        setroubleshoot
arptables_jf   hidd             NetworkManager single
arpwatch        httpd            nfs             smartd
asterisk       ibmasm          nfslock        smb
atd            inet             nscd            snmpd
atop           ip6tables       ntop            snmptrapd
auditd         ipmi             ntpd            spamassassin
autofs         ipmievd         openibd        squid
avahi-daemon   ipsec            opensmd        sshd
avahi-dnsconfd iptables        ospf6d        syslog
bgpd           irda             ospfd           sysstat
bluetooth      irqbalance     pand            tcsd
bootparamd    iscsi            pcscd          tog-pegaus
conman         iscsid           pkcsslslotd tomcat5
cpuspeed       kadmin           portmap        tux
crond          kdump            postfix        uidd
cups           killall          privoxy       vncserver
cups-config-daemon kprop            psacct        vsftpd
cyrus-imapd   krb524           qlgc_vnic    watchdog
dahdi          krb5kdc         radiusd       wdaemon
dc_client      ktune            radvd          webmin
dc_server      kudzu           rarpd           winbind
dhcp6r         ldap             rawdevices    wine
dhcp6s         lisa             rdisc           wpa_supplicant
dhcpd          lm_sensors      readahead_early xfs
dhcrelay       lvm2-monitor    readahead_later xinetd
dnsmasq        mailman          restorecond  ypbind
dovecot        mcstrans         rhnsd          yppasswdd
dund          mdmonitor       ripd            ypserv
edac          mdmpd            rpcgssd        ypxfrd
elastix-firstboot messagebus    rpcidmapd    yum-updatesd
exim          microcode_ctl  multipathd   zebra
fcoe          mysqld           rpcsvcgssd
firstboot     named            rstatd
functions
```

هر سرویسی را خواستیم می‌توانیم با فرمان less ببینیم:

```
[n.pardis@lpi init.d]$ less sendmail

#!/bin/bash
#
# sendmail      This shell script takes care of starting and
stopping
#           sendmail.
#
# chkconfig: 2345 80 30
# description: Sendmail is a Mail Transport Agent, which is the
program \
#           that moves mail from one machine to another.
# processname: sendmail
# config: /etc/mail/sendmail.cf
# pidfile: /var/run/sendmail.pid

# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
[ -f /etc/sysconfig/network ] && . /etc/sysconfig/network

# Source sendmail configuration.
if [ -f /etc/sysconfig/sendmail ] ; then
    . /etc/sysconfig/sendmail
else
    DAEMON=no
sendmail
```

```
[n.pardis@lpi init.d]$ less sendmail ss httpd

#!/bin/bash
#
# httpd      Startup script for the Apache HTTP Server
#
# chkconfig: - 85 15
# description: Apache is a World Wide Web server. It is used to serve \
#           HTML files and CGI.
# processname: httpd
# config: /etc/httpd/conf/httpd.conf
# config: /etc/sysconfig/httpd
# pidfile: /var/run/httpd.pid

# Source function library.
. /etc/rc.d/init.d/functions

if [ -f /etc/sysconfig/httpd ] ; then
    . /etc/sysconfig/httpd
fi

# Start httpd in the C locale by default.
HTTPD_LANG=${HTTPD_LANG-"C"}
```

سؤال فکر می کنید kill (rc0.d) shutdown داشته باشد؟

```
[n.pardis@lpi init.d]$ cd /etc/rc.d/rc0.d/
[n.pardis@lpi rc0.d]$ ls
K00ipmievd          K25squid           K65krb524        K87mcstrans
K01dnsmasq          K25sshd            K65krb5kdc      K87multipathd
K01setroubleshoot   K28amd             K69rpcsvcgssd  K87named
K01smartd            K30sendmail       K72autofs       K87portmap
K01tog-pegaus       K30spamassassin  K73ldap         K87restorecond
K02avahi-daemon     K34yppasswdd     K73ypbind       K88auditd
K02avahi-dnsconfd   K35cyrus-imapd  K74acpid        K88pcscd
K02NetworkManager    K35dhcpd          K74apmd         K88syslog
K03rhnsc            K35dhcrelay      K74dahdi        K88wpa_supplicant
K03yum-updatesd     K35dovecot        K74haldaemon   K89dund
K05anacron          K35smb             K74ipmi         K89hidd
K05atd               K35vncserver     K74lm_sensors  K89iscsi
K05conman           K35winbind       K74nscd          K89iscsid
K05saslauthd         K36dhcp6r        K74ntpd         K89netplugged
K05wdammon          K36dhcp6s        K74uidd         K89openibd
K09privoxy          K36elastix-firstboot K74ypserv      K89pand
K10dc_server         K36lisa            K74ypxfrd      K89rdisc
K10psacct            K36mysqld        K75netfs        K90bluetooth
K10radiusd           K44rawdevices   K76ipsec        K90network
K10tcasd             K45arpwatch      K80fcoe         K92arptables_jf
K10webmin            K46radvd          K80kdmp         K92ip6tables
K10xfs               K46watchdog     K84bgpd         K92iptables
K12dc_client          K50ibmasm        K84ospf6d      K95firstboot
K12mailman           K50inet           K84ospf4d      K95kudzu
K15atop              K50netconsole   K84ripd         K97sysstat
K15httpd             K50snmpd         K84ripngd      K99cpuspeed
K16rarpd             K50snmptrapd   K85mdmonitor   K99ktune
K19ntop              K50tux            K85mdmpd       K99lvm2-monitor
K20bootparamd        K50vsftpd       K85messagebus  K99microcode_ctl
K20nfs               K50xinetd       K85opensmd     K99readahead_early
K20rstatd            K60asterisk     K85rpcgssd    K99readahead_later
K20rusersd           K60crond          K85rpclmapd   S00killall
K20rwhod             K60edac          K85zebra        S01halt
K20tomcat5           K65kadmin        K86nfslck      K87irqbalance
```

این سرویس‌ها به ترتیب چیدمان انجام می‌شوند اول همه را kill نمی‌کند ولی بعضی killall می‌کند شوند پس در آخر killall می‌کند سرویس آخر هم پردازشگر را halt می‌کند. نکته‌ای که باید در اینجا به آن توجه داشته باشیم این است که killall فقط سرویس‌هایی را که می‌شناسد kill می‌کند پس اگر مثلاً شطرونچ بازی می‌کنیم آن را نمی‌بندد باید با kill جداگانه بسته شود.

```
[n.pardis@lpi rc0.d]$ man killall
KILLALL(1)                               User Commands                               KILLALL(1)

NAME
    killall - kill processes by name

SYNOPSIS
    killall [-Z,--context pattern] [-e,--exact] [-g,--process-group]
    [-i,--interactive] [-q,--quiet] [-r,--regexp] [-s,--signal signal]
    [-u,--user user] [-v,--verbose] [-w,--wait] [-I,--ignore-case]
    [-V,--version] [--] name ...
    killall -1
    killall -V,--version

DESCRIPTION
    killall sends a signal to all processes running any of the specified
    commands. If no signal name is specified, SIGTERM is sent.

    Signals can be specified either by name (e.g. -HUP) or by number (e.g.
    -1) or by option -s.

    If the command name is not regular expression (option -r) and contains
    a slash (/), processes executing that particular file will be selected
    for killing, independent of their name.
```

برای تغییر runlevel می‌توانیم یکی از روش‌های زیر را انتخاب کنیم:

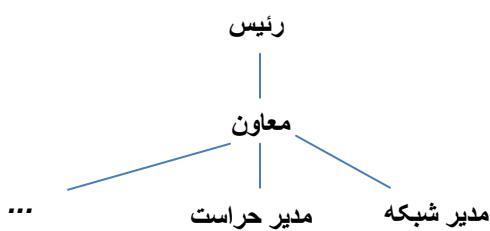
- **init New_runlevel** : init 2
- **telinit New_runlevel** : telinit 2

به مجرد اجرای یکی از فرمان‌های فوق فایل /etc/rc.d/rc مربوط به آن runlevel اجرا خواهد شد. مثلاً خواستیم کلاس را تعطیل کنیم init 0 می‌زنیم برای reboot و یا down نمودن سیستم، می‌توان یکی از دستورالعمل‌های زیر را اجرا نمود. shutdown همه کارها را انجام می‌دهد و در آخر init 0 را صدا می‌کند. init همه کاره است.

- **init 0**
- **shutdown**
- **init 6**
- **reboot**
- **poweroff**
- **halt**
- **ctrl+alt+del**

بررسی Super Server در لینوکس (xinetd)

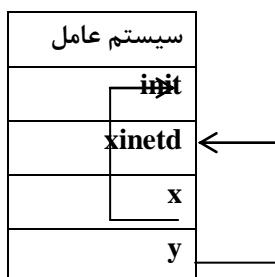
اگر init نباشد یا درست کار نکند می توان گفت که عملا سیستم غیرعملیاتی است. چون هر نرم افزاری بخواهد کار کند init هم به طریقی درگیر می شود. اما وظیفه hinet کند یا ping راه اندازی کند بلکه وظیفه اش این است که یک سری نرم افزار وارد چارت سازمانی کند:



اسم مدیر شبکه در لینوکس xinetd یا super server است. نرم افزارهای تحت سیستم عامل Linux به دو صورت زیر به کاربران تحت شبکه سرویس می دهند:

• به صورت مستقل (Standalone) مثل x

• تحت نظرارت و کنترل پروسس xinetd مثل y



بسیاری از سرویس های شبکه ای از جمله echo ، telnet وغیره تحت نظرارت و سرپرستی پروسسی به نام xinetd که اصطلاحا "Super Server" خوانده می شود قرار دارند.

شما باید در سازمانتان تصمیم بگیرید که نرم افزارتان زیر نظر xinetd باشد یا init

است (فرزنده stand alone . sendmail یا 1 است)

stand alone , telnet نیست.

```
[n.pardis@lpi ~]$ ps -aef |grep sendmail
root      2068  1  0 Aug04 ?
smmsp    2075  1  0 Aug04 ?
for /var/spool/clientmqueue
[n.pardis@lpi ~]$ psa |grep telnet
n.pardis 11258 10865 0 13:18 pts/14  00:00:00 telnet 192.168.100.1
root     11259 2466  0 13:18 ?
n.pardis 11447 11367 0 13:19 pts/20  00:00:00 grep telnet
[n.pardis@lpi ~]$ psa |grep 2466
root     2466  1  0 Aug06 ?
/var/run/xinetd.pid
root     11259 2466  0 13:18 ?
n.pardis 11512 11367 0 13:20 pts/20  00:00:00 grep 2466
00:00:00 sendmail: accepting connections
00:00:00 sendmail: Queue runner@01:00:00
00:00:00 telnetd: 192.168.100.1
00:00:00 in.telnetd: 192.168.100.1
00:00:00 grep telnet
00:00:00 xinetd -stayalive -pidfile
00:00:00 in.telnetd: 192.168.100.1
```

است telnet پدر xinetd

xinetd نرم افزاری است که می تواند نرم افزارهای سرویس دهنده شبکه را مدیریت کند. وقتی در LPI1 آپاچی سرور را راه اندازی کردیم فقط پورت را عوض کردیم ولی اگر پارامترهای config را می خواندید در جایی از شما می پرسید که stand alone باشد یا خیر؟

هر کس از هر جای دنیا به هر سرویسی وصل شود سرویس دهنده باید شبکه (پورت) را گوش کند نرم افزار stand alone خودش گوش می کند اما نرم افزارهایی که فرزند xinetd هستند به جایشان به پورت گوش می دهد و در موقع ضروری آنها را آگاه می کند. همه نرم افزارهای جدی در همه سیستم‌های عامل فایل پیکربندی دارند. سرویس‌ها زمانی که بالا می‌آیند فایل.conf. مربوط به خود را می‌خوانند تا بدانند چه کاری باید انجام دهند. سرویس xinetd که در بعضی از گونه‌های Unix و یا Linux با نام inetd شناخته می‌شود در زمان فعل شدن، به فایل‌ها و دایرکتوری زیر مراجعه نموده و با آنالیز نمودن اطلاعاتی که به دست می‌آورد آماده سرویس دهی می‌شود:

- /etc/xinetd.conf
- /etc/xinetd.d

فایل متنی xinetd.conf که همانند اکثر فایل‌های پیکربندی تحت دایرکتوری etc می‌باشد، شامل اطلاعات کلی برای سرویس دهی تحت شبکه بوده و تحت دایرکتوری /etc/xinetd.d/ نیز تعداد زیادی فایل متنی قرار داشته و به ازای هر سرویس (مثلًا "telnet") میتوان فایل متنی با همان نام مشاهده نمود.

راهبر سیستم عامل با تغییر دادن در فایل‌های فوق می‌تواند کنترل بیشتری را بر روی سرویس دهی داشته باشد.

در لینوکس اکثر فایل‌های پیکربندی text base هستند و با ویرایشگرهای متن مانند Vim قابل تغییر هستند ولی در ویندوز این نوع فایل‌ها داخل رجیستری قرار دارند و با regedit باید کار کرد.

Extended Internet Daemon =XINETD

در مثال رستوران گارسن همان shell است ، پشت صحنه عوامل بسیار زیادی فعالیت می کنند که آن رستوران سرویس بدهد به این عوامل Daemon می گویند که یک معنی آن پشت پرده است . xinetd هم خودش هیچ وقت شخصا برای سرویس دادن عمل نمی کند مانند سرپرست راننده هاست، کسی تماس می گیرد و ماشین می خواهد سرپرست هم برایش ماشین می فرستد.

```
[n.pardis@lpi xinetd.d]$ less /etc/xinetd.conf
#
#Simple configuration file for xinetd
#
#Some defaults, and include /etc/xinetd.d/
defaults
{
    instances          = 60           → حداکثر سرویس ها
    log_type           = SYSLOG authpriv → نوع log و محل قرار گرفتن آن
    log_on_success     = HOST PID      → نام کامپیوتر و شماره پروسس سرویس دهنده ثبت گردد
    log_on_failure     = HOST          → در صورت موفقیت آمیز نبودن ارتباط، نام کامپیوتر ثبت
    cps                = 25 30        → حداکثر 25 ارتباط در ثانیه و در صورت اضافه ترافیک 30 ثانیه مکث و سپس سه سه ده
    includedir         /etc/xinetd.d   → فایل های تحت این دایرکتوری نیز
}
...
```

در فایل xinetd.conf داخل { } مانند برنامه های C پارامترها و مقادیرشان تعریف شده اند که xinetd از روی اینها موقع بالا آمدن می فهمد که به چند نفر باید سرویس بدهد و مثلا یک میلیون نفر می توانند telnet کنند یا 500 هزار نفر ftp کنند. در واقع xinetd فایلی دارد که از روی آن می تواند بخواند که چه کسی می تواند چه کسی نمی تواند و چه کسی نباید بتواند!

instances=60 یعنی بیش تر از 60 نفر را سرویس نمی دهد حالا اگر 45 نفر ftp نفر 61 که تلاش کند وصل شود با پیغام connection refused مواجه می شود. این عدد را در یک سازمان مثلا برابر 600 می گذاریم و برای شبکه های خانگی و کوچک 2 یا 3 پس این عدد در سازمان ها برای سرویس هایی که stand alone نیستند باید عوض شوند.

همان طور که موقع ورود نگهبان از شما کارت تشخیص هویت می خواهد xinetd نیز برای هر ارتباطی identification طلب می کند و مجموعه این اطلاعات را نیز نگه داری می کند.

```
[n.pardis@lpi ~]$ less /var/log/messages

May 11 21:05:07 lpi syslogd 1.4.1: restart.
May 11 21:05:07 lpi kernel: klogd 1.4.1, log source = /proc/kmsg started.
May 11 21:05:07 lpi kernel: Linux version 2.6.18-194.el5 (mockbuild@x86-
007.buil d.bos.redhat.com) (gcc version 4.1.2 20080704 (Red Hat 4.1.2-
48)) #1 SMP Tue Mar 16 21:52:43 EDT 2010
May 11 21:05:07 lpi kernel: BIOS-provided physical RAM map:
May 11 21:05:07 lpi kernel: BIOS-e820: 0000000000010000 -
000000000009f400 (usable)
May 11 21:05:07 lpi kernel: BIOS-e820: 000000000009f400 -
000000000000a0000 (reserved)
```

اگر کسی telnet کند xinetd هاستش (ip) را نگه می دارد و اینکه با چه کسی کار دارد. در نگهبانی نیز شماره شناسایی مراجعه کننده و شماره کارمندی کسی را که با او کار دارد ثبت می شود. حال اگر شخص شماره شناسایی نداشت log out failure اتفاق می افتد ولی سیستم هاست را ثبت می کند و همه این کارها توسط xinetd انجام می شود. به همین دلیل است که اگر به سایتی حمله کنید شناسایی می شوید؛ userID؛ ip شما ثبت می شود. در مورد cps یا connection per second باید توجه شود که اگر 10000 کاربر طرفیت داشته باشیم و همه با هم وصل شوند مثل این است که 10000 نفر یک دفعه وارد یک اتاق شوند پس منطقی این است که هر دفعه (ثانیه) مثلا 25 نفر وارد شوند. اکثر سرویس های موجود در دایرکتوری xinetd.d xinetd تحت مدیریت xinetd اجرا می شوند مانند chargeon، telnet و ftp که این سه سرویس را در ادامه با به طور مفصل توضیح می دهیم و بقیه سرویس ها نیز شبیه این سه سرویس کار می کنند. حتی یکی از سوال های امتحان ممکن است این باشد که یک سرویس چندخطی با shell script بنویسید. به ازای هر سرویسی در دایرکتوری xinetd.d یک فایل داریم:

```
[n.pardis@lpi ~]$ cd /etc/xinetd.d
[n.pardis@lpi xinetd.d]$ ls
amanda          daytime-dgram   gssftp      rlogin~      tftp
amandaidx       daytime-stream  klogin      rmcp        time-dgram
amidxtape       discard-dgram  krb5-telnet rsh         time-
stream
auth            discard-stream  kshell     rsync        uucp
chargen-dgram   echo-dgram    ktalk      talk
chargen-stream  echo-stream   ntalk      tcpmux-server
chargen-stream~ eklogin      rexec      telnet
cvs             ekrb5-telnet  rlogin     telnet~

```

بررسی فایل های پیکربندی سرویس های ذیل که تحت /etc/xinetd.d وجود دارند:

• سرویس Login از راه دور port 23

این سرویس خیلی استفاده دارد مثلا زمانی که به hubswitch وصل می شوید یا به router.

الان در کلاس خیلی ها با telnet به سرور متصل هستند.

هر سرویسی یک پورت دارد مانند یک سازمان که در کنار شماره تلفن برای هر بخش شماره داخلی متمایز دارد.

کامپیوتر

IP: a.b.c.d

telnet port: 23

sendmail port: 25

ftp server: 21

سازمان

تلفن: 12345678

داخلی شکایات: 43

داخلی ثبت نام: 56

داخلی تعمیرات: 73

همانطور که در سازمان به جای گرفتن داخلی 43 می توانیم درخواست کنیم که به بخش شکایت وصل کنند در لینوکس هم به جای 23 می توانیم بگوییم به telnet وصل کن. هر سرویسی که وارد حافظه می شود اگر تحت شبکه باشد حتما باید یک پورت باز کند :

```
[n.pardis@lpi ~]$ less /etc/services

# /etc/services:
# $Id: services,v 1.42 2006/02/23 13:09:23 pknirsch Exp $
#
# Network services, Internet style
#
# Note that it is presently the policy of IANA to assign a single well-known
# port number for both TCP and UDP; hence, most entries here have two
entries
# even if the protocol doesn't support UDP operations.
# Updated from RFC 1700, ``Assigned Numbers'' (October 1994). Not all
ports
# are included, only the more common ones.
#
# The latest IANA port assignments can be gotten from
# http://www.iana.org/assignments/port-numbers
# The Well Known Ports are those from 0 through 1023.
# The Registered Ports are those from 1024 through 49151
# The Dynamic and/or Private Ports are those from 49152 through 65535
#
# Each line describes one service, and is of the form:
#
# service-name    port/protocol  [aliases ...]  [# comment]

tcpmux          1/tcp           # TCP port service
multiplexer
tcpmux          1/udp           # TCP port service
multiplexer
/etc/services
rje             5/tcp           # Remote Job Entry
rje             5/udp           # Remote Job Entry
```

پورت یک عدد 16 بیتی است بنابراین از 0 تا 65535 متغیر است که از 0 تا 1023 wellknown و تعریف شده است و از 1024 تا 49151 را خیلی از شرکت ها و سازمان ها هم ثبت کرده اند و بقیه پورت ها خالی است.

در کد بالا با telnet سرویس را جست وجو می کنیم.

```
telnet          23/tcp
telnet          23/udp
# 24 - private mail system
lsmtp           24/tcp                      # LMTP Mail Delivery
lsmtp           24/udp                      # LMTP Mail Delivery
smtp            25/tcp          mail
smtp            25/udp          mail
time            37/tcp          timserver
time            37/udp          timserver
rlp              39/tcp          resource      # resource location
rlp              39/udp          resource      # resource location
nameserver      42/tcp          name         # IEN 116
nameserver      42/udp          name         # IEN 116
nicname         43/tcp          whois
nicname         43/udp          whois
```

فرمت فایل services به این صورت است که اول اسم سرویس بعد شماره پورت ، اسلش ، پروتکل .
یک ارتباط دو طرفه مطمئن است و lpp یک ارتباط یک طرفه نامطمئن است ولی مثلا کسی دیگر از udp mail استفاده نمی کند. بینیم در کلاس ما چه پورت هایی باز هست و شنود می شود:

```
[n.pardis@lpi ~]$ nmap 192.168.100.1
Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2012-08-25
14:13 IRDT
Interesting ports on 192.168.100.1:
Not shown: 1674 closed ports
PORT      STATE SERVICE
13/tcp    open  daytime
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
873/tcp   open  rsync
4444/tcp  open  krb524

Nmap finished: 1 IP address (1 host up) scanned in 0.968 seconds
```

ipهای کشورهای ایران و کره حق استفاده از nmap را ندارند! چون بیشتر به درد هکرها می خورد.

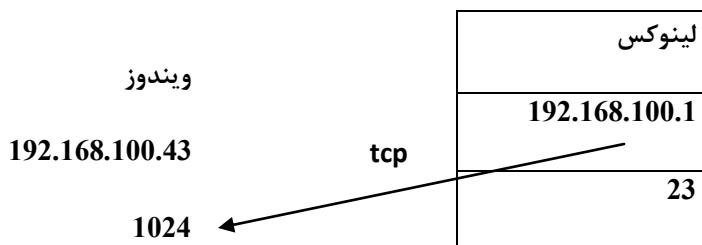
حال می خواهیم بدانیم سرویس دهنده واقعا با چه کسی مرتبط است ، این دستور در ویندوز هم هست:

```
[n.pardis@lpi ~]$ netstat -a | less
```

Active Internet connections (servers and established)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address
State	e			
tcp	0	0	*:rsync	*:*
LISTEN	EN			
tcp	0	0	*:daytime	*:*
LISTEN	EN			
tcp	0	0	*:ftp	*:*
LISTEN	EN			
tcp	0	0	*:telnet	*:*
LISTEN	EN			
tcp	0	0	lpi.aictc.edu:smtp	*:*
LISTEN	EN			
tcp	0	0	81.31.161.42:59489	hosted-by.lease:nfsd-status
ESTABLISHED				
tcp	0	0	192.168.100.1:40894	192.168.100.1:telnet
ESTABLISHED				
tcp	0	0	81.31.161.42:35505	199.21.149.89:http
ESTABLISHED				
tcp	0	0	81.31.161.42:35148	hosted-by.leaseweb.com:ftp
ESTABLISHED				
tcp	0	0	192.168.100.1:telnet	192.168.100.1:40894
ESTABLISHED				
: Active Internet connections (servers and established)				
Proto	Recv-Q	Send-Q	Local Address	Foreign Address
State				
tcp	0	0	*:rsync	*:*
LISTEN				
tcp	0	0	*:daytime	*:*
LISTEN				
tcp	0	0	*:ftp	*:*
LISTEN				
tcp	0	0	*:telnet	*:*
LISTEN				
tcp	0	0	lpi.aictc.edu:smtp	*:*
LISTEN				
tcp	0	0	81.31.161.42:59489	hosted-by.lease:nfsd-status
ESTABLISHED				

فرض کنید سیستمتان را با ویندوز بالا آورده اید و می خواهید با telnet به سرور کلاس وصل شوید به مجرد اینکه دستور telnet را اجرا می کنید این نرم افزار به سیستم عامل می گوید مثلا تو 192.168.100.43 هستی و من میخواهم به 192.168.100.1 داخلی 23 وصل شوم ؛ سیستم شما هم باید پورت داشته باشد ؛ سیستم عامل یک عدد تصادفی به عنوان پورت انتخاب می کند که این عدد معمولا در لینوکس بزرگتر از 32000 و در ویندوز بزرگتر از 1024 است که با این معیار و با احتمال خطای خوبی می توان سیستم عامل کاربر را شناسایی کرد. جفت IPها و پورت هایشان با پروتکل tcp به هم وصل می شوند که اصطلاحا به آن سوکت نرم افزاری می گویند.



```
[n.pardis@lpi xinetd.d]$ cat telnet
# default: on
# description: The telnet server serves telnet sessions; it uses \
#unencrypted username/password pairs for authentication.
service telnet
{
    disable= no

```

سرویس telnet فعال می باشد

در گونه جدید xinetd می توان این پیش فرض را حذف نمود

socket_type= stream → این سرویس با پروتکل tcp کار می کند

wait= no → خادم xinetd منتظر پایان این سرویس نباشد

user= root → این سرویس با userid root (شناسه) root اجرا شود

server= /usr/sbin/in.telnetd → خادمی که توسط xinetd در حافظه مستقر می

log_on_failure+= USERID HOST → در صورت عدم موفقیت login نام userid نیز در Log File ثبت گردد

}

کد بالا محتویات فایل telnet است یعنی اطلاعات به صورت رشته جابه جا می شود. wait را no گذاشته ایم دفعه دوم که درخواست telnet می گیرد اگر wait=yes باشد سرویس نمی دهد تا اولی خارج شود که بهتر است no باشد چون زیاد جالب نیست از هر سرویسی یک عدد ارائه دهیم! درخواست telnet ما در سرور با وقفه ای سیستم عامل را مطلع می کند سیستم عامل می فهمد که وقفه از طرف کارت شبکه اتفاق افتاده است در نتیجه کنترل را در اختیار Device Driver مربوطه قرار می دهد که درایور هم متوجه می شود چه کسی رجیستر کرده است و کنترل را در دست xinetd قرار می دهد که وقتی فهمید درخواست telnet بوده فایل بالا را می خواند و مثلا اگر در این فایل گفته بودیم از ساعت 12 شب به بعد کسی وصل نشود ساعت 2 ارتیاطی برقرار نمی شود. البته الان telnet در سازمان ها زیاد جالب نیست دلیل آن را هم در کامنت های اول فایل نوشته است؛ این پروتکل اطلاعات (از جمله نام کاربری و رمز عبور) را به صورت clear text و رمز گذاری نشده¹ می فرستد یعنی به راحتی می توان اطلاعات کاربری و پیغام های کاربران سیستم را به دست آورد. telnet بیشتر در شبکه های خصوصی کاربرد دارد.

با دستور psa متوجه می شویم که user id تل نت، root است.

تمرین

سرویسی به نام add بسازید که از طریق شبکه دو عدد از کاربر بگیرد و جمع آنها را برگرداند.

به تعداد کسانی که در کلاس telnet کرده اند پرسیس in.telnetd در سیستم وجود دارد:

```
[n.pardis@lpi xinetd.d] $ psa | grep in.telnetd
root    3340  3218  0 17:04 ?          00:00:00 in.telnetd: 192.168.100.223
root    3391  3218  0 17:05 ?          00:00:00 in.telnetd: 192.168.100.222
root    3537  3218  0 17:05 ?          00:00:00 in.telnetd: 192.168.100.217
root    3613  3218  0 17:06 ?          00:00:00 in.telnetd: 192.168.100.218
n.pardis 6982 14728  0 17:55 pts/17    00:00:00 grep in.telnetd
```

پدر همه این آنها پردازش pid شماره 3218 با xinetd کند telnet یک کپی از خودش می سازد(fork)

less /var/log/services

less /var/log/secure

آرشیو اتفاقات log_on_failure در فایل /var/log/secure قرار دارد.

¹ unencrypted

Chargen • port 19 : سرویس تست پایانه و شبکه

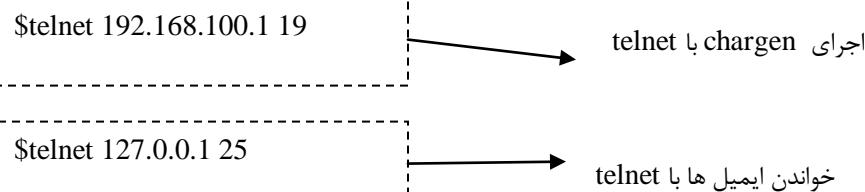
پورت 19 برای تولید حروف و تست شبکه و پایانه مورد استفاده قرار می‌گیرد و توسط خود نرم افزار xinetd سرویس داده می‌شود.

```
[n.pardis@lpi xinetd.d]$ cat chargen-stream
# This is the configuration for the tcp/stream chargen service.

service chargen
{
    disable=
    id= chargen-stream
    type= INTERNAL → سرویس دهنده ، خود xinetd می باشد
    wait= no
    socket_type= stream
    protocol= tcp
    user= root
}
```

با ویرایش گر vi در فایل بالا ، no می کنیم ولی اگر بلاfacslه nmap بزیم مشاهده می کنیم که هنوز chargen فعال نشده است یک راه حل این است که سرویس restart xinetd را کنیم که تا دوباره فایل سرویس ها را بخواند و بفهمد که باید chargen را اجرا کند البته راه حل های بهتری هم وجود دارد که در ادامه دوره آنها را ذکر می کنیم. هنگامی که یک شبکه را تحويل می گیرید روی هر یک از telnet clientها نصب می کنیم مانیتورها را خاموش می کنیم و روز بعد به سراغ سیستم ها می رویم اگر همه Kرده باشند یعنی hub switch مشکل دارد و هر سیستمی که به تنها ی هنگ کرده باشد یعنی کارت شبکه یا نرم افزار مشکل دارد. chargen که اجرا می شود بی نهایت کاراکتر و پیغام در شبکه می فرستد حتی شبکه و سرور کلاس را مختل می کند و هیچ دستوری کار نمی کند که با [ctrl+ و سپس quit از این وضعیت خارج می شویم.

سرویس internal , chargen است یعنی xinetd نرم افزاری کاراکتر فرستادن در شبکه اجرا نمی کند بلکه شخصا این کار را انجام می دهد.



• iPop 3 : سرویس بررسی نامه ها port 110

از این سرویس خیلی استفاده می کنید که در یاهو هزینه دارد ولی در گوگل مجانی است. نامه ها را به چند طریق می توانید بخوانید یکی با پورت 25 ویا با پورت 80 از طریق webmail مثل سایت یاهو. همچنین می توانید روی وینوز یا لینومستان نرم افزاری نصب کنید که نامه هایتان را از روی دیسک mailserver بخواند و روی کامپیوتر شما قرار دهد. برای این کار باید نرم افزار دیگری روی سیستم شما نصب باشد که پورت پیش فرض آن 110 است. شما در سیستم عامل خودتان به پورت 110 متصل می شوید و نامه بر نامه ها را از میل سرور گرفته و به شما تحويل می دهد؛ برای ارسال نامه نیز شما نامه را به نامه بر می دهید و نامه بر نامه ها را به میل سرور تحويل می دهد.

Internet Post Office Protocol version 3 = ipop3

```
service pop3
{
    disable = yes
    socket_type = stream
    wait = no
    user = root
    server = /usr/sbin/ipop3d
    log_on_success += HOST DURATION → مدت زمان ارتباط نیز log شود
    log_on_failure += HOST
}
```

اگر تمام سرویس های موجود تحت دایرکتوری xinetd را مشاهده کنید ساختارها شبیه به هم است :

```
cat *|less
```

اگر الان پورت telnet را عوض کنیم مثلا 196 بگذاریم هیچکس در کلاس نمی تواند telnet کند مگر اینکه در سیستم خود تغییراتی اعمال کند که وقتی telnet می کند پورت 196 باز شود.

جلوی پارامتر iponly_from، only_from را که اجازه ورود دارند را قرار می دهیم و جلوی پارامتر ip_no_access کسانی که نمی خواهیم به شبکه وصل شوند را وارد می کنیم.

```
man xinetd.conf
```

در این فایل پارامترهایی را که می توانیم در سرویس ها استفاده کنیم توضیح داده شده است. xinetd یک شاهکار نرم افزاری است که هر نیازی داشته باشید به راحتی با تغییر این فایل برطرف می شود مثلاً توضیحات access_times کاملاً گویاست که اگر خواستیم شبکه بین ساعت 2:14 تا 3:18 در دسترس باشد access_times را برابر 2:14=3:18 قرار می دهیم. با این manual های لینوکس به راحتی به هر نیاز جدیدی می توان پاسخ داد در حالی که حتی اگر چند سال با ویندوز کار کرده باشید نمی دانید وقتی به شبکه وصل می شوید چه سرویس ها یا نرم افزارهایی فعال می شوند. نمی توانیم بگوییم ویندوز خوب است یا بد، ولی doc ندارد به خاطر همین لینوکس اگر به هم بریزد می توان آن را درست کرد ولی ویندوز را باید reboot کنیم.

خیلی موقع می پرسند که پورت ها را چه طور بیندیم؟! پورت به تنها بی کاره ای نیست باید سرویس مربوطه را بیندید. به عنوان مثال برای بستن پورت 23 باید سرویس telnet را disable کنیم و سپس xinetd را restart می کنیم.

```
[n.pardis@lpi xinetd.d]$ cat telnet
# default: on
# description: The telnet server serves telnet sessions; it uses \
#unencrypted username/password pairs for authentication.
service telnet
{
    disable= yes
    flags= REUSE
    socket_type= stream
    wait= no
    user= root
    ...
}
[root@lpi xinetd.d]# /etc/init.d/xinetd restart
Stopping xinetd:
Starting xinetd:
```

یکی از بهترین راه ها برای اینکه سرویسی فایل config مربوط به خود را دوباره بخواند (reinitialize) این است که سیگنال 1 یا hang up را به آن ارسال کنیم . در ورودی بیمارستان ها برای هر پزشکی یک تابلو قرار دارد که اگر حضور داشته باشد چرا غ آن روشن است. در لینوکس هم سرویسی که بالا می آید رد پایش را در دایرکتوری /var/run به جا می گذارد. هر سرویس مهم در این دایرکتوری یک فایل pid دارد :

```
n.pardis@lpi ~]$ cd /var/run
[n.pardis@lpi run]$ ls *.pid
-rw-r--r-- 1 root root 5 Aug 4 13:55 atd.pid
-rw-r--r-- 1 root root 5 Aug 4 13:55 crond.pid
-rw-r--r-- 1 root root 5 Aug 4 13:55 dhcpd.pid
-rw----- 1 root root 5 Aug 11 11:39 klogd.pid
-rw-r--r-- 1 root root 5 Aug 4 13:55 messagebus.pid
-rw----- 1 root smmsp 33 Aug 4 13:55 sendmail.pid
-rw-r--r-- 1 smmsp smmsp 49 Aug 4 13:55 sm-client.pid
-rw-r--r-- 1 root root 5 Aug 4 13:55 sshd.pid
-rw----- 1 root root 5 Aug 11 11:39 syslogd.pid
-rw-r--r-- 1 root root 6 Aug 4 13:55 xfs.pid
-rw-r--r-- 1 root root 5 Aug 6 17:52 xinetd.pid
```

داخل این فایل ها process id سرویس مربوطه قرار دارد:

```
[n.pardis@lpi run]$ cat xinetd.pid
11413
```

هر کدام از سرویس ها را که start یا stop کنیم یک سطر به این فایل اضافه یا کم می شود:

start/stop vsftpd & 1 *pid

برای اعمال این تغییرات باید restart xinetd را کنیم البته توصیه می شود از reload استفاده کنیم که فقط یک بار دیگر فایل پیکربندی را بخواند. به منظور اطلاع دادن به پروسس xinetd در مورد تغییراتی که در فایل های /etc/xinetd.d انجام داده اید یکی از روش های زیر می تواند به کار گرفته شود:

A:

1.ps -aef|grep xinetd

2.kill -1 p#

P# شماره پروسس به دست آمده از ردیف 1 می باشد.

B:

kill -1 `cat /var/run/xinetd.pid`

در صورت دریافت سیگنال 1 مجددا فایل های تحت قلمرو خود را خوانده و به عبارتی خود را Reinitialize Xinetd می نماید.

نکته: بسیاری از نرم افزارها در شروع اجرا شماره پروسس خود را تحت /var/run قرار می دهد.

- راه اندازی سرویس (service xinetd start) xinetd
- توقف سرویس (service xinetd stop) xinetd
- گرفتن گزارش از وضعیت (service xinetd status) xinetd

برای دریافت اطلاعات بیشتر به سایت زیر مراجعه گردد:

<http://cr.yp.to/daemontools.html>

سرویس هایی وجود دارند که ردپایی در /var/run به جا نمی گذارند ولی راه های دیگری وجود دارد که متوجه شویم از نگهبانی عبور کرده اند.

بررسی پروتکل telnet

دستور nmap درصد استاده از پروتکل ها در دنیا را نیز می تواند نمایش دهد که طبق این آمار ، استفاده از telnet خیلی زیاد است چون همه می خواهند مودم ها یا router و Hub Switch را تنظیم کنند. با استفاده از امکانات telnet می توان از یک خادم Linux (یا ویندوز) گرفته و فعالیت های مورد نیاز را انجام داد. کاربران خادم Linux می توانند از راه دور از طریق فرمان telnet به خادم متصل شده Login ،

و این امکان را دارند تمام فرمان هایی که از طریق کنسول قادر به اجرای آن بودند را اجرا نمایند. در صورت نیاز به سرویس telnet بایستی تحت /etc/xinetd.d و در فایل telnet تغییرات لازمه را اعمال نمود. به صورت پیش فرض و به خاطر مسائل امنیتی telnet ندارید و باید /etc/xinetd.conf فایلش را ویرایش کنید (disable=no) ولی ssh فعال است. Xinetd بعد از بالا آمدن و فعال شدن به فایل پیکربندی telnet را مراجعه نموده و اصطلاحاً "خودش را Initialize می نماید. اگر راهبر در فایل /etc/xinetd.d تحت telnet کنترل راه اندازی telnet را داده باشد پروسس xinetd "الاما" خود را پاسخگوی تقاضاهای رسیده از پورت شماره 23 دانسته و در صورت رسیدن پیام از این پورت، فعالیت های لازمه را به منظور سرویس دهی انجام می دهد.

- رسیدن خبر از پورت 23 به منظور سرویس دهی
- کنترل فایل telnet (محتويات اين فایل در حافظه قرار دارد)
- کنترل های متعدد با توجه به فایل های پیکربندی
- قراردادن in.telnetd در حافظه به وسیله system call Fork & Exec متقاضی به عنوان آرگومان
- اعلام به سیستم عامل در مورد اینکه اگر child تولیدشده رفت، او را بی خبر نگذارد! به وسیلهتابع Signal
- پایان مأموریت و به دنبال کارهای دیگر رفتن!....

```
[n.pardise@lpi run] $ man fork
FORK(2)                               Linux Programmers Manual                      FORK(2)

NAME
    fork - create a child process

SYNOPSIS
    #include <sys/types.h>
    #include <unistd.h>

    pid_t fork(void);

DESCRIPTION
    fork() creates a child process that differs from the parent process
    only in its PID and PPID, and in the fact that resource utilizations
    are set to 0. File locks and pending signals are not inherited.

    Under Linux, fork() is implemented using copy-on-write pages, so the
    only penalty that it incurs is the time and memory required to dupli-
    cate the parents page tables, and to create a unique task structure
    for the child.
```

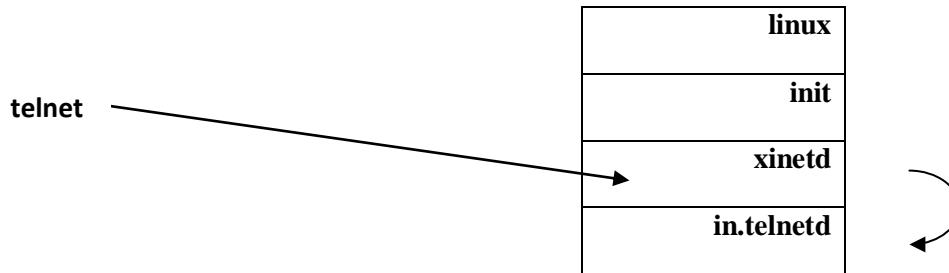
تمرین: چند نفر الان با telnet به سرور کلاس وصلند؟

```
[n.pardis@lp1 xinetd.d] $ psa |grep in.telnetd|wc -l  
13
```

telnet یک تابع یک به یک نیست از یک سیستم می‌توان چند telnet انجام داد.

در system call سیستم عامل را صدای زنیم ولی در function call تابعی مانند سیسنوس را اجرا می‌کنیم.

از هر جایی که telnet فرزند xinetd درست می‌کند و in.telnetd را در آن قرار می‌دهد:



هر نرم افزاری که از بین بود حتماً یک child از بین رفته (حتماً parent دارد) و همه یک عمر مفید دارند. دلیل اینکه در صورت kill شدن فرزند باید پدرش را خبر کنیم این است که پدر باید پایگاه داده اش را به روز کند.

فعالیت های in.telnet

- تبادل اطلاعات با سرویس گیر Do-Don't-Will-Won't
- ارسال فایل /etc/issue.net به سرویس گیر
- انتظار برای رسیدن Userid (با مهلت 1 دقیقه)
- فعال نمودن نرم افزار login و رد کردن Userid به آن
- اعلام به سیستم عامل در مورد اینکه اگر child تولیدشده از بین رفت او را بی خبر نگذارد! و به وسیله تابع signal
- استراحت و کنترل های مقطعي به منظور اطمینان از ارتباط

برای روشن شدن موضوع مثالی می زنیم : هر پنجه ای که در ویندوز باز است (اگر ویندوز را مادر فرض کنیم) حال مادر موقع تحويل فرزندش به مهدکودک (لینوکس) توصیه هایی می کند:

: اگر گرسنه شد غذا بدھید. Do

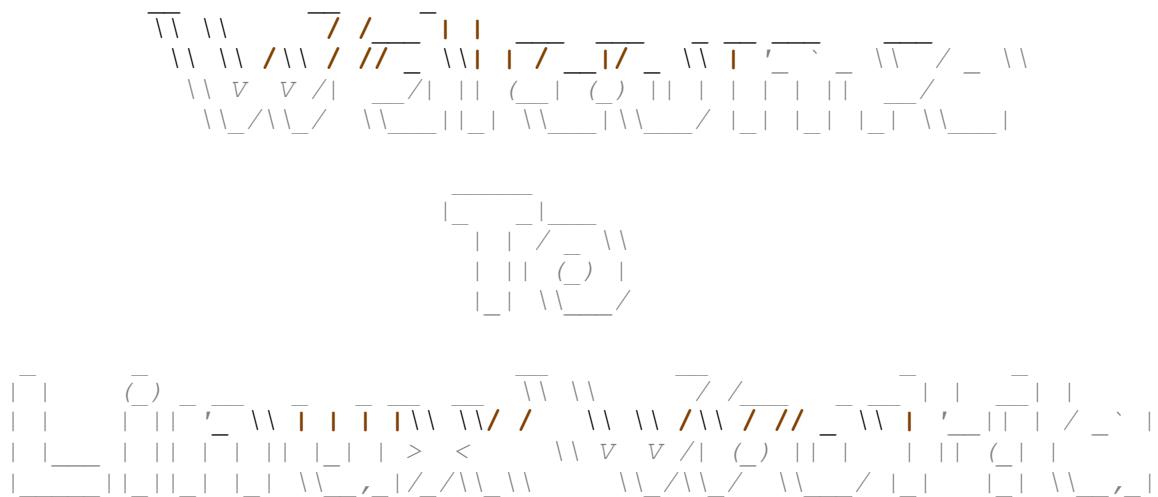
: شیر سرد ندهید! Don't

: اگر گریه کرد اسباب بازی بدھید. Will

: اگر خوابش برد بیدارش نکنید! Won't

به مجرد اینکه شما telnet می زنید telnet از ویندوز شما با daemon از سرور صحبت های زیادی می کنند.

```
[n.pardis@lpi run]$ cat /etc/issue.net
Red Hat Enterprise Linux AS release 5 (Update 5) [AICTC SHARIF UNIVERSITY]
Kernel \r on an \m
```



در فایل issue.net تنظیمات صفحه آغازین شبکه قرار دارد. یک جا اعلام می کنید که پسورد را نمایش ندهد. اگر احتمال حمله به سایتتان وجود دارد توصیه می شود این فایل را خالی بگذارید و از چاپ نسخه و اسم سیستم عامل سرور در این صفحه خودداری کنید.

```
[n.pardis@lpi ~]$ man issue
ISSUE(5)                               Linux Programmers Manual      ISSUE(5)

NAME
    issue - pre-login message and identification file

DESCRIPTION
    The file /etc/issue is a text file which contains a message or system
    identification to be printed before the login prompt. It may contain
    various @char and \char sequences, if supported by mingetty(8).

FILES
    /etc/issue

SEE ALSO
    mingetty(8), motd(5)

Linux                                         1993-07-24                                ISSUE(5)
(END)
```

```
[n.pardis@lpi ~]$ man mingetty
MINGETTY(8)                               Linux Programmers Manual      MINGETTY(8)

NAME
    mingetty - minimal getty for consoles

SYNOPSIS
    mingetty [--noclear] [--nonewline] [--noissue] [--nohangup] [--nohostname]
              [--long-hostname]     [--loginprog=/bin/login]   [--nice=10]
              [--delay=5]           [--chdir=/home]        [--chroot=/chroot]  [--autologin username]
              tty
...
ISSUE ESCAPES
    mingetty recognizes the following escapes sequences which might be
    embedded in the /etc/issue file:

    \d      insert current day (localtime),
    \l      insert line on which mingetty is running,
    \m      inserts machine architecture (uname -m),
    :
    \n      inserts machines network node hostname (uname -n),
    \o      inserts domain name,
    \r      inserts operating system release (uname -r),
    \t      insert current time (localtime),
    \s      inserts operating system name,
    \u      resp. \U the current number of users which are currently logged
          in. \U inserts "n users", where as \u only inserts "n".
    \v      inserts operating system version (uname -v).

EXAMPLE
    "Linux eos i386 #1 Tue Mar 19 21:54:09 MET 1996" was produced by
    putting "\s \n \m \v" into /etc/issue.
```

همان طور که در manual های بالا به روشنی گفته شده می توانید در صفحه اول تاریخ و زمان یا تعداد افرادی که login کرده اند را مشاهده کنید. نرم افزاری است که login می دهد نه اینکه فقط مخصوص issue باشد؛ فقط یک فایل قابل نمایش است.

”login“ را نرم افزار login نمی دهد:

```
[n.pardis@lpi ~]$ ps -aef
...
root      28799  2466  0 17:05 ?          00:00:00 in.telnetd: 5.113.147.148
root      28800 28799  0 17:06 ?          00:00:00 login -- n.pardis
....
```

همان طور که در ترمینال بالا مشاهده می کنید login (پدر userid 28799) in.telnetd را صدا زده (پدر login است) و پارامتر به آن ارسال می کندو سپس in.telnet به سیستم عامل می گوید که اگر login شد مرا خبردار کن به عبارتی اول in.telnetd وارد حافظه می شود و id را می گیرد سپس login می آید و userID را می گیرد.



فعالیت های login

- بررسی وجود فایل /etc/motd و نمایش آن
- بررسی فایل /etc/passwd به منظور یافتن shell
- اجرای fork برای فعال نمودن shell موردنظر کاربر
- اعلام به سیستم عامل در مورد اینکه اگر child تولید شده از بین رفت، او را بی خبر نگذارد! و به وسیله تابع signal
- استراحت مطلق

فرض کنید با telnet به جایی وصلید و در حال اجرای یک بازی هستید حال اگر ارتباط قطع شود بازی نمی فهمد . راه حل این است که in.telnetd هر چند وقت یک بار ping می کند که ببیند ویندوز هنوز در ارتباط است (مهدکوک هر دقیقه زنگ می زند تا مطمئن شود mainframe مادر گم نشده باشد!) و اگر نبود به بازی و login سیگنال Hangup می فرستد و آنها را از بین می برد. در جایی لینوکس و ping به هم وصل بوده اند ولی همیشه تعدادی از telnetها قطع می شدند. معلوم شد که ویروسی به شدت شبکه را مشغول می کرده و اطمینان از حضور client برنامی گشته است و سرور هم به تبع این ارتباط ها را قطع می کرده است.

برای اینکه بخواهیم از الان به بعد هیچ کس telnet نکند فایلی به نام nologin در /etc می سازیم (می توانیم دلیل قطع ارتباط را داخل آن بنویسیم تا کاربر مشاهده کند)

اگر داخل nologin چیزی ننویسیم موقع telnet کردن ، با پیغام زیر مواجه می شویم:

Connection closed by foreign host.

توصیه می شود کسی با root ، telnet نکند به صورت پیش فرض هم نمی توان با login کرد مگر اینکه در /etc/securstty عنوان کنیم که چه ترمینال هایی می توانند با root ، telnet کنند:

```
[root@lpi ~]# cat /etc/securetty
console
vc/1
vc/2
vc/3
vc/4
vc/5
vc/6
vc/7
vc/8
vc/9
vc/10
vc/11
tty1
tty2
tty3
tty4
tty5
tty6
tty7
tty8
tty9
tty10
tty11
```

tty بزندید اگر یکی از این ترمینال ها بودید می توانید با root login کنید:

```
[centos@localhost ~]$ tty  
/dev/pts/2
```

Pseudo Terminal Stream = pts

يا همان پایانه، از نظر لینوکس هر کس وصل می شود یک پایانه است که یک سری اطلاعات را به صورت رشته‌ای (stream) می فرستد. pseudo یعنی دروغی؛ لینوکس نمی داند این را دار است یا وسیله دیگری است!

. اگر ایمیل داشته باشید موقع login به شما اطلاع می دهد و کلا پیغام های موقع login را مدیریت می کند.

```
[n.pardis@lpi etc]$ man motd  
  
MOTD(5)                               Linux Programmers Manual  
MOTD(5)  
  
NAME  
      motd - message of the day  
  
DESCRIPTION  
      The contents of /etc/motd are displayed by login(1) after a  
      successful  
      login but just before it executes the login shell.  
  
      The abbreviation "motd" stands for "message of the day", and this  
      file  
      has been traditionally used for exactly that (it requires  
      much less  
      disk space than mail to all users).  
  
FILES  
      /etc/motd  
  
SEE ALSO  
      login(1), issue(5)  
  
Linux                                         1992-12-29  
MOTD(5)
```

Message Of The Day = Motd

پیغام هایی که در موقع ورود مشاهده می کنید که مثلاً موبایل هایتان را خاموش کنید یا در بعضی سازمان ها اطلاعیه هایی برای کارمندان چاپ می شود. لینوکس با خواندن فایل `passwd` متوجه می شود که چه کسی حق دارد داخل شود و چه shell بی داشته باشد. fork می کند و shell را وارد می کند. پس بعد از `login`، `bash` وارد حافظه می شود. به عنوان مثال اگر 15 نفر `telnet` کرده باشند 15 تا `login` داریم، 15 تا `bash` و 15 تا `daemon` 3000 پردازش در حافظه داریم. همین داستان هم برای `ftp` ووب سرور اتفاق می افتد. چرا `login` پس از این مراحل کار دیگری ندارد، اگر به یاد داشته باشید `in.telnetd` استراحت مطلق نداشت. اگر به مکان نظامی (مثلاً برای تعمیر سرور) بروید در نگهبانی سربازی با شما همراه می شود و تا شما روی سیستم کار می کنید سرباز هم در گوشه ای می خوابد و فقط موقع رفتن بیدار می شود و برگه خروج شما را امضا می کند و ساعت خروج را ثبت می کند. `login` هم موقع `logout` باید می شود و زمان ها را ثبت می کند.

محصول last است که زمان های ورود و خروج را در خود نگه می دارد:

```
[n.pardis@lpi etc]$ last | less
n.pardis pts/0      5.122.42.115      Tue Aug 28 09:42      still logged
in
a.ghavab pts/2      80.191.228.2      Mon Aug 27 23:42 - 23:44 (00:02)
h.kianer pts/1      89.165.16.174      Mon Aug 27 23:41 - 01:42 (02:01)
h.bashir pts/0      37.63.173.5      Mon Aug 27 23:23 - 00:20 (00:57)
v.shalch pts/0      31.58.81.136      Mon Aug 27 21:13 - 22:05 (00:52)
v.shalch pts/0      31.58.81.136      Mon Aug 27 21:12 - 21:13 (00:00)
h.bashir pts/0      37.63.173.5      Mon Aug 27 19:46 - 20:31 (00:45)
n.pardis pts/0      5.123.138.3      Mon Aug 27 17:20 - 17:23 (00:03)
m.tavako pts/1      130.255.233.100    Mon Aug 27 14:52 - 14:58 (00:05)
m.tavako pts/0      178.131.74.181    Mon Aug 27 14:27 - 15:41 (01:13)
n.pardis pts/0      5.113.211.165    Mon Aug 27 10:43 - 11:54 (01:10)
h.rajaei pts/0      66.197.219.226    Mon Aug 27 09:45 - 10:26 (00:41)
...

```

```
[n.pardis@lpi etc]$ last n.pardis
n.pardis pts/0      5.122.42.115      Tue Aug 28 09:42      still logged in
n.pardis pts/0      5.123.138.3      Mon Aug 27 17:20 - 17:23 (00:03)
n.pardis pts/0      5.113.211.165    Mon Aug 27 10:43 - 11:54 (01:10)
n.pardis pts/1      5.118.172.145    Sat Aug 25 23:26 - 23:28 (00:02)
n.pardis pts/2      5.112.119.163    Sat Aug 25 22:25 - 22:30 (00:05)
n.pardis pts/17     192.168.100.1    Sat Aug 25 14:11 - 14:50 (00:39)
n.pardis pts/12     5.119.82.206     Sat Aug 25 14:10 - 14:55 (00:45)
n.pardis pts/20     192.168.100.1    Sat Aug 25 13:18 - 13:29 (00:10)
n.pardis pts/14     5.122.66.110     Sat Aug 25 13:07 - 13:29 (00:22)
n.pardis pts/5      5.113.228.104    Fri Aug 24 20:02 - 20:16 (00:13)
n.pardis pts/3      5.113.228.104    Fri Aug 24 19:06 - 20:47 (01:41)
```

فعالیت های shell

- خواندن /etc/profile و اجرای آن
- بررسی bash_profile و اجرای home Directory
- بررسی وجود چند file دیگر
- نمایش prompt و منتظر خدمت به کاربر!
- اجرای bash_logout در صورت رسیدن خداحافظی
- بازنشسته شدن و مراسم خداحافظی shell

بررسی گزارش عملیات Linux

سؤالات لینوکس پیشرفته

چرا هسته اصلی لینوکس (kernel) مدیر پروسس ها نیست؟

کرنل بر روی منابع مدیریت می کند و برای انجام کارها معاون دارد. جرا `userid` را خودش چک نمی کند و چرا خودش `ftp` نمی کند؟ چون اگر خودش بخواهد در گیر ریزه کاری ها شود، کارهای بزرگ را از دست می دهد. فرض کنید سیستم در روال عادی خودش است و یک بازی یا سرویس `telnet` یا `ftp` در حال اجرا هستند. در این حین اگر `interrupt` اتفاق بیفتد سیستم عامل کنترل را در دست می گیرید و همان طور که ریسیس جمهور وقتی جلسه مهمی دارد کسی را نمی پذیرد و تلفنی را جواب نمی دهد، سیستم عامل هم، وقفه ها را در این مدت به اصطلاح `mask` می کند. `mask` مثل این است که اگر تلفن زنگ زد جواب ندهم حتی اگر رعدوبرق زد تکان نخورم؛ `mask` بگذارم! پس بیش ترمواقعی که سیستم عامل مشغول کار است `interrupt` هستند (غیر فعال آند). حالا اگر سیستم بخواهد همه کارها را خودش انجام دهد `interrupt` ها همیشه `mask` هستند. پس اولاً کرنل فوق العاده تنومند و سنگین می شود و ثانیاً وقفه ها همیشه `mask` هستند و بقیه نمی توانند کار کنند.

نقش پروسس `init` چیست و اگر به دلایلی `kill` یا `terminate` شود چه اتفاقی خواهد افتاد؟

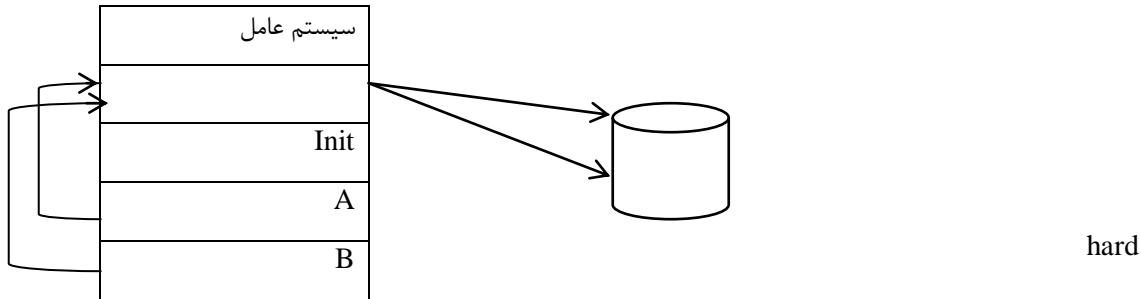
سروریس ها را بالا می آورد و روی آنها مدیریت می کند در واقع پدر بزرگ همه سرویس هاست. اگر `init` را `kill` کنید معمولاً سعی می کند دوباره بالا بباید و خودش را `back up` کند ولی اگر نتوانست `kernel` خودکشی می کند زیرا کرنل بدون `init` نمی تواند زندگی کند. یعنی وزیر داشته باشیم ولی زیر مجموعه نداشته باشد. اصطلاحاً سیستم `panic` می دهد. احتمالاً در ویندوز xp صفحه آبی را که نقطه چین دارد مشاهده کرده اید (در لینوکس هم هست)؛ این همان خودکشی سیستم عامل است.

چرا اکیدا توصیه می گردد که سیستم را بدون اجرای `shutdown` خاموش نکنید؟

اکثر نرم افزاره فایل هایی را باز می کنند که داخل آن بنویسنند، این فایل ها در حافظه نگه داری می شوند به این عمل `cache` یا `buffer` می گویند.

```
[n.pardis@lpi etc]$ top
top - 20:57:04 up 6:52, 6 users, load average: 0.00, 0.05, 0.02
Tasks: 136 total, 1 running, 134 sleeping, 0 stopped, 1 zombie
Cpu(s): 0.3%us, 1.0%sy, 0.0%ni, 98.7%id, 0.0%wa, 0.0%hi, 0.0%si,
0.0%si
Mem: 1555424k total, 1234996k used, 320428k free, 161276k buffers
Swap: 0k total, 0k used, 0k free, 840664k cached
```

اگر همین الان کامپیوچر را (بدون shutdown) خاموش کنید یا جریان برق قطع شود مقداری اطلاعات بافر وجود دارد که هنوز روی دیسک نرفته است. اگر نرم افزار A در حال نوشتن روی دیسک است ، داده ها فعلا در حافظه ذخیره می شود و بعدا سرر فرصت توسط یک نرم افزار به دیسک انتقال می یابد.



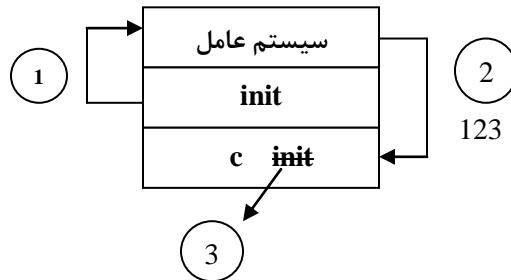
به خاطر همین است که performance لینوکس به نظر سریع تر است ؛ تا یک فلاش مموری را در هارد کپی می کنید ، به سرعت آن را در هارد می بینیم ، 100 برابر سریع تر از ویندوز! ولی در ویندوز وقتی یک فایل بزرگ را کپی می کنید تا به طور کامل کپی نشود ، shutdown نمی گیرید. بنابراین خاموش کردن سیستم بدون prompt در سیستم های عملیاتی و جدی که هزاران نفر به آن متصل اند خیلی خطرناک است چون مقداری از این اطلاعات از دست می روند.

runlevel چیست و معمولا لینوکس با چه runlevel‌یی بالا می آید؟

بستگی به توزیع دارد ؛ دیلان ممکن است با 2 بالا باید و سیستم کلاس ما با 3 بالا می آید.

respawn را تشریح نموده و بگویید که کاربرد استفاده از آن چیست؟

اگر قرار باشد نرم افزاری همیشه در حافظه باشد مثل رادر ، یا نرم افزار اتاق ccu یا رطوبت سنج به init می گوییم که همیشه یک نسخه از این نرم افزار ها در حافظه داشته باشد. در ابتدا init به سیستم عامل می گوید که یک کپی از او بسازد. سپس آن نرم افزار (مثلا C) را جای خودش می نشاند. init به سیستم عامل می گوید هر موقع C مرد مرا خبر کن. وقتی این اتفاق افتاد سیستم عامل پیغام Death of child را نمایند. اگر pid 123 متعلق به C باشد) را به init می فرستد و init دوباره یک کپی از خودش می سازد و C را جایگزین می کند و این روند همینطور ادامه دارد:



تعدادی دایرکتوری با نامهای `rc3.d` ، `rc2.d` ، `rc1.d` قرار دارد کاربرد این دایرکتوری ها چیست؟

دایرکتوری runlevel که قبلا توضیح داده شد.

برای آنکه یک سرویس تحت یک runlevel اجرا نگردد چه می کنید؟

یک راه این است که فایل آن سرویس را پاک کنیم ولی سعی کنید زیاد به دایرکتوری هایی که سیستم عامل نصب می کنید دست نزنید.

راه دوم استفاده از دستور `chkconfig` است:

```
[n.pardis@lpi etc]$ man chkconfig
CHKCONFIG(8)
CHKCONFIG(8)

NAME
    chkconfig - updates and queries runlevel information for system services

SYNOPSIS
    chkconfig --list [name]
    chkconfig --add name
    chkconfig --del name
    chkconfig [--level levels] name <on|off|reset>
    chkconfig [--level levels] name

DESCRIPTION
    chkconfig provides a simple command-line tool for maintaining the /etc/rc[0-6].d directory hierarchy by relieving system administrators of the task of directly manipulating the numerous symbolic links in those directories.
```

```
[n.pardis@lpi etc]$ chkconfig --list|less
...
rpcsvcgssd      0:off   1:off   2:off   3:off   4:off   5:off   6:off
setroubleshoot  0:off   1:off   2:off   3:on    4:on    5:on    6:off
smartd          0:off   1:off   2:on    3:on    4:on    5:on    6:off
squid           0:off   1:off   2:off   3:off   4:off   5:off   6:off
sshd            0:off   1:off   2:off   3:on    4:off   5:off   6:off
syslog          0:off   1:off   2:on    3:on    4:on    5:on    6:off
wdaemon         0:off   1:off   2:off   3:off   4:off   5:off   6:off
winbind          0:off   1:off   2:off   3:off   4:off   5:off   6:off
wpa_supplicant  0:off   1:off   2:off   3:off   4:off   5:off   6:off
xfs              0:off   1:off   2:on    3:on    4:on    5:on    6:off
xinetd          0:off   1:off   2:off   3:on    4:on    5:on    6:off
ypbind          0:off   1:off   2:off   3:off   4:off   5:off   6:off
```

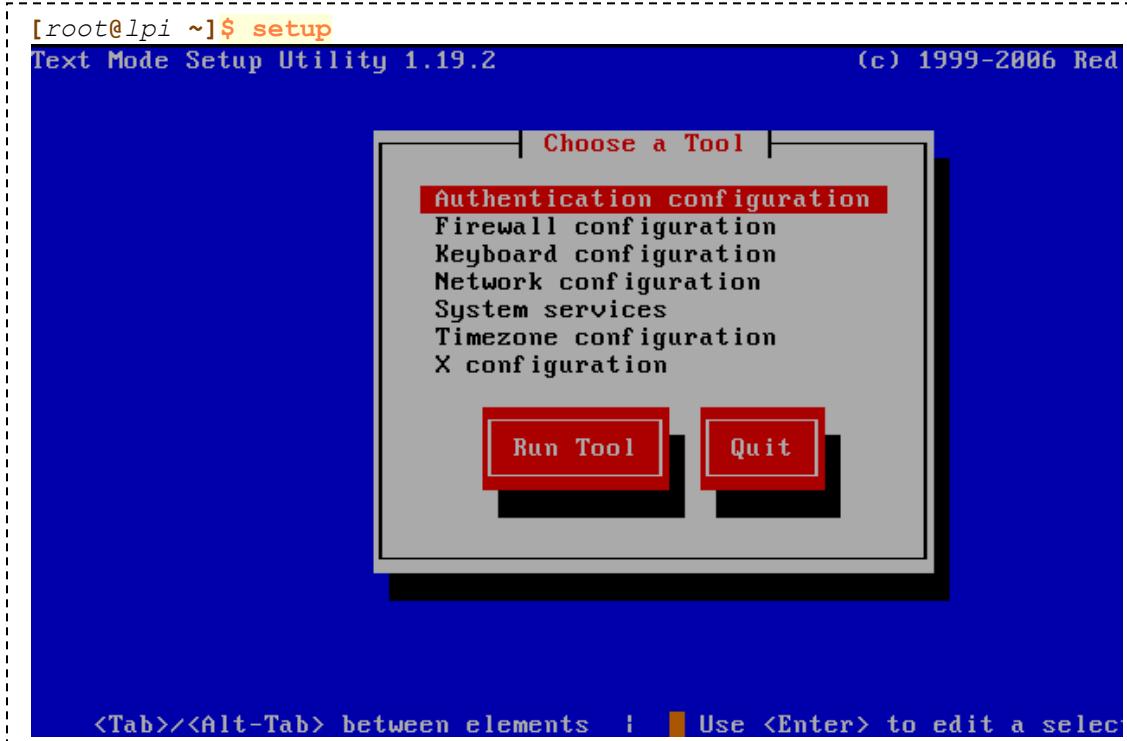
همانطور که در ترمینال بالا مشاهده می کنید ، در هر خط نام یک سرویس قرار دارد که شماره هر runlevel روبروی آن به همراه وضعیت سرویس در آن on (برای فعال و off (برای غیرفعال) لیست شده است. به همان ترتیبی که در manual ذکر شده می توانیم سرویس ها را در یک runlevel فعال یا غیرفعال کنیم. به عنوان مثال در ترمینال زیر سرویس sshd را برای rc-1 فعال می کیم:

```
[root@lpi ~]# chkconfig --level 1 sshd on
[root@lpi ~]# chkconfig --list|grep sshd
sshd           0:off    1:on     2:off    3:on     4:off    5:off    6: off
```

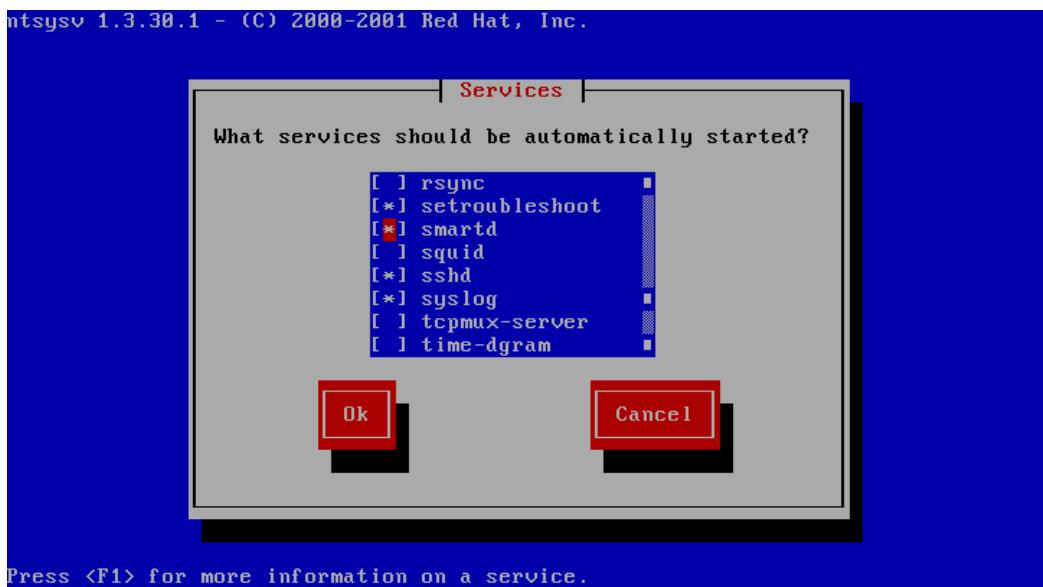
البته در محیط گرافیکی هم می شود این کار را انجام داد:

system controller > system services

ولی توصیه می شود عادت کنید که از محیط گرافیکی استفاده نکنید چون خیلی از data center ها و کامپیوترهای گران قیمت اصلا ندارندبا دستور setup هم می توان سرویس ها را کم و زیاد کرد ولی این دستور فقط در توزیع fedora ، redhat و centos وجود دارد ، همچنین تغییرات اعمال شده فقط برای runlevel جاری apply می شود:



که با انتخاب system services وارد پنجره زیر می شویم و هر سرویسی را خواستیم فعال کنیم با کلید space یک * در کنار آن قرار می دهیم:



فرمان shutdown چه می کند (دقیقاً شرح دهید)

ابتدا بررسی می کند grace period چقدر است و مهلتش چقدر است به ازای هر چند دقیقه یک بار ، یک پیغام می فرستد و مهلت را کم می کند ولی در آخر init را صدا می زند به عبارتی همه راه ها به رم ختم می شود!

به فایل inittab نگاهی اندادته و powerfail را تشریح کنید.

```
pf:::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"
```

از مطالب قبلی (kill -l) به یاد دارید که سیگنال powerfail ، 30 است و اگر وضع برق خوب نباشد init 2 پیغام power failure را برای همه ارسال می کند و 2 دقیقه بعد سیستم را shutdown می کند.

فرمان 3 init چه کاری را انجام می دهد؟

دوباره فایل inittab را می خواند ، با 3 runlevel فایل ها را می خواند . زیر rc3.d می بیند که چه سرویس هایی را باید بالا بیاورد (اگر قبلش هم در 3 بوده ایم هیچ کاری نمی کند چون همه سرویس ها بالاست). موقعی که unix درست شد به شاخه های مختلفی تقسیم شد که بعد ها join شدند ، یک شاخه از init استفاده می کرد و دیگری از telinit که زیاد با هم تفاوتی ندارند.

```
[n.pardis@lpi /]$ man itelinit
INIT(8)                         Linux System Administrator>s Manual
INIT(8)

NAME
    init, telinit - process control initialization

SYNOPSIS
    /sbin/init [ -a ] [ -s ] [ -b ] [ -z xxx ] [ 0123456Ss ]
    /sbin/telinit [ -t sec ] [ 0123456sSQqabcUU ]

DESCRIPTION
    Init
        Init is the parent of all processes. Its primary role is to create
        processes from a script stored in the file /etc/inittab (see inittab(5)). This file usually has entries which cause init to spawn gettys on each line that users can log in. It also controls autonomous
        processes required by any particular system.

    RUNLEVELS
        A runlevel is a software configuration of the system which allows only
        a selected group of processes to exist. The processes spawned by init
        for each of these runlevels are defined in the /etc/inittab file. Init
        can be in one of eight runlevels: 0>6 and S or s. The runlevel
        is
        changed by having a privileged user run telinit, which sends appropri-
```

سوالات بخش xinetd

اگر بخواهید هیچ سرویس شبکه ای ارائه نشود چه فعالیتی را انجام می دهید؟

یک جواب می تواند اجرای init باشد. ولی اگر فقط xinetd را stop کنیم فقط نرم افزارهایی که زیر نظر xinetd کار می کنند kill down و بقیه نرم افزارهای شبکه در حافظه می مانند. راه حل اصلی این است که سرویس شبکه را stop کنیم یعنی کارت شبکه را کنیم:

```
[n.pardis@lpi ~]$ ifconfig eth0
eth0      Link encap:Ethernet HWaddr 00:0C:29:29:7D:8B
          inet addr:192.168.206.141 Bcast:192.168.206.255
          Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe29:7d8b/64 Scope:Link
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
             RX packets:22547 errors:0 dropped:0 overruns:0 frame:0
             TX packets:5541 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:1829653 (1.7 MiB) TX bytes:612665 (598.3 KiB)
             Interrupt:67 Base address:0x2000

[n.pardis@lpi ~]$ ifconfig eth0 down
```

یک راه بدتر برای این کار که در data center ها هم غیر عملیاتی است؛ کابل ها را قطع کنیم.

توجه داشته باشید که اگر شبکه را از راه دور (remotely) قطع کنیم ارتباط خودمان هم قطع می شود.

اگر به طور تصادفی فایل xinetd.conf حذف شود، چه اتفاقی می افتد؟

```
[n.pardis@lpi ~]$ rm /etc/xinetd.conf
```

چون از قبل این سرویس فعال بوده پس فایل پیکربندی آن در حافظه قرار دارد ولی اگر xinetd را restart کنیم چون این فایل را پیدا نمی کند نمی تواند کار کند. در ویندوز اگر فایلی busy باشد نمی توان آن را پاک کرد ولی در لینوکس مثلاً اگر فایل در vi باز باشد و آن را پاک کنیم آن را علامت‌گذاری می کند تا بعد از آزاد شدن فایل آن را از روی دیسک پاک کند. کسی پرسیده بود که درایوم 95 درصد پر بوده، یک فایل 100 مگابایتی را پاک کردم ولی باز هم 95 درصد پر بود، فایل busy بوده است.

چه فعالیتی انجام شود که نتوان به سادگی فایل xinetd.conf را حذف نمود. (حتی با root) جواب: بر روی اکثر توزیع ها هست (نه همه) chattrib

```
[n.pardis@lpi ~]$ man chattr
CHATTR(1)

NAME
    chattr - change file attributes on a Linux second extended file system

SYNOPSIS
    chattr [ -RV ] [ -v version ] [ mode ] files...

DESCRIPTION
    chattr changes the file attributes on a Linux second extended file system.

    The format of a symbolic mode is +--[ASacDdIijsTtu] .

    The operator '+' causes the selected attributes to be added to the existing attributes of the files; '-' causes them to be removed; and '=' causes them to be the only attributes that the files have.

    The letters 'acdijsuADST' select the new attributes for the files: append only (a), compressed (c), no dump (d), immutable (i), data journaling (j), secure deletion (s), no tail-merging (t), undeletable (u), no atime updates (A), synchronous directory updates (D), synchronous updates (S), and top of directory hierarchy (T).
```

```
[root@lpi ~]# man lsattr
Formatting page, please wait...
LSATTR(1)

NAME
    lsattr - list file attributes on a Linux second extended file system

SYNOPSIS
    lsattr [ -RVadv ] [ files... ]

DESCRIPTION
    lsattr lists the file attributes on a second extended file system. See chattr(1) for a description of the attributes and what they mean.

OPTIONS
    -R      Recursively list attributes of directories and their contents.

    -V      Display the program version.

    -a      List all files in directories, including files that start with '.'.

    -d      List directories like other files, rather than listing their contents.

    -v      List the files version/generation number.
```

توابع فایل سیستم هستند و ممکن است در بعضی از توزیع ها در دسترس نباشند. همانطور که در manual گفته شده است می توانیم با افزودن `+i` به `chattr` فایل را تغییرناپذیر^۱ کنیم یا با `+d` کاری کنیم که نتوان از فایل `back up` گرفت.

```
[root@lpi ~]# chattr +i /etc/xinetd.conf
[root@lpi ~]# lsattr /etc/xinetd.conf
----i----- /etc/xinetd.conf
[root@lpi ~]# rm /etc/xinetd.conf
rm: remove write-protected regular file '/etc/xinetd.conf'? y
rm: cannot remove '/etc/xinetd.conf': Operation not permitted
```

پس در لینوکس با `chmod` ، `chattr` ، `attribute` ها را عوض می کنیم و با `admin` اشتباها یک فایل مهم را پاک می کند که اگر از این روش استفاده کنیم با دیدن پیغام `operation not permitted` متوجه می شود.

آیا دونرم افزار می توانند همزمان یک پورت (port) را بازنمایند؟

وقتی پورتی رجیستر شد دیگر نمی توان از آن استفاده کرد همانطور که دو نفر در یک سازمان نمی توانند یک داخلی داشته باشند.

```
[root@lpi ~]# less /etc/services
...
ftp          21/tcp
ftp          21/udp      fsp  fspd
ssh          22/tcp
                           # SSH Remote Login
Protocol
ssh          22/udp      # SSH Remote Login
Protocol
telnet       23/tcp
telnet       23/udp
:# 24 - private mail system
lmtp         24/tcp      # LMTP Mail Delivery
lmtp         24/udp      # LMTP Mail Delivery
smtp         25/tcp      mail
smtp         25/udp      mail
```

پورت شماره 67 به چه کاری می خورد؟

```
[root@lpi ~]# less /etc/services
...
bootps       67/tcp      # BOOTP server
bootps       67/udp
bootpc       68/tcp      dhcpc
bootpc       68/udp      dhcpc
```

¹ immutable

پورت 67 دو کاربرد دارد؛ وقتی سیستم را روشن می کنید ip ندارد با پورت 67 اعلام می کند که ip می خواهد و با پورت 68 هم جوابش را می گیرد.

آیا پورت 104 کاربردی دارد؟

این پورت در services وجود ندارد پس می توانیم آن را برای نرم افزاری که خودمان نوشته ایم استفاده کنیم مثل نرم افزارهایی که عکس های رادیوگرافی و یا سی تی اسکن بیماران را به کامپیوتر پردازش کنند و دیگر نیازی به چاپ عکس نیست از این پورت استفاده می کنند.

اگر xinetd را بوسیله فرمان kill از سیستم حذف نماییم، چه اتفاقی می افتد؟

خودش دوباره بالا نمی آید چون در respawn ندارد. جواب این سوال تا حدی بستگی دارد. بعضی از نرم افزارها موقعی که پدرشان می میرد خودکشی می کنند. سیگنال hangup یعنی parent از بین رفته است؛ نرم افزارهایی که سیگنال hangup را ignore کنند در حافظه می مانند.

```
[n.pardis@lpi ~]$ man nohubp
NOHUP(1)                               User Commands                               NOHUP(1)

NAME
    nohub - run a command immune to hangups, with output to a non-tty

SYNOPSIS
    nohub COMMAND [ARG]...
    nohub OPTION

DESCRIPTION
    Run COMMAND, ignoring hangup signals.

    --help display this help and exit
    --version
        output version information and exit

    NOTE: your shell may have its own version of nohub, which usually
          supersedes the version described here. Please refer to your
shell[Span>s
    documentation for details about the options it supports.

AUTHOR
    Written by Jim Meyering.
```

اگر من با مودم به شرکت فرش گیلان وصل باشم و گیری 3 ساعت وقت ببرد اگر logout کنم سیگنال hangup به دستور tar می رسد و از بین می روید. اگر nohup tar را اجرا کنیم وقتی به tar خبر برسد که پدرش کشته شده باز هم در حافظه می ماند. وقتی init با fork کپی می شود این کپی هم userID و متغیرهای محلی را می دارد ولی وقتی نرم افزار a را جایگزین خود کند این نرم افزار به متغیرهای محلی دسترسی ندارد.

اگر متغیر محلی تعريف کنید فقط برای همان shell ، set شده است و دیگران به آن دسترسی ندارند(به انتهای env افزوده نمی شود):

```
[n.pardis@lpi ~]$ ip=1.2.3.4
[n.pardis@lpi ~]$ env | less
...
:LESSOPEN=| /usr/bin/lesspipe.sh %s
G_BROKEN_FILERAMES=1
_=bin/env
(END)
```

ولی با پارامتر export ، متغیر به فایل env افزوده شده و اگر نرم افزاری در حافظه داشته باشد این نرم افزارها اطلاعات شما را به ارث می بردند:

```
[n.pardis@lpi ~]$ export ip=1.2.3.4
[n.pardis@lpi ~]$ env | less
...
:LESSOPEN=| /usr/bin/lesspipe.sh %s
ip=1.2.3.4
G_BROKEN_FILERAMES=1
_=bin/env
(END)
```

فلسفه fork این است که سرپرست رانده ها همه چیز را می داند و وقتی خودش را کپی می کند آن کپی هم اطلاعات لازم را می دارد. دقیقاً متغیرهای export را نشان می دهد، وقتی shell دستور date را اجرا می کند date می داند که به کدام ترمینال بددهد، می داند که userid کسی که اجرایش کرده چیست (همه اطلاعات env را می دارد)

```
[n.pardis@lpi ~]$ help export
export: export [-nf] [name[=value] ...] or export -p
NAMES are marked for automatic export to the environment of
subsequently executed commands. If the -f option is given,
the NAMES refer to functions. If no NAMES are given, or if '-p'
is given, a list of all names that are exported in this shell is
printed. An argument of '-n' says to remove the export property
from subsequent NAMES. An argument of '--' disables further option
processing.
```

آیا می دانید چه تفاوت هایی بین **xinetd** و **inetd** وجود دارد؟

سوکت چیست؟ (از نوع نرم افزاری)

ip+port+protocol

برای راه اندازی **xinetd** حداقل چند کارت شبکه نیاز می باشد؟

حداقل 0 تا می خواهیم! چون با مودم هم می توان **xinetd** telnet داد. هیچ ربطی به کارت شبکه ندارد فقط اگر کسی از بیرون وصل شد به او سرویس می دهد و می توانید از **usb** یا پورت سریال (com1 و com2 در مودم های قدیمی) به آن وصل شد و حتی از راه دور کرد.

گزارش عملیات لینوکس

لینوکس یک سرور است و اگر اطلاعات **log** نشوند؛ اتفاقی بیفتند نمی توان بررسی کرد که ریشه اش کجا بوده و آن را حل و فصل کرد. در سیستم عامل لینوکس دو نرم افزار (حداقل) برای **log** کردن در حافظه قرار می گیرند:

Syslogd • بررسی اصلی مربوط به **log** نمودن

Klogd • بررسی **log** مربوط به هسته (kernel)

```
[n.pardis@lpi ~]$ ps -aef|less
...
root      3160      1  0 19:01 ?        00:00:00 syslogd -m 0
root      3163      1  0 19:01 ?        00:00:00 klogd -x
...
```

هر دوی این نرم افزارها فرزند **init** هستند (اکثر نرم افزارهای اساسی فرزند **init** هستند). همانطور که مثلا در دانشگاه یک حراست برای دانشجویان داریم و یکی برای اساتید دو نوع **log** داریم. پروسس **Syslogd** در زمان بالا آمدن سیستم اجرا می گردد که تحت دایرکتوری /etc/rc.d/init.d قرار دارد.

```

[n.pardis@lpi ~]$ less /etc/rc.d/init.d/syslog
#!/bin/bash
#
# syslog      Starts syslogd/klogd.
#
#
# chkconfig: 2345 12 88
# description: Syslog is the facility by which many daemons use to log \
# messages to various system log files. It is a good idea to always \
# run syslog.
### BEGIN INIT INFO
# Provides: $syslog
### END INIT INFO

# Source function library.
. /etc/init.d/functions

RETVAL=0

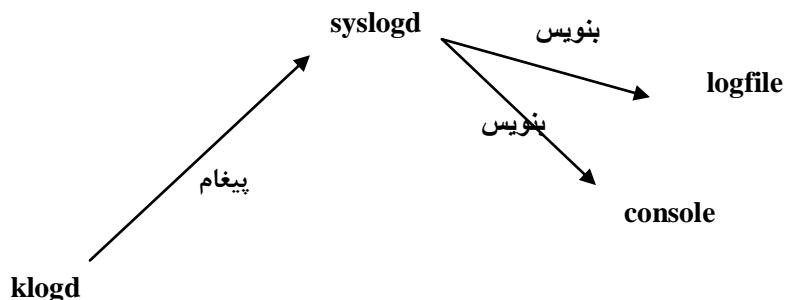
start() {
    [ -x /sbin/syslogd ] || exit 5
    [ -x /sbin/klogd ] || exit 5

```

به هیچ وجه Syslogd را در سیستم غیرفعال نکنید، در غیر این صورت در صورت بروز هر مشکلی شما مسئول خواهید بود.

خودتان هم می توانید داخل syslog ، log بنویسید. معمولاً "خواندن log فایل‌ها پیچیده است به دلیل اینکه دستگاه‌های مختلف بر روی آن پیغام‌های مختلف می‌گذارند و برای فهم پیغام گذاشته شده نیاز به آشنایی با دستگاه تولید‌کننده وجود دارد (اسم رجیسترها و اطلاعات فنی).

Klogd نیز پیام‌های Kernel را به Syslogd ارسال می‌نماید و Syslogd پیام را در فایل می‌نویسد.



اگر یک پروسس تصمیم بگیرد پیامی را log نماید آن را به پروسس Syslogd ارسال می نماید و معمولاً "پیامها در فایل نوشته می شوند. فرمات log در فایل به صورت زیر می باشد (bash این قوانین را دقیق اجرا نمی کند):

Tm_stmp host application[pid]: message

تاریخ و ساعت واقعه:Tm_stmp

Host : نام کامپیوتری که پیام در آن تولید شده است.

نام برنامه و شمارهٔ پروسس: application[pid]

پیام مربوطه message

```
[n.pardis@lpi ~]$ less ./var/log/messages
May 11 21:05:07 lpi kernel: using mwait in idle threads.
May 11 21:05:07 lpi kernel: CPU: Trace cache: 12K uops, L1 D cache: 16K
May 11 21:05:07 lpi kernel: CPU: L2 cache: 2048K
May 11 21:05:07 lpi kernel: CPU: Hyper-Threading is disabled
May 11 21:05:07 lpi kernel: Intel machine check architecture supported.
May 11 21:05:07 lpi kernel: Intel machine check reporting enabled on
CPU#0.
May 11 21:05:07 lpi kernel: CPU0: Intel P4/Xeon Extended MCE MSRs (24)
available
```

حال سوال این است که اسم سیستم (lpi) در همه خط‌ها تکرار شده، نمی‌توان از نوشتan آن صرف نظر کرد؟

پاسخ این است که لینوکس درا می‌توان به عنوان log server استفاده کرد. مثلاً 10 سال پیش در یکی از طبقات شهرداری تهران 30 تا سرور داشتیم و به تبع بررسی log file تک تک اینها کار زمان بری می‌شد این بود که یکی از آنها را log server کردیم و به بقیه گفتیم که یک کپی از log‌هایتان را هم برای این بفرستید.

```
[root@lpi ~]# less /etc/services
shell          514/tcp        cmd                      # no passwords used
syslog         514/udp
```

این یکی از استثناهاست چون بقیه پورت‌ها udp و tcp یکسانی دارند (کاربرد یکسان) ولی پورت 514، tcp متعلق به shell است و syslog هم از udp استفاده می‌کند. این یکی از پرسش‌های مشکل شبکه هم هست. یک سوال امتحانی می‌تواند این باشد که چه طوری log server را linux کنیم؟

در کلاس history‌ها را تغییر دادهایم به طوری که حتی اگر bash log history را پاک کنید می‌توان با showhist وبا پارامتر اسم کاربر دستورات اجرا شده (حتی rm history) را هم مشاهده کرد.شما باید بلد باشید که چطور به log server بگویید که به پورت 514 گوش کند.در پارامترهای log که قبلاً تعدادی از آنها را بررسی کردیم ، این تنظیمات وجود دارد و به بقیه سیستم عامل‌ها هم می‌گوییم که هایشان را در پورت 514 بریزند.

دلایل ایجاد log‌ها

Log‌ها معمولاً" به دلایل زیر تولید می‌گردند:

- خطاهای کنترل
- عدم اجرای درست su (وارد نمودن درست اسم رمز)
- خطاهای سخت افزاری
- telnet,ssh,ftp از طریق Login
- فعالیت‌های نامه بر(mail server)
- فرآیندها shutdown و Startup
-

مثلاً اگر برنامه سنگینی را اجرا کنید(!n برای n‌های بزرگ) درایور cpu مرتب روی کنسول می‌نویسد که درجه حرارت cpu بالا رفته است(log می‌دهد).

Log ها برای وضوح بیشتر با توجه به حساسیت‌شان دسته بندی می‌شوند که در جدول زیر آورده شده است و همان طور که می‌بینید هرچه به صفر نزدیک‌تر می‌شویم وضعیت وخیم‌تر است.

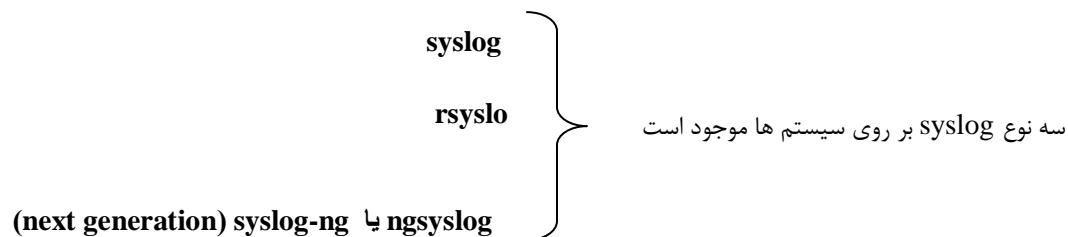
رتبه بندی	واژه	شرح
0	emergencies	سیستم عملاً "غیرقابل استفاده" است
1	alerts	باید سریعاً "عكس العمل" نشان دهیم
2	critical	شرایط بحرانی می‌باشد
3	errors	خطایی در سیستم وجود دارد
4	warnings	اخطار....
5	notifications	شرایط عادی ولی مشکلاتی وجود دارد
6	informational	جهت اطلاع....
7	debugging	پیام‌های مربوط به Debugging سیستم

موقعی که فن cpu گیر کرده و درجه حرارت به شدت زیاد شود، این emergencies است.

alert = swap file در حال پرشدن است

درجه حرارت بالای critical = 80 درجه

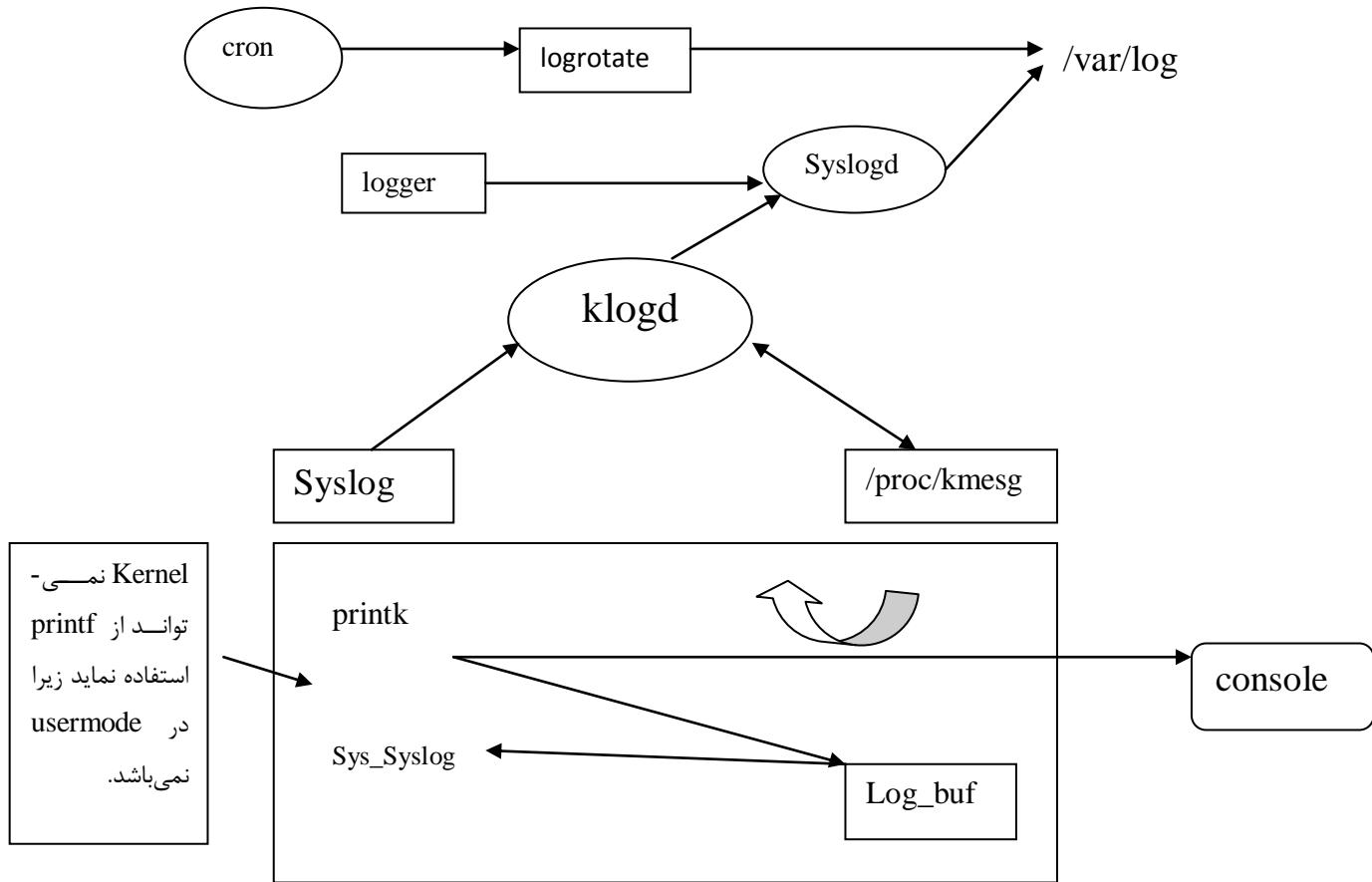
اینها سطوح 0 تا 7 هستند که با 3 بیت قابل ذخیره اند. در خیلی از سیستم عامل‌ها عبارت **FYI: For Your Information** چاپ می‌شود (Information



که قدیمی‌ترین و ساده‌ترین آنها syslog است و به صورت پیش فرض روی اکثر توزیع‌ها نصب است. next generation روی توزیع‌های جدیدتر (فدورای 10 و بالاتر) نصب است و قابلیت‌ها و قدرت‌بیشتری دارد.

ساختار logging

همان طور که در شکل زیر مشخص است همه راه ها به پارتیشن var ختم می شود:



ارتباطات موجود برای log

در شکل بالا همه راه ها به var ختم می شود. logها به این دلیل در var ذخیره می شود که var مخفف variable است و طول logها هم اصلا ثابت نیست. زیر klog هر چیزی که هست در ذات کرنل است. هیچ درسی نداریم که در آن کرنل را کد کنند چون خیلی پیچیده و پیشرفته است ولی همین را بدانید که حق نداریم از printf استفاده کنیم.

می خواهیم با زبان برنامه نویسی C کدی بنویسیم که پیغامی را log کند:

```
[n.pardis@lpi ~]$ vi test.c
#include <syslog.h>
main()
{
openlog("mytest", LOG_PID, LOG_USER);
syslog(LOG_WARNING, "Testing ...");
closelog();
}
[n.pardis@lpi ~]$ gcc -o log test.c
[n.pardis@lpi ~]$ ./log
[n.pardis@lpi ~]$ less /var/log/messages
Aug 30 19:03:37 lpi bash: HISTORY: PID=27364 UID=8022 ./log
Aug 30 19:03:37 lpi mytest[27430]: Testing ...
Aug 30 19:03:55 lpi bash: HISTORY: PID=27364 UID=8022 less
/var/log/messages
```

با دستور logger هم می توانیم پیغام بنویسیم ، logger را در shell script استفاده می کنیم و C را در پروژه ها.

پس خودمان هم می توانیم در برنامه هایمان و با زبان های مختلف log بنویسیم و بهتر خواهد بود اگر ادمین زودتر و از طریق log ها از مشکلات (مثل مشکلی در پایگاه داده) خبردار شوند نه این که توسط کاربران مطلع شوند.

به مجرد تولید خطای توسعه یکی از اجزای kernel ، نرم افزار klogd برای تبدیل آدرس های عددی به نام routine ، به مکان های زیر مراجعه می نماید.

- boot/system.map
- /usr/src/linux/system.map

و با تبدیل آدرس های عددی به اسمی، log file تولید شده را بهتر می توان آنالیز نمود.

در داخل فایل اول کد قابل اجرای^۱ فایل log قرار دارد:

```
[n.pardis@lpi ~]$ nm log|less

0804953c d _DYNAMIC
08049608 d _GLOBAL_OFFSET_TABLE_
08048508 R _IO_stdin_used
    w _Jv_RegisterClasses
0804952c d __CTOR_END__
08049528 d __CTOR_LIST__
08049534 D __DTOR_END__
08049530 d __DTOR_LIST__
08048524 r __FRAME_END__
08049538 d __JCR_END__
08049538 d __JCR_LIST__
0804962c A __bss_start
08049628 D __data_start
080484c0 t __do_global_ctors_aux
08048360 t __do_global_dtors_aux
0804850c R __dso_handle
08049528 d __fini_array_end
08049528 d __fini_array_start
    w __gmon_start
080484b9 T __i686.get_pc_thunk.bx
08049528 d __init_array_end
08049528 d __init_array_start
08048440 T __libc_csu_fini
:
08048450 T __libc_csu_init
    U __libc_start_main@@GLIBC_2.0
08049528 d __preinit_array_end
08049528 d __preinit_array_start
0804962c A _edata
08049634 A _end
080484e8 T _fini
08048504 R _fp_hw
08048294 T _init
08048310 T _start
08048334 t call_gmon_start
    U closelog@@GLIBC_2.0
08049630 b completed.5791
08049628 W data_start
0804962c b dtor_idx.5793
080483c0 t frame_dummy
080483e4 T main
    U openlog@@GLIBC_2.0
    U syslog@@GLIBC_2.0
(END)
```

¹ executable

این دستور سیمبل های executable را می خواند و نشان می دهد (name list) بعضی از اینها جدول اند و برخی دیگر نرم افزار هستند. جزئیات اینها را در درس کامپایلر دانشگاه آموزش می دهند فقط بدانید شما وقتی برنامه ای می نویسید سیستم هم یک سری symbol به آن اضافه می کند. به هر زبان برنامه نویسی که برنامه بنویسید کامپایلر جیزهایی را به آن اضافه می کند. به عنوان مثال فایل log.exe با این سیمبل ها 4 کیلوبایت حجم دارد ولی بعد از اینکه با فرمان strip این سیمبل ها را حذف کنیم حجم آن 3 کیلوبایت می شود:

```
[n.pardis@lpi ~]$ ls log
-rwxr-xr-x 1 n.pardis lpi1 4957 Aug 30 19:03 log
[n.pardis@lpi ~]$ strip log
[n.pardis@lpi ~]$ ls log
-rwxr-xr-x 1 n.pardis lpi1 3100 Aug 30 23:29 log
[n.pardis@lpi ~]$ nm log
nm: log: no symbols
```

کرنل لینوکس symbol table ندارد و گرنه خیلی بزرگ می شد و الان فقط حاوی کدهای صفر و یک اسامبلی است. به همین دلیل اگر درایور cd-rom بخواهد پیغام دهد که مشکل دارد یا کارت شبکه بخواهد بننویسد که فریم آشغال آمده است چون اسم روتین ها را نمی داند آدرس ماثول ها را می نویسد.

```
[n.pardis@lpi boot]$ ls System.map-2.6.18-194.el5
-rw-r--r-- 1 root root 967675 Mar 17 2010 System.map-2.6.18-194.el5
[n.pardis@lpi boot]$ less System.map-2.6.18-194.el5

00000400 A __kernel_vsyscall
00000410 A __SYSENTER_RETURN
00000420 A __kernel_sigreturn
00000440 A __kernel_rt_sigreturn
00400000 A phys_startup_32
c0400000 T _text
c0400000 T startup_32
c0401000 T startup_32_smp
c0401080 t checkCPUtype
c0401101 t is486
c0401108 t is386
c040116a t check_x87
c0401192 t setup_idt
c04011af t rp_sidt
c04011bc t ignore_int
c04011f0 T _stext
c04011f0 t run_init_process
c04011f0 T stext
c040122c t init_post
```

فایل symbol table .map کرنلی است که کامپایل شده است. در یکی از فازها این system table از کرنل جدا می شود. در این فایل klogd می فهمد که کدام روتین مربوط به کدام آدرس است. به عنوان مثال در آدرس c04011f0 کرنل init را صدا می زند. در واقع در range نوشته شده که هر آدرسی متعلق به چه روتینی است و وقتی شما لینوکس را نصب می کنید این فایل هم روی سیستم نصب می شود. نماد T به معنی table است. برای پیکربندی logging سیستم، از فایل /etc/syslog.conf استفاده می شود با تنظیمات مربوطه در این فایل می توانیم با توجه به نوع و رتبه log محل قرارگیری آن را تعیین کنیم.

```
[n.pardis@lpi boot]$ less /etc/syslog.conf

## Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                                                 /dev/console
*.*                                                 /dev/tty4
#*.*/                                                 /var/log/all-messages
#*.*/                                                 *

#*.*/                                                 *
# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none           /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                           /var/log/secure
#authpriv.*                                         @hadi.myself.org

# Log all the mail messages in one place.
mail.*                                               -/var/log/maillog
mail.1                                              /home/h.kianersi/mail
#mail.[2-6]                                         /home/n.pardis/all-mail-
lo g

# Log cron stuff
/etc/syslog.conf                                     /var/log/cron

cron.*                                               *

# Everybody gets emergency messages
*.emerg                                             *

# Save news errors of level crit and higher in a special file.
uucp,news.crit                                       /var/log/spooler

# Save boot messages also to boot.log
local7.*                                             /var/log/boot.log
(END)
```

فرمت این فایل به این صورت است که معمولاً سمت چپ اسم سرویس قرار دارد و سپس اعداد 0 تا 7 یا * قرار می‌گیرد(این اعداد نشان گر سطح خطاها بود که قبلاً بحث شد) به عنوان مثال در فایل بالا عنوان کرده ایم که برای نامه بر خطاهای رتبه 1 را به h.kianersi ارسال کن. در شهرداری هم که سیصد تا لینوکس داشت هر کسی مسئول log‌های بخش خاصی بود.

* یعنی همه سطح خطاها و [5-2] یعنی خطاهاي سطح 2 الى 5.*. emerg یعنی خطاهاي emergency را برای همه بفرست. اگر خط *. Ra وارد کنید هر دستور و اتفاقی پیش بباید برای همه log ارسال می شود(برای اعمال شدن تغییراتی که در این فایل ایجاد کرده ایم درسیستم، باید دستور service syslog restart را اجرا کنیم.)

```
[n.pardis@lpi ~]$ man syslog.conf
...
Everyone logged on
    Emergency messages often go to all users currently online to
notify
    them that something strange is happening with the system. To
specify
    this wall(1)-feature use an asterisk (*).

EXAMPLES
Here are some example, partially taken from a real existing site
and
configuration. Hopefully they rub out all questions to the
configuration, if not, drop me (Joey) a line.

        # Store critical stuff in critical
        #
        *.=crit;kern.none          /var/adm/critical

This will store all messages with the priority crit in the
file
/var/adm/critical, except for any kernel message.

        # Kernel messages are first, stored in the kernel
        # file, critical messages and higher ones also go
        # to another host and to the console
        #
        kern.*                      /var/adm/kernel
        kern.crit                   @finlandia
        kern.crit                   /dev/console
        kern.info;kern.!err         /var/adm/kernel-info
```

پیغام های critical کرنل به /var/adm/critical می روند و @finlandia اسم هاستس در فنلاند است که اگر کرنل مشکلی داشته باشد ادمینش در فنلاند از آن آگاه می شود، همچنین یک کپی از آن نیز به console ارسال می شود. در خط آخر !err یعنی هر چیزی که نبود (info). می توانیم تغییراتی در این فایل ایجاد کنیم که این خطاهای هر کس که تجربه خوبی در زمینه لینوکس دارد ارسال کنیم.

Log rotating

gmail برای هر نامه ای که ارسال می شود چند خط log نگه می دارد (گیرنده، فرستنده و...) ولی در روز چند میلیون نامه جایه جا می شود. دانشگاه شریف سال 44 افتتاح شده است؛ اگر از آن موقع ورود و خروج ها در یک دفتر ثبت می شد مشکل با یک دفتر 10 هزار صفحه ای هم حل نمی شودا به همین خاطر هر ماه یک دفتر جدید به نگهبانی می دهنده و قبلی ها را در آرشیو حراست نگه داری می کنند. با توجه به اینکه فایل log هر آن در حال بزرگتر شدن می باشد مدیریت آن اجتناب ناپذیر خواهد بود. یکی از ابزار اصلی برای این کار استفاده از logrotating است که با توجه به نیازمان فایل پیکربندی مربوطه را باید تنظیم کنیم. برای مدیریت بهتر بر روی logها بهتر است در مقاطع زمانی مورد نظر، فایل log را بسته و فایل جدیدی را با شماره ی جدید باز نماییم و به همین منظور:

1- تغییرات در /etc/logrotate.conf لازم می باشد.

2- اجرای cron.daily/logrotate بوسیله crond

ضمنا با اجرای logwatch می توانید گزارشات موردنظر را دریافت نمایید.

سوال : دلایل rotate کردن logها را بنویسید.

زمانی که مشکلی پیدا کنیم و بخواهیم از کسی کمک بگیریم باید log file را برای او ایمیل کنیم ولی اگر حجم آن 30 مگا بایت باشد یا هو اجازه ارسال نمی دهد ولی اگر log ها را به قطعات کوچک تر تقسیم کرده بودیم می توانستیم مدیریت بهتری روی آنها داشته باشیم.

دلایل log فایل های rotate

- جلوگیری از بزرگ شدن log file ها
- مدیریت بهتر بر روی وقایع با توجه به شکسته شدن فایل به قطعات کوچکتر
- انتقال و یا کپی سریعتر فایل با توجه به حجم کمتر
- حذف نمودن log های نسبتاً قدیمی

در صورت استفاده از امکانات log rotating خروجی که توسط فرمان ls -l ظاهر می شود در سمت راست شماره خورده است که نمایانگر فایل های log جداگانه برای بازه های زمانی مختلف است.

```

[root@lpi ~]# less /etc/logrotate.conf
# see "man logrotate" for details
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# uncomment this if you want your log files compressed
#compress

# RPM packages drop log rotation information into this directory
include /etc/logrotate.d

# no packages own wtmp -- we'll rotate them here
/var/log/wtmp {
    monthly
    minsize 1M
    create 0664 root utmp
    rotate 1
}
/etc/logrotate.conf
# system-specific logs may be also be configured here.
(END)

```

براساس محتويات ترمinal بالا log ما هر هفته rotate می شود ، 4 تا از log ها (4 هفته) نگه داشته می شود و اگر log وجود نداشته باشد يكی می سازد، log ها را فشرده نکند(کامنت است!) بهتر است قبل از اينکه سیستم روی اين فایل ها overwrite کند آنها را در جایی (rotate) کنيم ؛ اين کار در سازمان های بزرگ مثل شهرداری تهران انجام می شود ، ممکن است شروع و ریشه مشکلی که 3 سال بعد پيش بيايد الان باشد. زير var /تعداد زيادي فایل log است برای همین است که می گويند لينوكس برای سرور بودن مناسب است.دو تا لينوكس هم پيدا نمي کنيد که log هایشان مثل هم باشد چون سرويس ها و اتفاقات متفاوت هستند.

اگر جای log file ها را عوض کردید حتما doc بگذاريد که مسئول بعدی سیستم را پر نکند.

برای دیدن لیست کسانی که userid اشتباه وارد کرده‌اند یا می‌خواستند نفوذ کنند از lastb (last bad) استفاده می‌کنیم:

```

[root@lpi ~]# lastb
n.pardis  ssh:notty      192.168.206.1      Sat Sep  1 04:37 - 04:37
(00:00)
n.pardis  ssh:notty      192.168.206.1      Sat Sep  1 04:36 - 04:36
(00:00)
n.pardis  ssh:notty      192.168.206.1      Sat Sep  1 04:36 - 04:36
(00:00)
(unknown :0

```

```
[root@lpi ~]# man last
...
NOTES
      The files wtmp and btmp might not be found. The system only
logs infor-
      mation in these files if they are present. This is a local
configura-
      tion issue. If you want the files to be used, they can be
created with
      a simple touch(1) command (for example, touch /var/log/wtmp).

FILES
      /var/log/wtmp
      /var/log/btmp
```

در انتهای ترمینال بالا به دو فایل اشاره شده است؛ login که ناموفق در آن ذخیره می‌شوند. این log را معمولاً روی cd ذخیره می‌کنند و هر زمانی که مشکلی پیش بیاید از آنها استفاده می‌شود.

```
[root@lpi ~]# man login
...
FILES
      /var/run/utmp
      /var/log/wtmp
      /var/log/lastlog
      /var/spool/mail/*
      /etc/motd
      /etc/passwd
      /etc/nologin
      /etc/usertty
      .hushlogin
```

این فایل‌ها در سیاست‌گذاری امنیتی برای روز مبادی نگه داری می‌شوند. به عنوان مثال ۵ نفر را به جرم دستکاری یک سند دستگیر کرده بودند. دو نفر می‌گفتند که در آن رور اداره نبوده‌اند و از روی همین فایل‌ها ثابت شد که در روز مذکور این دو نفر اصلاً login نکرده‌اند و تبرئه شدند.

حال سوال این است که آیا این فایل‌ها قابل edit هستند؟ با chattr (که در LPI1 گفته شد) ادمین می‌تواند آنها را تغییر دهد (فایل بازتری است).

```
[root@lpi ~]# cd /var/log/
[root@lpi log]# chattr +a wtmp
```

در ترمینال بالا wtmp را append only کردیم تا کسی نتواند در وسط فایل تغییراتی ایجاد کند.

فعالیت های shell

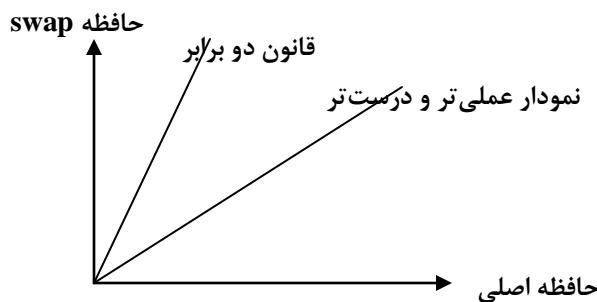
- خواندن etc/profile واجرای آن
- بررسی .bash_profile واجرای home directory
- بررسی وجود چند فایل دیگر
- نمایش prompt ومنتظر خدمت به کاربر
- اجرای bash_logout. در صورت رسیدن خداحافظی
- بازنشسته شدن ومراسم خداحافظی shell

بررسی حافظه مجازی در لینوکس (swap)

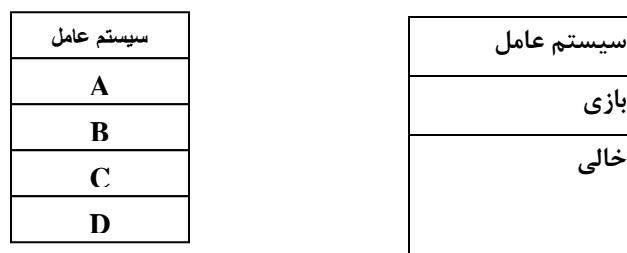
وقتی ویندوز را نصب می کنید به طور پیش فرض فضایی تقریباً دو برابر حافظه رم سیستم را با عنوان pagefile.sys اختیار می کنداگر به هر دلیلی حافظه جا نداشت و نرم افزاری خواست اجرا شود نرم افزار دیگری با اولویت پایین تر از حافظه به این فایل انتقال می یابد و نرم افزار با اولویت بالاتر در حافظه قرار می گیرد ولی در لینوکس فایل نیست بلکه پارتبیشن جدایی برای این منظور در نظر گرفته می شود.

در زمان نصب سیستم عامل لینوکس بایستی فضایی برای نگه داری موقت فرآیندها بر روی دیسک پیش بینی گردد. طبق آمارها و گزارشات و توصیه هایی که در کتاب های راهبردی آمده است، حداقل دو برابر فضای حافظه باید پیش بینی گردد و در صورت اضافه شدن حافظه به کامپیوتر، فضای swap نیز اضافه گردد.

ولی قانون دو برابر حافظه وحی منزل نیست! مورچه می تواند دو برابر وزن خود را حمل کند ولی فیل نمی تواند؛ هر چه قدر فضای حافظه بیش تر می شود به همان مقدار نباید swap بالا برود.



سوال: چرا حافظه swap باید از حافظه اصلی بیش تر باشد.
شاید یک جواب این باشد که حافظه پر است و قرار است نرم افزاری اجرا شود که حافظه زیادی می خواهد. حال فرض کنید رم 16 گیگابایتی داریم و یک بازی اجرا کرده ایم و بقیه حافظه خالی است:



نرم افزار برای اینکه سرویس بگیرد مجبور است fault کند) system call یا event (حال فرض کنید در این بازی گفتم $y=0$ و در خط $z=2/y$ پس از اجرای خط خطا رخ می دهد ؛ سیستم عامل روتینی دارد که چک می کند این خطا متعلق به چه نرم افزاری است اگر متعلق به بازی است چک می کند که آیا بازی می تواند خطا خودش را handle کند؟ اگر می تواند ، کنترل را به خودش می دهد و گرنه این برنامه را کنار می گذارد. خطا همیشه از نرم افزار نیست بعضی مواقع کرنل هم خطا می دهد ؛ وقتی درایور یا سخت افزار مشکل دارد کرنل fault می دهد(عده ترین آن خرابی سخت افزار یا روتین های است که به کرنل اضافه می شود).

حال این خطا را چه کسی باید آنالیز کند؟ حال که خطا از خود سیستم عامل است این نرم افزار(سیستم عامل) به همراه یک سری جدول دیگر روی دیسک قرار می گیرد تا بعداً تحلیل شود(هم در لینوکس و هم در ویندوز این روال وجود دارد). حال سوال این است که این حجم برنامه را در کجای دیسک قرار دهد چون ممکن است فضای کافی وجود نداشته باشد ؛ آن را در swap قرار می دهد که باید بیش تر از فضای حافظه ظرفیت داشته باشد. در واقع یکی از کاربردهای اصلی swap این است که اگر سیستم crash کرد کل برنامه ها روی swap قرار می گیرد. صفحه آبی رنگی که بعد از crash ویندوز مشاهده می کنیم نقطه چین هایی دارد که به ازای هر 4 کیلوبایتی که در pagefile قرار می گیرد یک نقطه به آن اضافه می کند. به خاطر همین است با اینکه سیستم crash کرده تعمیر کار می گوید امروز با اینکه سیستم کار می کند آن را بررسی می کند. ما باید از swap back up بگیریم و آن را به متخصص خودش بسپاریم چون آنالیز crash شدن سیستم کار بسیار سختی است و درست مثل کاری است که پژوهشکی قانونی انجام می دهد.

از دیگر دلایل استفاده از حافظه swap می توان به موارد زیر اشاره کرد:

- کافی نبودن حافظه فیزیکی و اجرای تعداد زیادی فرآیند بر روی کامپیوتر.
- بی کار بودن برنامه ها در حافظه ، بدین معنی که یک فرآیندمنتظر رویدادی می باشد و هسته فرآیند را در فضای مجازی قرار می دهد و به مجرد وقوع رویداد ، فرآیند در صف ورود به حافظه اصلی قرار خواهد گرفت.

```
[n.pardis@lpi ~]$ man crash
```

CRASH(8) CRASH(8)

NAME
crash - Analyze Linux crash data or a live system

SYNOPSIS
crash [**-h** [**opt**]] [**-v**] [**-s**] [**-i file**] [**-d num**] [**-S**] [**mapfile**] [**namelist**] [**dumpfile**]

DESCRIPTION
Crash is a tool for interactively analyzing the state of the Linux system while it is running, or after a kernel crash has occurred and a core dump has been created by the Red Hat netdump, diskdump, kdump, or xendump facilities. It is loosely based on the SVR4 UNIX crash command, but has been significantly enhanced by completely merging it with the gdb debugger. The marriage of the two effectively combines the kernel-specific nature of the traditional UNIX crash utility with the source code level debugging capabilities of gdb.

سیستم را crash نمی کند بلکه اطلاعات crash dump لینوکس را آنالیز می کند؛ داده های crash dump را می خواند و به عنوانمثال می گوید زمانی که سیستم crash کرده 20 نفر on بودند، دو تا فایل بالا بوده، 2 نفر ftp کرده اند و ...

با این آنالیز می توانیم حدس بزنیم که چرا سیستم crash کرده است ولی خیلی کار مشکلی است! دستور crash در خیلی از توزیع ها نیست (در fedora و ubuntu نیست) بیشتر در سرورها وجود دارد. به طور خلاصه اگر swap نداشته باشیم و سیستم crash کند جایی نداریم swap که محتویات حافظه را نگه داریم و بررسی کنیم به همی خاطر کسی که بر روی سیستم خانگی ubuntu نصب کرده، مهم نیست که داشته باشد ولی یک سرور جدی که در بیمارستان قرار دارد crash کرد حتما باید آنالیز شود.

سوال: مگر موقعی که کامپیوتر swap شود restart تعییر نمی کند؟ (overwrite)

باید بلا فاصله بعد از بالا آمدن سیستم با دستور dd که در ادامه توضیح می دهیم swap را dump کنید. در بعضی از توزیع ها ابتدا می پرسد که می خواهید swap در جای دیگری نگه داری شود؟ پس اگر سیستم crash کرد نباید دفعه بعد multi user بالا باید چون file خراب می شود.

اضافه نمودن حافظه موقتی swap

اتفاقی که زیاد رخ می دهد؛ اگر swap پر شود روی کنسول پیغام out of swap space چاپ می شود. یک راه این است که سیستم را reboot کنیم در این صورت مشکل حل می شود چون کاربران سایت را ترک می کنند! خیلی اوقات هم نرم افزارها اجبارا swap می شوند چون از سیستم عامل حافظه زیادی می خواهند. در صورتی که فضای کافی بر روی دیسک سخت برای نگه داری فرآیندهایی که کاندید swap شدن می باشند وجود نداشته باشد، پیغامی از طرف سیستم عامل مبنی بر اخطار اتمام فضای حافظه swap ظاهر می شود. در این شرایط، راهبر می تواند با اضافه نمودن فضای swap به طور موقت مشکل را برطرف نموده و در فرستی مناسب به تنظیم نمودن سیستم اقدام نماید و توصیه می گردد که یک پارتیشن دیگر را به فضای swap اختصاص داده و یا مجددا سیستم را پارتیشن بندی نماید. اول باید بینیم الان چقدر swap داریم، برای نمایش وضعیت swap می توانید فرمان -s را به swapon کار ببرید.

```
[n.pardis@lpi ~]$ swapon -s
Filename      Type            Size        Used       Priority
/dev/sda11    partition     4192924    7344          -1
```

همین طور که در ترمینال بالا مشخص است پارتیشن 13 ام دیسک را برای swap گرفتیم.

```
[n.pardis@lpi ~]$ df
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sda3   29753588 17331508 12114920 59% /
/dev/sda10  9920592  561212  8847312 6% /tmp
/dev/sda9   9920592  3677340  5731184 40% /var
/dev/sda7   24797380 17398792  6118612 74% /opt
/dev/sda6   29753556 4423348  23794396 16% /usr
/dev/sda8   9920592  406280  9002244 5% /usr/local
/dev/sda5   29753556 3924240  24293504 14% /home
/dev/sda1   101086   16965   78902  18% /boot
tmpfs      1032828    0    1032828 0% /dev/shm
/dev/sda2   39674224 29670100  7956240 79% /var/ftp/pub
```

بعضی ها در کتابهایشان گفته اند که swap یک فایل سیستم است (فایل سیستم نحوه چیدمان فایل هاست) ولی swap فایل سیستمی نیست که کاربر روی آن کار کند چون همین طور که در ترمینال بالا مشاهده می کنید کاربر آن را نمی بیند (پارتیشن 13 در لیست وجود ندارد و دستور mount هم آن را نشان نمی دهد) در واقع فایل سیستمی است که بیش تر مورد استفاده کرنل و memory manager است. پس اگر سوال شد چرا یک پارتیشن را df و mount نمایش نمی دهند؟ جواب این است که حتماً آن پارتیشن swap است.

اگر دستور **fdisk -l** را اجرا کنیم جلوی پارتیشن swap نوشته شده linux swap/solaris که فایل با id=82 است (هر فایل سیستمی برای خودش یک علامتی (magic number) دارد).

راجع به swap top صحبت می کند و هر سه ثانیه وضعیت سیستم را نمایش می دهد؛ شامل اینکه چقدر swap گرفته ایم چقدر مصرف شده و چقدر خالی است:

```
[n.pardis@lpi ~]$ top
top - 20:52:12 up 27 days, 6:57, 4 users, load average: 0.02, 0.01, 0.00
Tasks: 79 total, 1 running, 78 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.4%us, 0.3%sy, 0.0%ni, 98.9%id, 0.2%wa, 0.0%hi, 0.1%si, 0.0%st
Mem: 2065656k total, 2020240k used, 45416k free, 381380k buffers
Swap: 4192924k total, 7344k used, 4185580k free, 1520056k cached
```

اگر Swap داشته باشیم خواه ناخواه performance سیستم پایین می آید. لینوکس را که نصب می کنید چندین سرویس ناخواسته بالا می آید مثلا سیستم خانگی اصلا nfs نمی خواهد؛ ادمین باید سرویس های ناخواسته را از سیستم کنار بگذارد.

```
[n.pardis@lpi ~]$ ps -aef|grep kswapd
root      182      7  0 Aug04 ?          00:00:42 [kswapd0]
n.pardis 29271 29215  0 20:52 pts/3        00:00:00 grep kswapd
```

اسم نرم افزاری که وظیفه اش swap می باشد در پایگاه داده، اگر update زیاد باشد نرم افزارهایی که باید اطلاعات را روی دیسک ببرند چندتا می آیند؛ کاغذ زیاد پاره کنیم دو تا سطل آشغال می خواهیم.

تمرین برنامه ای بنویسید که اگر swap به میزان 70 درصد پر شد یک ایمیل به ادمین بنویسد یا روی کنسول چاپ کند.

تمرین اگر چند تا دیسک داشته باشیم swap را چطور توزیع کنیم که balance باشد؟

موقع crash موقع panic اطلاعات را روی هارد می ریزد. حال فرض کنید به swap احتیاج پیدا کردیم (تعداد یا حجم برنامه ها زیاد بود) در این شرایط مثلا تا فرمان date را میزنید در کنسول kill چاپ می کند. موقعی که لینوکس می خواهد نرم افزاری را اجرا کند یک سری checking انجام می دهد که اگر این نرم افزار را وارد حافظه کرد و در حافظه جا نبود می تواند آن را روی دیسک قرار دهد (swap)؟ و اگر swap پرشده باشد kill می دهد.

سوال: یک برنامه قرار است swap شود (یا اولویتیش پایین بوده یا منتظر event است) آیا تمام برنامه swap می شود یا بخشی از آن؟

یک برنامه exe (نه در ویندوز بلکه برنامه ای که permission =x دارد) یک بخش نیست:

```
main()
{
    int x=1;
    char y[100];
    x=12*x;
}
```

یک object نیست ، وقتی برنامه بالا کامپایل شود ، متغیرهایی که مقدار دارند در DATA. آرایه ها در BSS و دستورات در TEXT. ذخیره می شود.

```
[n.pardis@lpi ~]$ size /bin/date
text      data      bss      dec      hexfilename
46390     2588      328    49306  c09a/bin/date
```

دستور date هم سه بخش دارد و مجموع حجم این سه بخش (dec) ، 49 کیلوبایت است.

```
[n.pardis@lpi ~]$ ls -l /bin/cp date
-rwxr-xr-x 1 root root 53124 Jan 21 2009 /bin/date
```

در امتحانات و مصاحبه ها می پرسیم که این نرم افزار executable است ؛ آیا وقتی در حافظه می نشیند 53k جا می گیرد؟ if ,while و جواب مسلما بله نیست! چون یک سری اطلاعات header هم دارد.text را روی دیسک ثابت است نمی توانیم دستورات برنامه (...) را عوض کنیم اما می توانیم متغیرها و آرایه ها را عوض کنیم ولی text هر جای رم که قرار بگیرد هم ویندوز و هم لینوکس آن قسمت از حافظه را به عنوان read only علامت گذاری میکنند. زمانی در دانشگاه شریف سیستم ها آنقدر قدیمی بود که همه جای حافظه می توانستیم read و write کنیم ؛ می گفتیم برنامه ای بنویسید که کدتان را تغییر دهدا! پس اگر برنامه ای قرار باشد swap شود آیا دلیلی دارد که text آن هم swap شود در حالی که نسخه اصل آن روی هارد است و دلیل اصلی اینکه بعضی موقع فلش یا cd را نمی توانیم eject کنیم همین است ؛ نرم افزاری در حافظه دارد که swap شده است حال اگر وسیله را جدا کنیم text را از کجا بیاوریم؟

man size

کسانی که اسمنبلی کار کرده اند این اصطلاحات را می شناسند ؛ در اسمنبل section. تعريف می کنیم حتی بعضی exe ها خودشان comment. دارند که در Ram قرار می گیرد.

مراحل ایجاد حافظه موقتی swap

1. تولید فایلی با اندازه مناسب و ترجیحا تحت /tmp
2. اجرای فرمان mkswap
3. اجرای فرمان swapon

اگر زمانی out of swap space دریافت کردیم و مثلا پایگاه داده در حال update بود یعنی نمی توانیم reboot کنیم با سه دستور فضای موقتی swap ایجاد می کنیم و مثلا آخر هفته که کاربر کم است یک پارتیشن به صورت دائمی اضافه می کنیم.

اول باید یک فایل ایجاد کنیم :

```
[root@lpi ~]# cd /temmp
[root@lpi tmp]# >a
[root@lpi tmp]# ls -l a
-rw-r--r-- 1 root root 0 Aug 31 10:55 a
```

در لینوکس وقتی فایلی ایجاد می کنیم by default سایز آن صفر است و نمی توانیم به عنوان swap (مثلا با حجم 1 گیگابایت) از آن استفاده کنیم. می توانیم با vi یک فایل 1 مگابایتی بسازیم (داخل آن کاراکتر بنویسیم)! و روی بعضی از توزیع های لینوکس دستوری به نام mktemp داریم:

```
[root@lpi tmp]# man mktemp
MKTEMP(1)
```

NAME
mktemp - **make** temporary filename (**unique**)

SYNOPSIS
mktemp [-V] | [-dgtu] [-p directory] [template]

DESCRIPTION
The **mktemp** utility takes the given filename template and overwrites a portion of it to create a unique filename. The template may be any filename with some number of 'Xs' appended to it, for example /tmp/tfile.XXXXXXXXXX. If no template is specified a default of tmp.XXXXXXXXXX is used and the **-t** flag is implied (see below).

The trailing 'Xs' are replaced with a combination of the current process number and random letters. The name chosen depends both on the number of 'Xs' in the template and the number of collisions with pre-existing files. The number of unique filenames **mktemp** can return depends on the number of 'Xs' provided; ten 'Xs' will result in **mktemp** testing roughly 26^{10} combinations.

If **mktemp** can successfully generate a unique filename, the **file** (or **directory**) is created with **file** permissions such that it is only readable and writable by its owner (unless the **-u** flag is given) and the filename is printed to standard output.

هر دفعه یک فایل unique ایجاد می کند در application جدی گزارش ها را زیر فایل های unique قرار می دهد.

```
[root@lpi tmp]# mktemp
/tmp/tmp.ZvUPuJ7988
[root@lpi tmp]# mktemp
/tmp/tmp.AtFtOe7989
```

ولی ما می خواهیم که فایلی درست کنیم که اسم آن ملموس باشد، حجم آن 1 مگابایت باشد و داخل آن چیزی نباشد. در LPI1 گفته شده که /dev/zero چقدر که بخواهیم به ما null می دهد. یک دستور استثنایی است که پیش بینی می شود در آینده ساختار دستورها به این شکل شود. ساختار سنتی به شکل زیر است:

```
[root@lpi tmp]# cp A B
```

```
[root@lpi tmp]# cp B A
```

حال اگر به اشتباه دستور زیر را وارد کنیم:

فایل A از بین می روید. احتمالا در سیستم عامل های جدید دستورات به صورت

```
[root@lpi tmp]# cp INPUT=A OUTPUT=B
```

که اگر جابجا هم شود مشکلی ندارد. الان اکثر دستورات اوراکل به این صورت است. می خواهیم یک فایل 500 مگا بایتی درست کنیم ، خطر این دستور چیست؟

```
[root@lpi tmp]# cp /dev/zero a
[root@lpi tmp]# ls -l a
-rw-r--r-- 1 root root 19730432 Aug 31 11:24 a
```

پس از چند ثانیه از اجرای دستور با `ctrl+c` آن را متوقف کردیم که در عرض همین مدت کوتاه 19 مگابایت از فضای هارد را پر کرد(بستگی به سرعت I/O دارد) پس دقیق نیست و نمی توانیم به درستی حجم فایل را کنترل کنیم. `dd` درست مثل `cp` است ولی ما عنوان می کنیم که چقدر کپی کند:

```
[n.pardis@lpi ~]$ man dd
```

DD(1)	<i>User Commands</i>	DD(1)
--------------	----------------------	--------------

NAME
`dd` - convert and copy a **file**

SYNOPSIS
`dd [OPERAND]...`
`dd OPTION`

DESCRIPTION
Copy a **file**, converting and formatting according to the operands.

bs=BYTES
force **ibs=BYTES** and **obs=BYTES**

cbs=BYTES
convert BYTES bytes at a **time**

conv=CONVS
convert the **file** as per the comma separated symbol list

count=BLOCKS
copy only BLOCKS input blocks

```
[root@lpi tmp]# dd if=/dev/zero of=swap1 bs=1k count=1000
1000+0 records in
1000+0 records out
1024000 bytes (1.0 MB) copied, 0.0565686 seconds, 18.1 MB/s
[root@lpi tmp]# ls -l swap1
-rw-r--r-- 1 root root 1024000 Aug 31 11:35 swap1
```

Input File = if

Output File = of

Block Size = bs

= تعداد ، یعنی مثلا 1000 تا 1 کیلوبایت کپی کن!

دستور سایز فقط برای فایل های exe کاربرد دارد.

خیلی وقتها در اداره ای می گویند که 200 گیگابایت از هارد را کنار بگذارید ؛ کسی استفاده نکند. حالا اگر بخواهیم آشغال داخل آن بروزیم باید از if=/dev/random استفاده کنیم ؛ کنتر است چون cpu باید رشتہ تصادفی تولید کند. زیر /dev/random زیادی فیزیکی و منطقی داریم ؛ این هم یک device است که ورودی و خروجی دارد.

```
[root@lpi tmp]# dd if=/dev/zero of=swap3 bs=1k count=200
200+0 records in
200+0 records out
204800 bytes (205 kB) copied, 0.00428445 seconds, 47.8 MB/s
[root@lpi tmp]# swmkswap swap3
Setting up swapspace version 1, size = 200 kB
[root@lpi tmp]# swapon -a swap3
[root@lpi tmp]# swapon -s
Filename      Type      Size      Used      Priority
/dev/sda11    partition 4192924   7344      -1
/tmp/swap1    file      992       0         -2
/tmp/swap2    file      292       0         -3
/tmp/swap3    file      192       0         -4
```

```
[n.pardis@lpi ~]$ man swapon
```

SWAPON(8)

Linux Programmers Manual

SWAPON(8)

NAME

swapon, swapoff - enable/disable devices and files for paging and swapping

SYNOPSIS

```
/sbin/swapon [-h -V]
/sbin/swapon -a [-v] [-e]
/sbin/swapon [-v] [-p priority] specialfile ...
/sbin/swapon [-s]
/sbin/swapoff [-h -V]
/sbin/swapoff -a
/sbin/swapoff specialfile ...
```

DESCRIPTION

Swapon is used to specify devices on which paging and swapping are to take place.

The device or file used is given by the specialfile parameter. It may be of the form -L label or -U uuid to indicate a device by label or uuid.

در manual نوشته که کاربرد ستون priority این است که اول 1- را بر می کند بعد 2- و حتی می توانید برنامه ای با c بنویسید که فضای swap را زیاد کند و باید از include هایی که در ترمینال زیر لیست شده است استفاده کنید:

```
[n.pardis@lpi ~]$ man -a swapon
SWAPON(2)                               Linux Programmer's Manual      SWAPON(2)

NAME
    swapon, swapoff - start/stop swapping to file/device

SYNOPSIS
    #include <unistd.h>
    #include <asm/page.h> /* to find PAGE_SIZE */
    #include <sys/swap.h>

    int swapon(const char *path, int swapflags);
    int swapoff(const char *path);

DESCRIPTION
    swapon() sets the swap area to the file or block device specified by
    path. swapoff() stops swapping to the file or block device specified
    by path
```

داخل پرانتز دستورات اگر 1 بود فرمان عمومی است اگر 2 بود system call است ، و اگر 3 بود function call ..

```
[n.pardis@lpi ~]$ man -a man
The manual sections are traditionally defined as follows:

1 Commands
    Those commands that can be executed by the user from
    within a shell.

2 System calls
    Those functions which must be performed by the kernel.

3 Library calls
    Most of the libc functions, such as qsort(3).

:
4 Special files
    Files found in /dev.

5 File formats and conventions
    The format for /etc/passwd and other human-readable
    files.

6 Games

7 Conventions and miscellaneous
    A description of the standard file system layout, net-
    work protocols, ASCII and other character codes, this
    man page, and other things.

8 System management commands
    Commands like mount(8), many of which only root can
    execute.

9 Kernel routines
```

زبان برنامه نویسی C دستور printf و scanf تابع هستند.

با -p در swapon می توانیم priority را تعیین کنیم :

```
[root@lpi tmp]# swapon -p 5 swap1
swapon: swap1: Device or resource busy
```

که این کار را باید در زمان ایجاد swap انجام دهیم!

حال می خواهیم یکی از swap ها را حذف کنیم :

```
[root@lpi tmp]# man swapoff
Formatting page, please wait...
SWAPON(8)                               Linux Programmers Manual      SWAPON(8)

NAME
    swapon, swapoff - enable/disable devices and files for paging and swapping

SYNOPSIS
    /sbin/swapon [-h -V]
    /sbin/swapon -a [-v] [-e]
    /sbin/swapon [-v] [-p priority] specialfile ...
    /sbin/swapon [-s]
    /sbin/swapoff [-h -V]
    /sbin/swapoff -a
    /sbin/swapoff specialfile ...

DESCRIPTION
    Swapon is used to specify devices on which paging and swapping are to take place.

    The device or file used is given by the specialfile parameter. It may be of the form -L label or -U uuid to indicate a device by label or uuid.
```

```
[root@lpi tmp]# swapon -s
Filename      Type     Size   Used   Priority
/tmp/swap1    file    192     0    -1
/tmp/swap2    file    192     0    -2
/tmp/swap3    file    192     0    -3
root@lpi tmp]# swapoff swap1
[root@lpi tmp]# swapon -s
/tmp/swap2    file    192     0    -2
/tmp/swap3    file    192     0    -3
```

توصیه می گردد که به فایل /etc/fstab مراجعه نموده و پارتبیشن مربوط به swap را ملاحظه فرمائید. پس در هر زمانی که نیاز به حافظه swap بود با دستوراتی که گفته شد فضای swap را اضافه می کنیم چون همیشه reboot کردن راه حل بدون دردسری نیست. به عنوان مثال زمانی سیستم شهرداری 2 ساعت بعد از reboot بالا نمی آمده چون قبل از آن در حال به روز کردن پایگاه داده بوده است و مثلاً فرض کنید اگر قرار بوده حقوق 100 هزار نفر زیاد شود و قبل از reboot حقوق 50 هزار نفر را زیاد کرده باشد دفعه دوم حقوق 50 هزار نفر اول دو بار افزایش می یابد؛ پایگاه داده مجبور است roll back کند و این کار به همان میزان طول می کشد.

memory management یکی از بخش های پیچیده سیستم عامل است. تصویر زیر حافظه اصلی سیستم را ترسیم می کند که با تعدادی نرم افزار (که اولویت آنها هم در کنارشان نوشته شده است) پر شده است. حال اگر برنامه E بخواهد اجرا شود باید برنامه ای با اولویت پایین تر از اولویت E (7) از حافظه به swap منتقل شود؛ اولویت D از همه کم تر است و از حافظه کنار گذاشته می شود. اگر باز هم فضای کافی برای E وجود نداشته باشد C هم به swap منتقل می شود. حال اگر D خارج شود با توجه به اینکه اولویتش از همه پایین تر است؛ چه زمانی دوباره در حافظه قرار می گیرد؟

یک زمانی در کلاس جای نشستن نیست می گوییم هر کس خارج از کلاس بایستد و به درس گوش کند یک نمره تشویقی می گیرد و از هر کس داخل کلاس نشسته باشد یک نمره کسر می شود؛ سیستم عامل هم در هر دور یکی به اولویت D اضافه می کند و یکی از برنامه های داخل حافظه کم می کند تا D هم دوباره بتواند اجرا شود.

بحاطر همین است که لینوکس پیشنهاد می کند همه نرم افزارها با اولویت یکسان وارد حافظه شوند.

سیستم عامل	
A	16
B	11
C	4
D	3
E	7

اولویت ها به طور مرتب عوض می شوند:

```
[root@lpi tmp]# ps -alef|less
F S UID          PID  PPID  C PRI  NI ADDR SZ WCHAN  STIME TTY          TIME CMD
4 S root          1     0  0 75   0 -  516 -  20:46 ?          00:00:01 init [3]

1 S root          2     1  0 -40  - -    0 migrat 20:46 ?          00:00:00
[migration/0]
1 S root          3     1  0 99  19 -    0 ksoftti 20:46 ?          00:00:00
[ksoftirqd/0]
5 S root          4     1  0 -40  - -    0 watchdog 20:46 ?          00:00:00
[watchdog/0]
1 S root          5     1  0 70  -5 -    0 worker  20:46 ?          00:00:00
[events/0]
1 S root          6     1  0 71  -5 -    0 worker  20:46 ?          00:00:00
[khelper]
5 S root          7     1  0 72  -5 -    0 worker  20:46 ?          00:00:00
[kthread]
1 S root         10     7  0 79  -5 -    0 worker  20:46 ?          00:00:00
[kblockd/0]
1 S root         11     7  0 80  -5 -    0 worker  20:46 ?          00:00:00
[kaclpid]
1 S root        175     7  0 77  -5 -    0 worker  20:46 ?          00:00:00
```

در لینوکس غیر از nice مفهوم دیگری به نام priority را کم و زیاد کنیم نرم افزارها از حافظه خارج یا وارد آن می شوند:

```
[n.pardis@lpi ~]$ man nice
NICE(1)                               User Commands                               NICE(1)

NAME
    nice - run a program with modified scheduling priority

SYNOPSIS
    nice [OPTION] [COMMAND [ARG]...]

DESCRIPTION
    Run COMMAND with an adjusted niceness, which affects process scheduling. With no COMMAND, print the current niceness. Nicenesses range from -20 (most favorable scheduling) to 19 (least favorable).

    -n, --adjustment=N
        add integer N to the niceness (default 10)

    --help display this help and exit

    --version
        output version information and exit
```

بعضی مواقع می گویند این نرم افزار خیلی مهم است و شما باید renice را اجرا کنید. برای این کار با root دستور top را اجرا کنید و سپس کلید r را فشار دهید که pid و مقدار درخواستی برای اولویت آن از شما پرسیده می شود. مثلا در زیر اولویت init را به -19 تغییر داده ایم:

```
[root@lpi ~]$ top
  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 5043 root      15   0  2192 1064  832 R  0.3  0.1  0:00.10 top
    1 root      15   0  2064  652  560 S  0.0  0.0  0:01.12 init
    2 root      RT -5   0     0   0 S  0.0  0.0  0:00.00 migration/0
    3 root      34  19   0     0   0 S  0.0  0.0  0:00.00 ksoftirqd/0
PID to renice: 1
Renice PID 1 to value: -19
[root@lpi ~]$ top
  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 3424 root      25   0  9360 1156  904 S  0.1  0.1  0:00.15 automount
 5043 root      25   0  2192 1064  832 R  0.1  0.1  0:00.13 top
    1 root      0 -19  2064  652  560 S  0.0  0.0  0:01.12 init
    2 root      RT -5   0     0   0 S  0.0  0.0  0:00.00 migration/0
```

سوال چه اقدامی انجام دهیم که اگر سیستم reboot شد این swap file باقی بمانند؟

اگر زیر /etc فایل ها را درست کنیم دفعه بعد فایل ها وجود دارند ولی کاری انجام نمی دهد در ضمن در زمان backup گیری این فایل عظیم نیز ذخیره می شود. اگر یک اسکریپت (همان سه خط دستور) بنویسیم و در rc3.d ذخیره کنیم؛ شاید دفعه بعد سیستم با 2 بالا آمد! داخل inittab می توانیم mkswap را قرار دهیم ولی در عرض یک ثانیه تمام می شود و inittab می خواهد آن را دوباره اجرا کند (با این وجود می توانیم once بگذاریم و درست است اما توصیه می شود که زیاد فایل های سیستمی را تغییر ندهید!).

فایل /etc/rc.local بعد از همه اسکریپت های init اجرا می شود. به عنوان مثال گفتیم وقتی سیستم بالا آمد دو تا ip 172.16.11 و دیگری برای فایل سیستم.

```
[root@lpi ~]# less /etc/rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.
touch /var/lock/subsys/local

ifconfig eth0 192.168.100.1
ifconfig eth0:1 172.16.1.1
```

حال سوال این است که اگر قرار است دستورات ساخت فایل swap را اینجا وارد کنیم آیا لازم است که dd را هم وارد کنیم؟

فایل های تحت /tmp/ طبق سیاست گذاریهای سازمان ممکن است هر روز یا هر هفته پاک شود (یا اصلاً پاک نشود) به خاطر همین در اسکریپتمن قید می کنیم که اگر فایل swap وجود نداشت یکی بسازد:

```
[root@lpi ~]# vi /etc/rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.
touch /var/lock/subsys/local

ifconfig eth0 192.168.100.1
ifconfig eth0:1 172.16.1.1
if [ ! -f /tmp/swap1 ]
then
dd if=/dev/zero of=/tmp/swap1 bs=1k count=512
fi
mkswap /tmp/swap1
swapon -a /tmp/swap1
```

تمرین برنامه ای بنویسد که اگر 80 درصد swap پر شد یک ایمیل به admin بدهد.

برای به دست آوردن درصد باید used size را برابر cut استفاده کنیم و برای استخراج این فیلدها از swapon -s ، از

```
[root@lpi ~]# man cut
CUT(1)                               User Commands                               CUT(1)

NAME
    cut - remove sections from each line of files

SYNOPSIS
    cut [OPTION]... [FILE]...

DESCRIPTION
    Print selected parts of lines from each FILE to standard output.

    Mandatory arguments to long options are mandatory for short options
    too.

    -b, --bytes=LIST
        select only these bytes

    -c, --characters=LIST
        select only these characters

    -d, --delimiter=DELIM
        use DELIM instead of TAB for field delimiter
```

```
[root@lpi ~]# x=`swapon -s|grep swap2|cut -f3`  
[root@lpi ~]# y=`swapon -s|grep swap2|cut -f4`  
[root@lpi ~]# echo $x $y  
192 0  
[root@lpi ~]# z=$((x/y))  
bash: x/y: division by 0 (error token is "y")
```

X برابر با سایز swap و y برابر با used است. باید با شما (به عنوان ادمین) تماس بگیرند که swap پر شده است! اسکریپتی بنویسید که شما را مطلع کند و ادامه آن به عنوان تمرین به دانشجو و اگذار می شود. در صورتی که سیستم عامل swap گردد down فایل موقت تولید شده نیز از بین خواهد رفت و بهتر است که دو خط زیر را تحت فایل /etc/rc.local قرار دهیم تا پس از بالا آمدن سیستم یک swap file موقت تولید گردد.

- swap/tmp/swap1
- Mkswapon/tmp/swap1

چه طور از swap بگیریم؟

با cp هم می توان این کار را انجام داد ولی بهتر است فایل های خاصی که زیر دایرکتوری dev/ قرار دارد را با dd جایه جا کنیم.

```
[root@lpi ~]# dd if=/dev/swap of=/dev/tmp ...
```

بررسی روش های محدود کردن کاربران در لینوکس (Limitation)

خطور می توان کاربران را محدود کرد که نتوانند فایل های بزرگ درست کنند یا چندین فایل باز کنند. اکثر ISP ها اجازه نمی دهند دو تا login با یک userID داشته باشید. یا اینکه از یک سری IP ها نتوانید متصل شوید. اگر محدودیت نباشد کاربران می توانند 50 تا n! اجرا کنند و سیستم را down کنند. در خیلی از سازمان ها و بانک ها برنامه های گرافیکی برای اعمال این محدودیت ها نوشته شده است که این فایل ها را تغییر می دهند. سیستم عامل لینوکس این امکان را به راهبر می دهد که محدودیت های عمومی و یا برای کاربران خاصی اعمال کند که اهم محدودیت های خاص (برای userID) عبارتند از:

- تعداد login ها
- تعداد پروسس ها
- فضای گرفته شده در حافظه
- اتصال از هر نقطه ای
- زمان اجرا
- به کار گرفتن سرویس

```
[n.pardis@lpi ~]$ cd /etc/security
[n.pardis@lpi security]$ ls
access.conf      console.handlers    group.conf      namespace.conf
pam_env.conf     chroot.conf        console.perms     limits.conf    namespace.init
pam_winbind.conf console.apps       console.perms.d   limits.d      opasswd          time.conf
```

برای اعمال محدودیت هم خودتان می توانید این فایل ها را ویرایش کنید هم اینکه با جاوا ، c++ یا shell برنامه هایی بنویسید که این فایل ها را تغییر دهند.

مثال 1: تغییراتی در سیستم دهید که کاربر student فقط بتواند یک login داشته باشد.

```
#<domain>      <type>  <item>           <value>
#
#*
#*          soft    core      0
#*          hard    rss       10000
#@student    hard    nproc     20
#@faculty    soft    nproc     20
#@faculty    hard    nproc     50
#ftp         hard    nproc     0
#@student    -       maxlogins 1
```

تفسیر خط آخر که اضافه کرده ایم این است که می تواند داشته باشد حداکثر یکی است. حال اگر برای دومین بار با login student کنیم (در حالی که هنوز از اولی خارج نشده ایم) با پیغام خطای زیر مواجه می شویم:

```
[n.pardis@lpi ~]$ su ps -aef|grep student
root      3696  1  0 Aug31 ?          00:00:00 login -- student
n.pardis  6105  6068  0 01:45 pts/0    00:00:00 grep student
[root@lpi security]# su student
Too many logins for 'student'.
could not open session
```

مثال 2 : تغییراتی در سیستم دهید که کاربر student فقط بتواند حداکثر 6 پروسس همزمان را اجرا نماید. با فرمان & می توان چندین پردازش همزمان را در پشت پرده¹ انجام داد که مثلا یکی از آنها می تواند ساختن یک فایل یک تراپایتی با دستور dd باشد یا اینکه یک سنگین در پایگاه داده داشته باشیم.

```
[n.pardis@lpi ~]$ sleep 10&
[n.pardis@lpi ~]$ sleep 10&
[2] 6125
[n.pardis@lpi ~]$ sleep 10&
[3] 6126
[n.pardis@lpi ~]$ jobs
[1] Done                      sleep 10
[2]- Running                  sleep 10 &
[3]+ Running                  sleep 10 &
[n.pardis@lpi ~]$ jobs
[2]- Done                      sleep 10
[3]+ Done                      sleep 10
```

¹ background

با دستور jobs می توانید کارهای پشت پرده را مشاهده کنید.(علامت + و - به چه معنی هستند؟)

می خواهیم کاری کنیم که student نتواند تعداد زیادی(بیش از 6) پردازش پشت پرده داشته باشد:

```
[root@lpi security]$ vi limits.conf
#<domain>      <type>   <item>          <value>
#
#*
#*           soft    core     0
#*           hard    rss     10000
#@student    hard    nproc    20
#@faculty    soft    nproc    20
#@faculty    hard    nproc    50
#ftp         hard    nproc    0
#@student    -       maxlogins 1
@student     hard    nproc     6
# End of file
```

```
[student@lpi ~]$ sleep 100&
[1] 6362
[student@lpi ~]$ sleep 100&
[2] 6363
[student@lpi ~]$ sleep 100&
[3] 6364
[student@lpi ~]$ sleep 100&
[4] 6365
[student@lpi ~]$ sleep 100&
[5] 6366
[student@lpi ~]$ sleep 100&
-bash: fork: Resource temporarily unavailable
[student@lpi ~]$ date
-bash: fork: Resource temporarily unavailable
[student@lpi ~]$ cd /etc
[student@lpi etc]$ help cd
cd: cd [-L|-P] [dir]
...
[student@lpi etc]$ man date
popen: Resource temporarily unavailable
[n.pardis@lpi etc]$ exit
logout
-bash: fork: Resource temporarily unavailable
```

قبل اگفتیم که برای اجرای یک دستور shell خودش را کپی می کند (fork) ولی دفعه ششم می گوید که موقتاً^۱ منابع نداریم. در این حالت همانطور که در بالا مشاهده می کنید دستور date اجرا نمی شود ولی cd اجرا می شود چون cd built-in است و خود bash آن را اجرا می کند ولی date یک دستور خارجی است که برای اجرای آن bash باشد. به همین ترتیب fork شود. به همین ترتیب help را هم می توانیم ببینیم ولی نمی توانیم man را اجرا بگیریم. به سادگی هم نمی توانیم exit کنیم احال سوال این است که در فایل limits.conf حداقل چند تا پردازش می توانیم قرار دهیم یعنی کم ترین تعداد پردازشی که یک کاربر می تواند داشته باشد چندتاست؟ (که در LPI1 .bash_profile توضیح دادیم) یک اسکریپت است که بلا فاصله بعد از login اجرا می شود و خودش یک shell می خواهد پس اگر حداکثر پردازش ها را

^۱ temporarily

دوتا قرار دهیم prompt می دهد ولی چون .bash_profile را چاپ می کند. اگر بخواهیم محدودیت هایی را برای تعدادی از کاربران (که مثلا نام کاربری آنها با kaaramuz شروع می شود) اعمال کنیم میتوانیم از امکانات فایل wildcard مانند limits.conf استفاده کنیم که به روشنی در توضیحات این فایل گفته شده است.

مثال ۳ : تغییراتی در سیستم دهید که کاربر student فقط بتواند فایل هایی با حداکثر 200 کیلوبایت تولید نماید.

```
[n.pardis@lpi ~]$ cp /dev/zero test
File size limit exceeded
[n.pardis@lpi ~]$ [n.pardis@lpi ~]$ ls -l
-rw-rw-r-- 1 n.pardis n.pardis 204800 Sep 1 04:42 test
[n.pardis@lpi ~]$ ls -l -h
-rw-rw-r-- 1 n.pardis n.pardis 200K Sep 1 04:42 test
```

مطلوب مریوط به محدود نمودن کاربر نسبتا مفصل بوده و فقط در مورد محدودیت های تعداد Login و پروسس و .. مطالب زیادی وجود دارد که اسامی آنها در ادامه می آید و برای به دست آوردن اطلاعات بیشتر باید مطالب مریوط به PAM و محدودیت ها دقیقا مطالعه گردد. ضمنا فرمان ulimit نیز محدودیت هایی را برای کاربر به وجود می آورد و مطالعه manual آن قویا توصیه می شود. اگر راهبر تغییراتی در کاربر بددهد، معمولا کاربر نمی تواند تغییرات را خنثی ویا بی اثر نماید. محدودیتهای کاربر n.pardis را در زیر مشاهده می کنید (که size را قبل محدود کردیم):

```
[n.pardis@lpi root]$ ulimit
200
[n.pardis@lpi root]$ ulimit -a
core file size          (blocks, -c) 0
data seg size            (kbytes, -d) unlimited
scheduling priority      (-e) 0
file size                (blocks, -f) 200
pending signals          (-i) 24576
max locked memory        (kbytes, -l) 32
max memory size          (kbytes, -m) unlimited
open files               (-n) 1024
pipe size                 (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority        (-r) 0
stack size                (kbytes, -s) 10240
cpu time                  (seconds, -t) unlimited
max user processes        (-u) 24576
virtual memory             (kbytes, -v) unlimited
file locks                 (-x) unlimited
```

محدودیت های root را در زیر مشاهده می کنید:

```
[root@lpi ~]# ulimit
unlimited
[root@lpi ~]# ulimit -a
core file size          (blocks, -c) 0
data seg size            (kbytes, -d) unlimited
scheduling priority      (-e) 0
file size                (blocks, -f) unlimited
pending signals          (-i) 24576
max locked memory        (kbytes, -l) 32
max memory size          (kbytes, -m) unlimited
open files               (-n) 1024
pipe size                (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority        (-r) 0
stack size                (kbytes, -s) 10240
cpu time                 (seconds, -t) unlimited
max user processes        (-u) 24576
virtual memory             (kbytes, -v) unlimited
file locks                  (-x) unlimited
```

با ulimit تغییراتی ایجاد می کنیم که cpu time ، 1 ثانیه باشد:

```
[n.pardis@lpi ~]$ ulimit -t 1
[n.pardis@lpi ~]$ ulimit -a
core file size          (blocks, -c) 0
data seg size            (kbytes, -d) unlimited
scheduling priority      (-e) 0
file size                (blocks, -f) 200
pending signals          (-i) 24576
max locked memory        (kbytes, -l) 32
max memory size          (kbytes, -m) unlimited
open files               (-n) 1024
pipe size                (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority        (-r) 0
stack size                (kbytes, -s) 10240
cpu time                 (seconds, -t) 1
max user processes        (-u) 24576
virtual memory             (kbytes, -v) unlimited
file locks                  (-x) unlimited
```

برای اینکه این محدودیت ها را برای چند کاربر اعمال کنیم باید limits.conf را تغییر دهیم.

فاکتوریل 5 درصد cpu را گرفته است در حالی که بقیه cpu ، cpu idle است ؛ دلیل آن کندی tcp/ip (شبکه) است. یک برنامه با سرعت کنترین device خود کار می کند. بک زمانی در دانشگاه شریف پروژه ها cpu limit داشت و باید برنامه ها را efficient می نوشتیم که زودتر اجرا شود.

```
[n.pardis@lpi ~]$ ulimit -f 50
[n.pardis@lpi ~]$ cp /dev/zero test2
File size limit exceeded
[n.pardis@lpi ~]$ ls -l -h test2
-rw-rw-r-- 1 n.pardis n.pardis 50K Sep 1 08:43 test2
```

همان طور که وقتی در صفحه ایستاده اید می توانید عقب تر بروید (نوبتتان را به یک فرد مسن بدھید) ولی نمی توانید جلو بروید در لینوکس هم به عنوان userID معمولی می توانید از محدودیت خودتان کم کنید ولی نمی توانید به آن اضافه کنید مگر اینکه logoff کنید.

```
[n.pardis@lpi ~]$ ulimit -f 50000
-bash: ulimit: file size: cannot modify limit: Operation not permitted
```

همانطور که در ترمینال –a ulimit مشاهده کردید حداکثر پردازش ها (max user processes) 24576 تاست. یک دانشجوی ماهر می تواند دانشگاه را به هم بریزد ، نه اینکه سیستم crash کند بلکه آنقدر پروسس اجرا کند که swap پرشود و پیغامی روی کنسول به نمایش درآید.در شهرداری از سال 76 ، سرور لینوکس استفاده شده و بیش تر از 3000 userID به این سیستم ها متصل بوده اند. محدودیت ها در جلسات مسئولین بررسی می شد و گزنه ممکن بود یک update مهم به دلیل اتمام محدودیت زمانی استفاده از cpu قطع شود و در rollback هم به تبع همین اتفاق بیفت. به همین خاطر امور فنی در دست Admin است و سیاست گذاری ها توسط مسئولین انجام می شود. فایل های دیگر تحت دایرکتوری /etc/security limits.conf دارند و برای اعمال تغییرات نیز manual کاملی دارند. هر کاربری از هرجا نتواند به سیستم متصل شود:

```
[n.pardis@lpi security]$ less access.conf
# Login access control table.
#
# Comment line must start with "#", no space at front.
# Order of lines is important.
#
# When someone logs in, the table is scanned for the first entry that
# matches the (user, host) combination, or, in case of non-networked
# logins, the first entry that matches the (user, tty) combination. The
# permissions field of that table entry determines whether the login will
# be accepted or refused.
#
# Format of the login access control table is three fields separated by a
# ":" character:
#
# [Note, if you supply a 'fieldsep=' argument to the pam_access.so
# module, you can change the field separation character to be
# '|'. This is useful for configurations where you are trying to use
# pam_access with X applications that provide PAM_TTY values that are
# the display variable like "host:0".]
#
#     permission : users : origins
```

هر کاربری در هر زمان نتواند از سیستم سرویس بگیرد:

```
[n.pardis@lp1 security]$ less time.conf
# this is an example configuration file for the pam_time module. Its syntax
# was initially based heavily on that of the shadow package (shadow-960129).
#
# the syntax of the lines is as follows:
#
#       services;ttys;users;times
#
# white space is ignored and lines maybe extended with '\\n' (escaped
# newlines). As should be clear from reading these comments,
# text following a '#' is ignored to the end of the line.
#
# the combination of individual users/terminals etc is a logic list
# namely individual tokens that are optionally prefixed with '!' (logical
# not) and separated with '&' (logical and) and '|' (logical or).
#
# services
#       is a logic list of PAM service names that the rule applies to.
#
# ttys
#       is a logic list of terminal names that this rule applies to.
#
# users
#       is a logic list of users or a netgroup of users to whom this
#       rule applies.
```

هر کاربری نتواند از طریق کنسول وارد سیستم شود:

```
#!/bin/bash
# version 0.1 very simple and needs many modification..
#date 1389/12/15
swapsize=`tail -1 /proc/swaps | cut -f2`
swapused=`tail -1 /proc/swaps | cut -f3`

if [ $swapused != 0 ]

then
eightypct=$((4*swapsize/5))
if [ $swapused -gt $eightypct ]
then
pct=$((100*swapused/swapsize))
echo "warning: check swap partition($pct% used)" | mail -s swap root
logger "Warning:check swap partition,$pct% used"
fi
fi
```

کد زیر ورژن 0.1 تمرینی است که در جلسه قبل گفته شد (یعنی بسیار ابتدایی است و نیاز به تغییرات زیادی دارد). در این جلسه یاد می گیریم که چطور این اسکریپت را اتوماتیک کنیم.

```

[n.pardis@lpi security]$ less console.perms
# /etc/security/console.perms
#
# This file determines the permissions that will be given to privileged
# users of the console at login time, and the permissions to which to
# revert when the users log out.

# format is:
#   <class>=list of regexps specifying consoles or globs specifying files
#   file-glob|<class> perm dev-regex|<dev-class> \
#       revert-mode revert-owner[.revert-group]
# the revert-mode, revert-owner, and revert-group are optional, and default
# to 0600, root, and root, respectively.
#
# For more information:
# man 5 console.perms
#
# This file should not be modified.
# Rather a new file in the console.perms.d directory should be created.

# file classes -- these are regular expressions
<console>=tty[0-9][0-9]* vc/[0-9][0-9]* :[0-9] .[0-9] :[0-9]
<xconsole>=: [0-9] .[0-9] :[0-9]

```

یک نرم افزار اگر executable باشد خودش کاملا مستقل اجرا می شود ولی اگر مثل این کد ، اسکریپت باشد سیستم نمی داند چه نرم افزاری را وارد حافظه کند تا این اجرا شود (؟) موقعی که خط اول را می خواند و !# را می بیند متوجه می شود که این نرم افزار را چه کسی باید اجرا کند پس اگر شما برنامه را با shell نوشتهید خط اول اختیاری است و گرنه در خط اول باید آدرس مفسری را که باید کد را تفسیر کند قرار دهید (اکثر نرم افزارهای سیستم خط اول را دارند.

Swaps دستوری است که فایل /proc/swaps را می خواند:

```

[root@lpi tmp]# cat /proc/swaps
Filename      Type      Size    Used     Priority
/dev/dm-1    partition 2064376 892      -1
/tmp/swap1    file        992      0      -2

```

در اسکریپتی که ما نوشته ایم فقط یک فایل swap را بررسی می کند و اگر در سازمانی چندتا فایل swap داشته باشیم باید آن را تغییر دهیم.(تمرین) در ادامه با cut سایز را برداشته ایم. چرا f2 زده ایم در حالی که سومین فیلد سایز است؟ دلیل این است که به صورت پیش فرض جداگانه در tab cut است و همان طور که در فایل زیر مشخص است پس از filename فاصله قرار دارد و به همین دلیل اسم فایل نوع آن را با هم یک فیلد حساب می کند.

```
[root@lpi tmp]# swapon -s|od -c|less
0000000 F i l e n a m e \t \t \t T y p e
0000020 \t \t S i z e \t U s e d \t \t \t P r i o
0000040 r i t y \n / d e v / d m - 1
0000060
0000100
0000120 t i t i o n \t 2 0 6 4 3 7 6 \t a r o
0000140 \t - 1 \n / t m p / s w a p 1
0000160
0000200
0000220 \t \t 9 9 2 \t 0 \t - 2 \n f i l e
0000233
(END)
```

درون فایل را نشان می‌دهد.

هر tab روی صفحه نمایش به اندازه 5 تا ستون فضا اشغال می‌کند ولی در حافظه به اندازه یک کاراکتر جا می‌گیرد. در ادامه چک کردیم که اگر swapused غیرصفر است محاسبه کند که اگر بیش تراز 80 درصد شده، آن را محاسبه کند و در نهایت آن را در قالب \$pct در کنسول چاپ کند و سپس یک میل با موضوع swap به Root ارسال کند و حتی اگر ایمیل نزد با logger کنسول log کند. اگر تعداد فایل‌ها زیاد باشد یک راه بهتر شاید استفاده از محتویات فایل meminfo باشد:

```
root@lpi tmp]# cat /proc/meminfo
MemTotal:      1030888 kB
MemFree:       512636 kB
Buffers:        88284 kB
...
SwapTotal:     2065368 kB
SwapFree:      2065368 kB
```

با این فایل کار راحت‌تر است و نیازی به محاسبات ریاضی نیست:

```
[root@lpi tmp]# cat /proc/meminfo |grep SwapFree
SwapFree:      2065368 kB
```

تمرین: دستور بالا را سبک‌تر کنید طوری که cat را به grep نکنیم. (optimization)

The Cron System

نحوه اجرای اتوماتیک برنامه ها در لینوکس

برنامه ای که برای چک کردن فضای استفاده شده swap نوشته شد که هر نیم ساعت خودش در پشت پرده اجرا گردد. برای اینکه بخواهیم دستوری در زمان خاصی اجرا شود می توانیم از at استفاده کنیم:

```
[root@lpi tmp]# man at
AT(1)                               Linux Programmer's Manual                  AT(1)

NAME
    at, batch, atq, atrm - queue, examine or delete jobs for later execution

SYNOPSIS
    at [-V] [-q queue] [-f file] [-m ldbv] TIME
    at [-V] [-q queue] [-f file] [-m ldbv] -t time_arg
    at -c job [job...]
    atq [-V] [-q queue]
    atrm [-V] job [job...]
    batch

DESCRIPTION
    at and batch read commands from standard input or a specified file
    which are to be executed at a later time.

    at      executes commands at a specified time.

    atq     lists the user's pending jobs, unless the user is the superuser; in that case, everybody's jobs are listed. The format
           of the output lines (one for each job) is: Job number, date,
```

البته دستوری مثل batch کمکی به ما نمی کند چون موقعی که cron سیستم پایین باشد اجرا می شود. cron مخفف load time و معنای زمان سنج می باشد و از آن می توان برای ایمیل logها برای ادمین و گرفتن backup در زمان های معین استفاده کرد. مثلا سیستم هر شب cpio بزند و همه لینوکس را روی هارد اکسترنال قرار دهد. کاربرد دیگر برای حذف فایل هاست ؛ در /tmp/ فایل های متفرقه زیادی وجود دارد و ممکن است پس از مدتی پر شود. خیلی از سایت هایی که جهانی نیست جمعه ها اوایل بامداد اقدام به پاک کردن محتويات /tmp/ میکنند ولی پاک سازی سایتی مثل گوگل خیلی پیچیده تر از اينهاست (تمام جستجوهايی که شما انجام می دهيد زير /tmp/ ذخیره می شود)

```
cd /tmp
rm *
```

پس آیا اتوماتیک کردن دستورات زیر کار درستی است ؟

نیاید زیر /tmp فایل جدی باشد چون کسی از آن backup نمی‌گیرد و گرنه یک سری که در حال کار هستند فایل هایشان پاک می‌شود (علامت زده می‌شود که بعداً پاک شود) با این حال * rm به صلاح نیست:

```
[root@lpi tmp]# lsof |wc -l  
6113
```

الان 6113 فایل باز شده داریم که اکثر آنها کتابخانه است. پس اگر خواستید با یک اسکریپت زیر tmp را پاک کنید lsof می‌کنیم اونایی که زیر tmp هست را کنار می‌گذاریم برای زمانی که busy نیستند:

```
[root@lpi tmp]# lsof |grep tmp  
...
```

bash	3083	n.pardis	cwd	DIR	253,0	4096	259587 /tmp
su	3330		root cwd	DIR	253,0	4096	259587 /tmp
bash	3337		root cwd	DIR	253,0	4096	259587 /tmp
lsof	6034		root cwd	DIR	253,0	4096	259587 /tmp
grep	6035		root cwd	DIR	253,0	4096	259587 /tmp
lsof	6036		root cwd	DIR	253,0	4096	259587 /tmp

استفاده دیگر اتوماتیک کردن در حذف پردازش‌های غیر ضروری است. گفتیم in.telnetd چک می‌کند اگر کاربر وصل نبود پردازش مربوط به آن را از بین می‌برد ولی همه سرویس‌ها از این روش استفاده نمی‌کنند مثلاً اوراکل همچین مسئله‌ای دارد. به همین خاطر ممکن است پس از چند روز سیستم کند شود و با reboot درست شود اما پس از مدتی باز هم سیستم کند می‌شود. زمانی در شهرداری ده هزار کاربر داشتیم، روز عاشر 6 زدیم ps را بودند اسکریپتی نوشتم که اگر برنامه‌ای 48 ساعت در سیستم بود آن را کنار بگذار pss را گرفتیم اگر تاریخ بود یعنی 24 ساعت گذشته، بعد process time را چک می‌کنیم، سپس چک می‌کنیم که idle است یا خیر؟ یکی از وظایف لینوکس کارها این است که هر کاری که تکراری بود را باید اتوماتیک کنند. نرم افزاری که این کارها را انجام می‌دهد cron است. این سرویس موقعی که سیستم بالا می‌آید، باید وارد شود. top نرم افزارهای پرمشغله را نشان می‌زند و cron این طور نیست. پس برای اینکه ببینیم بالا آمده یا نه باید زیر /var/run/را ببینیم. دستور دیگری هم هست که ادمین استفاده می‌کند:

```
[root@lpi tmp]# service crond status  
crond (pid 2171) is running...
```

مولفه‌های cron

- /usr/sbin/crond
- /etc/crontab
- /usr/bin/crontab
- /var/spool/cron
- /etc/cron.d
- /etc/cron.allow & /etc/cron.deny

یک سرویس است که تعداد زیادی فایل دارد و باید با آنها کار کند. از پایین شروع می کنیم: دو تا فایل دارد که تعیین می کند چه کسانی می توانند و چه کسانی نمی توانند. ممکن است کاربری عمدتاً یا سهوا سیستم را مشغول کند.

```
[n.pardis@lpi Desktop]$ man crond
```

CRON(8) Cronie Users Manual CRON(8)

NAME
cron - daemon to execute scheduled commands

SYNOPSIS
cron [-n | -p | -s | -m<mailcommand>]
cron -x [ext,sch,proc,pars,load,misc,test,bit]

DESCRIPTION
Cron should be started from /etc/rc.d/init.d or /etc/init.d

Cron searches /var/spool/cron for crontab files which are named after accounts in /etc/passwd; The founded crontabs are loaded into memory. Cron also searches for /etc/anacrontab and the files in the /etc/cron.d directory, which are in a different format (see crontab(5)). Cron examines all stored crontabs, checking each command to see if it should be run in the current minute. When executing commands, any output is mailed to the owner of the crontab (or to the user named in the MAILTO environment variable in the crontab, if such exists). Job output can also be sent to syslog by using the -s option.

مرتب چک می کند؛ فایل هایش را نگاه می کند و ساعت را چک می کند که آیا موعد کارها رسیده یا نه. برای بالا آوردن سرویس crond از دستور زیر استفاده کنید:

```
[n.pardis@lpi Desktop]$ service crond start
```

البته اگر می خواهید مستقل از توزیع عمل کنید بهتر است از دستور زیر استفاده کنید:

```
[n.pardis@lpi Desktop]$ /etc/init.d/crond start
```

فایل crontab شامل تعدادی رکورد می باشد و crond با استفاده از اطلاعاتی که از این فایل به دست می آورد نرم افزارهای موردنظر را اجرا می نماید. معمولاً توزیع کننده های Linux/Unix اطلاعات از پیش تعیین شده ای را در این فایل قرار می دهند.

```

[n.pardis@lpi Desktop]$ cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# For details see man 4 crontabs

# Example of job definition:
# ----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .---- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .--- day of week (0 - 6) (Sunday=0 or 7) OR
# | | | | sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed

```

در قسمت shell مشخص کرده‌ایم که چه کسی باید این دستورات را اجرا کند و سپس گفته‌ایم شل از کجا بگردد و دستور را پیدا کند و اگر مشکلی پیش آمد به چه کسی باید اطلاع بدهیم. path بعضی موقع در performance برنامه خیلی تاثیرگذار است مخصوصاً در سیستم‌های سنگین که مسیرهای زیادی وجود دارد اینکه مسیری را در ابتدا قراردهیم که احتمال وجود دستور در آن بیشتر است، در سرعت سیستم خیلی موثر است. باید با history آمار بگیریم که کدام دستورها استفاده بیشتری دارند و مسیر آنها را در اولویت قرار دهیم. سرویس‌هایی مثل Crond معمولاً استخوان‌بندی محکمی دارند و در طی این چند سال تغییر چندانی نکردند بلکه بهینه شده‌اند. فایل crontab شامل بخش‌های زیر می‌باشد:

- خطوط توضیح (comments)
- تنظیم و یا مقدار دهنده متغیرهای محیطی
- فرمان‌های cron

در ضمن این فایل به صورت متنی (text) بوده و قابل تغییر و یا تصحیح می‌باشد.

تمرین 1: صورت مسئله این است که در بیمه استخدام شده‌ایم و باید هر جمعه در ماه دسامبر ساعت 3/5 از سیستم backup بگیریم. یک راه این است که خودمان در موعد مقرر این کار را انجام دهیم که البته کار درستی نیست و بهتر این است که در فایل crontab یک خط برای آن درست کنیم. فرمت فرمان cron به صورت زیر می‌باشد:

minute	hour	day_of_month	month	day_of_week	command
30	15	*	12	5	/usr/local/bin/backup

* یعنی همه روز خاصی از ماه مدنظر ما نیست. جمعه روز پنجم هفته است (شروع هفته از دوشنبه است و فارسی نشده است)

تمرین 2 هر شب ساعت 11/5 backup بگیرد:

```
30 23 * * * /bin/cpio
```

تمرین 3 هر دقیقه یک ایمیل بفرستیم:

```
* * * * * /bin/mail
```

تمرین 4 می خواهیم دقیقه اول ، ششم و بیست و ششم ایمیل بفرستیم:

```
1,6,26 * * * * /bin/mail
```

با کمک اتوماتیک کردن می توان سایت هایی را ساپورت کرد بدون اینکه نیازی به چک کردن آنها باشد. حتی اگر برق قطع شود سرویسی به نام anacron که کارها را در بالا آمدن سیستم اجرا می کند:

```
setup>system services
```

```
[n.pardis@lpi ~]$ man anacron
```

ANACRON(8)	Anacron Users Manual	ANACRON(8)
NAME		
anacron - runs commands periodically		
SYNOPSIS		
anacron [-s] [-f] [-n] [-d] [-q] [-t anacrontab] [-S spooldir] [job]		
anacron [-S spooldir] -u [-t anacrontab] [job]		
anacron [-V -h]		
anacron -T [-t anacrontab]		
DESCRIPTION		
Anacron is used to execute commands periodically, with a frequency specified in days. Unlike cron(8), it does not assume that the machine is running continuously. Hence, it can be used on machines that aren't running 24 hours a day, to control regular jobs as daily, weekly, and monthly jobs.		
Anacron reads a list of jobs from a configuration file, /etc/anacrontab (see anacrontab(5)). This file contains the list of jobs that Anacron controls. Each job entry specifies a period in days, a delay in minutes, a unique job identifier, and a shell command.		

برای به اطلاعات بیش تر در مورد فایل contab به آن مراجعه کنید:

```
[n.pardis@lpi ~]$ man cron5 5 crontab
ANACRONTAB(5)           Cronie Users Manual          ANACRONTAB(5)

NAME
    crontab - tables for driving cron (ISC Cron V4.1)

DESCRIPTION
    A crontab file contains instructions to the cron(8) daemon of the general form: "run this command at this time on this date". Each user has their own crontab, and commands in any given crontab will be executed as the user who owns the crontab. Uucp and News will usually have their own crontabs, eliminating the need for explicitly running su(1) as part of a cron command.

    Blank lines and leading spaces and tabs are ignored. Lines whose first non-space character is a pound-sign (#) are comments, and are ignored. Note that comments are not allowed on the same line as cron commands, since they will be taken to be part of the command. Similarly, comments are not allowed on the same line as environment variable settings.

    An active line in a crontab will be either an environment setting or a cron command. An environment setting is of the form,
```

/usr/bin/crontab

کاربر می تواند با دسترسی به cron system فعالیت های زیر را انجام دهد:

- بررسی ، تجدیدنظر و تغییر در crontab
- تهیه لیستی از cron jobs
- حذف crontab

فعالیت ها فقط می توانند بر روی crontab مربوط به خود کاربر باشد و فقط root مجاز می باشد که به همه کاربران دسترسی داشته باشد.

crontab فرمان

- crontab –e به منظور ویرایش اقلام فایل

- crontab -r به منظور حذف نمودن اقلام فایل
- crontab -l به منظور مشاهده فایل
- crontab -u user ... فقط برای کاربر ریشه

کاربر به وسیله فرمان crontab می تواند فایل مربوط به خودش را تغییر دهد یا حذف نماید و کاربر root نیز می تواند تغییرات لازمه را روی فایل کاربر اعمال کند. با اجرای دستور زیر وارد محیط ویرایشگر vi می شویم (گاهی اوقات مجبور به استفاده از vi هستیم):

```
[n.pardis@lpi ~]$ crontab -e
```

و یا کاربر root می تواند با اجرای فرمان زیر تغییرات لازمه را بر روی فایل کاربر بدهد:

```
[root@lpi ~]# crontab -u n.pardis -e
```

ادمین می تواند با دستور زیر بفهمد که چه user هایی چه crond را اجرا کرده اند ولی این کار از لحاظ اخلاقی و حرفة ای درست نیست!

```
[root@lpi ~]# crontab -l -u n.pardis
17 23 01 10 01 date
```

```
[root@lpi ~]# crontab -l
17 23 01 10 01 date
```

کارها در دایرکتوری /var/spool/cron در فایلی به نام کاربر وجود دارد:

```
[root@lpi ~]# cd /var/spool/cron/
[root@lpi cron]# ls -l
total 4
-rw-----. 1 root root 20 Oct 1 23:15 root
```

جالب اینجاست که شما به عنوان کاربر نمی توانید در این دایرکتوری فایل ایجاد کنید ولی در اینجا فایل هایی به نام شما وجود دارد که خودتان با دستور crontab درست کرده اید:

```
[n.pardis@lpi mail]$ type -a crontab
crontab is /usr/bin/crontab
[n.pardis@lpi mail]$ ls -l /usr/bin/crontab
-rwSr-xr-x. 1 root root 46680 Mar 4 2011 /usr/bin/crontab
```

این فایل s permision دارد. خیلی خطناک است؛ شل نتوانست وارد آن دایرکتوری شود ولی crontab اجازه s دارد و قدرت root را دارد، می‌تواند در هر دایرکتوری فایل بسازد. اگر این اجازه را برداریم کاربر دسترسی به crontab ندارد:

```
[root@lpi ~]# chmod 755 /usr/bin/crontab
[root@lpi ~]# ls -l /usr/bin/crontab/crontab
-rwxr-xr-x. 1 root root 46680 Mar 4 2011 /usr/bin/crontab
[n.pardis@lpi root]$ crontab -l
/var/spool/cron/n.pardis: Permission denied
```

کسانی که در data center ها یا سایت‌های جدی کار می‌کنند نباید به نرم افزاری s permision بدهیم. پنتاگون کتابی برای کشورهای عضو ناتو دارد که به آنها توصیه می‌کند اگر می‌خواهید نرم افزاری نصب کنید که اجازه s می‌خواهد حتماً سورس آن را بررسی کنید که رخنه‌ای در آن نباشد و سپس کامپایل کنید.

4775 دوباره اجازه s را می‌دهد. تا 777 برای کاربرهای معمولی است. از 1000 به بالا Admin است.

تمرین:

هر چه فایل s permision در سیستم دارید را چاپ کنید. یکی از کارهای ادمین همین است که روزی دو دفعه (در قید کند) که اگر هکری نفوذ کرد و s permision اضافه شد ادمین مطلع شود. نصب لینوکس که تمام شد خروجی این دستور را در فایلی ذخیره می‌کنیم و سپس در crontab قید می‌کنیم که روزی دو بار فایل‌های s permision را استخراج کرده و با دستور dif تفاوت این فایل و فایل اولیه را به دست می‌آوریم و اگر فایلی اضافه شده باشد ادمین را خبر می‌کند. در ویندوز کسی نمی‌داند چطور بعضی از دستورها می‌توانند با فایل‌های سیستمی کار کنند. مثل اینترنت اکسپلورر که با سیستم عامل عجین است و همه فایلهایش را می‌تواند ببینند.

/etc/cron.d

این دایرکتوری شامل فایل‌هایی می‌باشد که در زمان نصب بسته‌های نرم افزاری (package) مانند mrtg و sysstat تولید شده‌اند.

```
[root@lpi ~]# cd /etc/cron.d
[root@lpi cron.d]# ls -l
total 20
-rw-r--r--. 1 root root 113 Mar 4 2011 0hourly
-rw-r--r--. 1 root root 170 Jun 27 2011 mailman
-rw-r--r--. 1 root root 139 Jun 29 2010 mrtg
-rw-r--r--. 1 root root 108 Mar 28 2011 raid-check
-rw-r--r--. 1 root root 245 Mar 31 2011 sysstat
```

اینجا اکثرا یا command است یا اسکریپت است. mrtg یک نرم افزار monitoring است که مرتباً باید پشت صحنه اجرا شود.

```
[root@lpi cron.d]# cat mrtg
*/5 * * * * root LANG=C LC_ALL=C /usr/bin/mrtg /etc/mrtg/mrtg.cfg --
lock-file /var/lock/mrtg/mrtg_1 --confcache-file
/var/lib/mrtg/mrtg.ok
```

این نرم افزار هر 5 دقیقه یک بار اجرا می شود و ترافیک router را به صورت یک منحنی رسم می کند.

```
[root@lpi cron.d]# cat mailman
# DO NOT EDIT THIS FILE!
#
# Contents of this file managed by /etc/init.d/mailman
# Master copy is /usr/lib/mailman/cron/crontab.in
# Consult that file for documentation
```

گفته که این فایل را نباید دست بزنید. این فایل توسط نرم افزار نامه بر که با up شدن سیستم بالا می آید تست و ویرایش می شود. به یاد داشته باشد که وقتی mrtg را نصب می کنید خودش این فایل را در cron.d ایجاد می کند (در rpm ، lpi1 را درس داده ایم) ولی اگر زمانی گفته شد که mrtg سیستم را کند کرده بدانید که در اینجا تنظیم شده که هر 5 دقیقه اجرا شود.

- /etc/cron.allow & /etc/cron.deny

این فایل ها توسط راهبر مقدار گرفته و مجوز استفاده و یا عدم استفاده از امکانات cron را با قرار دادن userid در یکی از فایل های فوق صادر می گردد. به عنوان مثال اگر مقدار usera در فایل cron.deny قرار گرفته شود ، کاربر usera تخواهد توانست از امکانات سیستم cron بهره برداری نماید.

```
[root@lpi etc]# cat cron.deny
[root@lpi etc]#
```

حالی است ؟ همه مجازند cron کنند!

سوال: چرا دو تا فایل (deny و allow) داریم (یکی کافی نبود)؟

به عنوان مثال زمانی که در کلاس یک نفر حاضر است و زمانی که یک نفر غایب است ما به دو طریق می توانیم برخورد کنیم.

برنامه هایی که قرار است هر ساعت اجرا گردد ، می توانند تحت دایرکتوری /etc/cron.hourly قرار بگیرند (درست نیست در cron اصلی نوشته شوند) و به صورت پیش فرض این فایل محتویاتی ندارد.

```
[n.pardis@lpi etc]$ ls -l cron.hourly/
total 4
-rwxr-xr-x. 1 root root 424 Mar  4 2011 0anacron
```

```
[n.pardis@lpi etc]$ cat cron.hourly/*
#!/bin/bash
#in case file doesn't exist
if test -r /var/spool/anacron/cron.daily; then
    day=`cat /var/spool/anacron/cron.daily`
fi
if [ `date +%Y%m%d` = "$day" ]; then
    exit 0;
fi

# in case anacron is already running,
# there will be log (daemon won't be running twice).
if test -x /usr/bin/on_ac_power; then
    /usr/bin/on_ac_power &> /dev/null
    if test $? -eq 1; then
        exit 0
    fi
fi
/usr/sbin/anacron -s
```

برنامه هایی که قرار است هر روز یا اصطلاحاً روزانه اجرا شوند می توانند تحت دایرکتوری /etc/cron.daily قرار بگیرند و به صورت پیش فرض این دایرکتوری شامل فایل های زیر است:

```
[ n.pardis@lpi:/etc/cron.daily time=17:50:52 command number=3871 ]

>ls
00webalizer      certwatch     makewhatis.cron   rpm
0anacron          cups          mlocate.cron    squirrelmail.cron
0logwatch         cyrus-imapd  prelink        tetex.cron
asterisk_cleanup logrotate    rkhunter      tmpwatch
```

در کلاس شل گفته می شود که نرم افزاری به نام khunter داریم که یک شل اسکریپت چندهزار خطی است و وظیفه اش این است که نرم افزارهای محل را پیدا می کند.

```

[ n.pardis@lpi:/etc/cron.daily time=17:50:56 command number=3872 ]

>less rkhunter

#!/bin/sh
# 01-rkhunter A shell script to update and run rkhunter via CRON
export LANG="en_US.UTF-8" # possible bugfix for SCRIPTDIR error message

XITVAL=0

# Get a secure tempfile
TMPFILE1=`/bin/mktemp -p /var/run/rkhunter rkhcronlog.XXXXXXXXXX` || exit 1

if [ ! -e /var/lock/subsys/rkhunter ]; then

    # Try to keep the SysInit boot scan from colliding with us (highly unlikely)
    /bin/touch /var/lock/subsys/rkhunter

    # Source system configuration parameters.
    if [ -e /etc/sysconfig/rkhunter ] ; then
        . /etc/sysconfig/rkhunter
    else
        MAILTO=root@localhost
    fi

```

این نرم افزار مثل آنتی ویروس روزی یک بار update می‌شود.

```

[root@lpi share]# type rkhunter
rkhunter is /usr/bin/rkhunter
[root@lpi share]# less /usr/bin/rkhunter

#!/bin/sh

#
# rkhunter -- Scan the system for rootkits and other known security issues.
#
# Copyright (c) 2003-2012, Michael Boelen ( michael AT rootkit DOT nl )
#
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111, USA.

```

```
[root@lp1 ~]# rkhunter -c
[ Rootkit Hunter version 1.4.0 ]

Checking system commands...

Performing 'strings' command checks
  Checking 'strings' command[ OK ]

Performing 'shared libraries' checks
  Checking for preloading variables[ None found ]
  Checking for preloaded libraries[ None found ]
  Checking LD_LIBRARY_PATH variable[ Not found ]

Performing file properties checks
  Checking for prerequisites[ Warning ]
  /sbin/chkconfig[ OK ]
  /sbin/depmod[ OK ]
  /sbin/fsck[ OK ]
  /sbin/fuser[ OK ]
  /sbin/ifconfig[ OK ]
  /sbin/ifdown[ Warning ]
  /sbin/ifup[ Warning ]
  /sbin/init[ OK ]
```

این نرم افزار با دستور strings همه سرویس ها و نرم افزارها را بررسی می کند که ویروس نباشند. rkhunter روی لینوکس نصب نیست باید دانلود کنید.

```
[ n.pardis@lp1:/etc/cron.daily time=22:54:30 command number=3924 ]

>cat rpm
#!/bin/sh

tmpfile=`/bin/mktemp /var/log/rpmpkgs.XXXXXXXXXX` || exit 1
/bin/rpm -qa --qf '%{name}-%{version}-%{release}.%{arch}.rpm\n' 2>&1 \
| /bin/sort > "$tmpfile"

if [ ! -s "$tmpfile" ]; then
rm -f "$tmpfile"
exit 1
fi

/bin/mv "$tmpfile" /var/log/rpmpkgs
/bin/chmod 0644 /var/log/rpmpkgs
```

```
[ n.pardis@lp1:/etc/cron.daily time=22:54:37 command number=3925 ]

>cat logrotate
#!/bin/sh

/usr/sbin/logrotate /etc/logrotate.conf
EXITVALUE=$?
if [ $EXITVALUE != 0 ]; then
  /usr/bin/logger -t logrotate "ALERT exited abnormally with [$EXITVALUE]"
fi
exit 0
```

می توانیم برنامه swapf را که برای ساخت فایل swap موقتی نوشنیم در فایل cron.hourly قرار دهیم که هر ساعت چک کند که اگر وضع swap بد بود فایل swap اضافه کند (تعداد تکرار را می توان با تجربه فهمید) برنامه هایی که می خواهیم هر هفته اجرا شود در فایل cron.weekly قرار می دهیم که به طور پیش فرض شامل anacron و makewhatis است.

```
[root@lpi etc]# man makewhatis
MAKEWHATIS(8)                                         MAKEWHATIS(8)

NAME
makewhatis - Create the whatis database

SYNOPSIS
makewhatis [-u] [-v] [-w] [-s sections] [-c [catpath]] [manpath]

DESCRIPTION
makewhatis reads all the manual pages contained in the given sections
of manpath or the preformatted pages contained in the given sections of
catpath. For each page, it writes a line in the whatis database; each
line consists of the name of the page and a short description, sepa-
rated by a dash. The description is extracted using the content of the
NAME section of the manual page.

Since other languages use a different term for the NAME section, make-
whatis recognizes the equivalent terms in Czech, Italian, Finnish,
French, German and Spanish.

If no manpath argument is given, /usr/man is assumed by default.

OPTIONS
```

دستور apropos از کجا اطلاعات می آورد و می خواند ؟ manual همه makewhatis را می خواند و دسته بندی می کند. الان اگر نرم افزاری را نصب کنید apropos اطلاعات آن را نمی تواند پیدا کند چون بعد از نصب باید makewhatis بزند که manual را به داخل پایگاه داده منتقل کنید. همین خاطر تنظیم می کنیم که هفته ای یک بار makewhatis اجرا شود و اگر هر روز نصب و update دارید آن را روزانه اجرا کنید. روش اشتباہی که رایج است؛ ما لینوکس نصب می کنیم، هر وقت که به مشکل برخورد دنبال چاره و درمان می گردیم. کار درست این است که با امکاناتی مثل cron از مشکلات پیش گیری کنیم. برنامه هایی که هفتگی است چه روزی اجرا می شود؟ آنهایی که روزانه است چه ساعتی؟

```
[ n.pardis@lpi:~ time=22:53:41 command number=3921 ]

>cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

مثلا روزانه ها ، هر روز ساعت 4 و دو دقیقه اجرا می شود. powershell در ویندوز خیلی قوی است و عمدتا system call است . سیاست مایکروسافت همین است که ویندوز مثل bmw آنقدر بسته است که خراب نمی شود ، اگر هم خراب شودهیچ کس نمی تواند آن را درست کند. ولی لینوکس مثل پیکان! ؛ پیچ و مهربه هایش معلوم است. ویندوز ساپورت می کند ولی لینوکس ساپورت نمی کند اما بهترین ویندوز کارها اول در لینوکس خبره بوده اند چون مکانیزم ها یکسان است.

تمرین چه کار کنیم که از صبح تا ظهر کسی نتواند cron اجرا کند؟ راه حل این است که از ترکیب cron و limitation استفاده کنید.

core file

```
[root@lpi etc]# ulimit -a
core file size          (blocks, -c) 0
data seg size            (kbytes, -d) unlimited
scheduling priority      (-e) 0
file size                (blocks, -f) unlimited
pending signals          (-i) 7937
max locked memory        (kbytes, -l) 64
max memory size          (kbytes, -m) unlimited
open files               (-n) 1024
pipe size                (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority        (-r) 0
stack size                (kbytes, -s) 10240
cpu time                 (seconds, -t) unlimited
max user processes        (-u) 1024
virtual memory            (kbytes, -v) unlimited
file locks               (-x) unlimited
```

```
[root@lpi etc]# ulimit help ulimit
ulimit: ulimit [-SHacdefilmnpqrstuvx] [limit]
Modify shell resource limits.
```

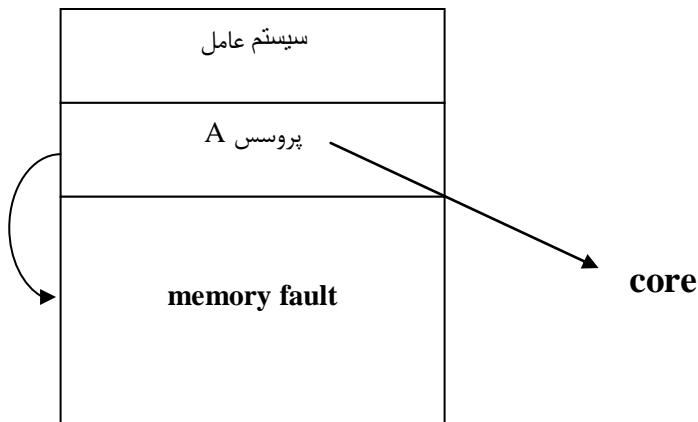
Provides control over the resources available to the shell and processes it creates, on systems that allow such control.

Options:

- Suse the 'soft' resource limit
- Huse the 'hard' resource limit
- aall current limits are reported
- bthe socket buffer size
- cthe maximum size of core files created
- dthe maximum size of a process's data segment
- ethe maximum scheduling priority ('nice')

```
main()
{
int x=123456;
char x[5];
y[x]=a;
}
```

اگر کد بالا مربوط به برنامه A باشد . این برنامه segmentation fault می دهد . کامپایلر جایی را به برنامه و متغیرهایش در رم اختصاص می دهد به مجرد اینکه cpu ، y[x] را حساب می کند از حوزه برنامه بیرون می رود. در هر cpu و سیستم عاملی ، بخواهد دستوری را انجام دهد چک میکند اگر از حوزه خودش خارج شد fault می دهد.



اگر نرم افزاری به هر دلیلی (بیرون افتادن از حافظه یا تقسیم بر صفر) در حین اجرا خطأ بدهد سیستم عامل به آن برنامه می گوید تو خطا کردی و باید از بین بروی . پس از آن کاربر از کجا بفهمد که مشکل چه بوده است؟ هم ویندوز و هم لینوکس این کار را انجام می دهند که زیر دایرکتوری جاری برنامه را به جای این که دور بریزد در غالب core file ذخیره می کند. حتما در ویندوز دیدید که بعضی مواقع می پرسد send/don't send . به بعضی از نرم افزارها اگر ctrl+send ارسال کنید هم از بین می روند و هم core dump می دهند ولی فقط ctrl+c می دهد:

[n.pardis@lpi ~]\$ mna an -a signal			
Signal	Value	Action	Comment
SIGHUP	1	Term	Hangup detected on controlling terminal or death of controlling process
SIGINT	2	Term	Interrupt from keyboard
SIGQUIT	3	Core	Quit from keyboard
SIGILL	4	Core	Illegal Instruction
SIGABRT	6	Core	Abort signal from abort(3)
SIGFPE	8	Core	Floating point exception
SIGKILL	9	Term	Kill signal
SIGSEGV	11	Core	Invalid memory reference
SIGPIPE	13	Term	Broken pipe: write to pipe with no readers
SIGALRM	14	Term	Timer signal from alarm(2)
SIGTERM	15	Term	Termination signal
SIGUSR1	30,10,16	Term	User-defined signal 1
SIGUSR2	31,12,17	Term	User-defined signal 2
SIGCHLD	20,17,18	Ign	Child stopped or terminated
SIGCONT	19,18,25	Cont	Continue if stopped

```
[n.pardis@lpi ~]$ ulimit -c 40000
[n.pardis@lpi ~]$ ulimit -a
core file size          (blocks, -c) 40000
...

```

وسط ctrl+\ . cat می زنیم تا core تولید شود:

```
[n.pardis@lpi ~]$ cat
hi
hi
hello
hello
^ \Quit (core dumped)
[n.pardis@lpi ~]$ ls -l
total 84
-rw-----. 1 n.pardis n.pardis 290816 Oct  9 22:23
core.20702
```

قبلای که آن صفر بود core تولید نمی شد و دلیل صفر بودن هم اینکه در عرض دو روز دیسک سرور پر میشد. یکی از وظایف ادمین این است که هر هفته core file را دور بریزد. در ضمن اگر در خانه خودتان نباشید core نمی گیرید.

سوال: فکر می کنید عددی که به انتهای اسم core چسبانده شده چه معنی دارد؟

است که اگر نچسباند و یک نرم افزار چند بار core بدهد هم نام می شوند.

ادمین های حرفه ای باید core بخوانند البته باید حداقل سه سال سابقه کار داشته باشند.

```
[n.pardis@lpi ~]$ file core.2*
core.20702: ELF 32-bit LSB core file Intel 80386, version 1 (SYSV), SVR4-
style, from 'cat'
```

گفته core موردنظر از یک برنامه (sysv) exe است به نام .cat

```
[root@lpi home]# find -name "core.2" -exec rm -i {} \;
lsfind -name "core" -exec rm -i {} \;
lsfind . -name "core" -exec rm -i {} \;/*
rm: remove regular file './n.pardis/core.20702'? y
```

می خواهیم inform کند i- می گذاریم و گرنه بدون سوال همه را پاک می کند. {} یعنی چیزی که پیدا کردی به جای این قرار بده!

می توان file core tmp را به منتقل کرد.

```
[root@lpi home]# find . -name "core*" -exec mv {} /tmp/allcores \;
```

برای load کردن cpu نیست چون در کلاس lan کرده اید و گرنه با adsl این دستور سرعت چندانی ندارد.

امنیت در لینوکس

```
[n.pardis@lpi ~]$ man gpg  
GPG2(1)                               GNU Privacy Guard                               GPG2(1)  
  
NAME  
      gpg2 - OpenPGP encryption and signing tool  
  
SYNOPSIS  
      gpg2 [--homedir dir] [--options file] [options] command [args]  
  
DESCRIPTION  
      gpg2 is the OpenPGP part of the GNU Privacy Guard (GnuPG). It is a tool  
      to provide digital encryption and signing services using the OpenPGP  
      standard. gpg2 features complete key management and all bells and whis-  
      tles you can expect from a decent OpenPGP implementation.  
  
      In contrast to the standalone version gpg, which is more suited for  
      server and embedded platforms, this version is installed under the name  
      gpg2 and more targeted to the desktop as it requires several other mod-  
      ules to be installed. The standalone version will be kept maintained  
      and it is possible to install both versions on the same system. If you  
      need to use different configuration files, you should make use of some-  
      thing like gpg.conf-2 instead of just gpg.conf.
```

یک فایل متنی را با `gpg` کد کنید، آیا اندازه آن بزرگتر می شود؟ آن را `less` کنید.

به عنوان مثال سیستم پسورد در یک جای نظامی به صورتی بود که یک رشته کاراکتر می داد و کاربر باید با الگوریتمی که از قبل حفظ کرده بود آن رشته را تغییر می داد و از آن به عنوان کلمه عبور استفاده می کرد و هر دفعه این رشته عوض می شد به این ترتیب احتمال فاش شدن پسورد خیلی کم بود. یک مسئله قدیمی راجع به امنیت این است که مثلاً شما می خواهید سکه طلایی را در یک صندوقچه قفل شده برای کسی بفرستید ولی چون به پستچی اعتماد ندارید نباید کلید آن را با درسته قرار دهید!

راه حل این است که شما صندوقچه قفل شده را ارسال می کنید؛ در مقصد گیرنده یک قفل دیگر که فقط خودش کلید آن را دارد در کنار قفل شما روی صندوقچه قرار می دهد و صندوقچه را دوباره برای شما ارسال می کند. شما قفل خودتان را باز می کنید و صندوقچه را (که فقط قفل دوم یعنی قفل متعلق به گیرنده روی آن قرار دارد) دوباره برای گیرنده ارسال می کند؛ گیرنده هم که کلید قفل خودش را دارد صندوقچه را به راحتی باز می کند.

فاجعه و بازیابی

فاجعه چیست؟

بهم ریختنگی سیستم های حیاتی کامپیوتری ، به طوری که بسیاری و یا کلیه فعالیت های یک مرکز کامپیوتری دچار اختلال گردد. به عنوان مثال ، خراب شدن دیسک سخت یک سیستم عملیاتی در اطاق عمل که فایل پشتیبانی از آن وجود نداشته و کلیه فعالیت های مربوط به عمل بیمار که بخشی از آن توسط کامپیوتر کنترل می گردید (کنترل نبض ، فشارخون ...) دچار اختلال می گردد.

بازیافت چیست؟

مجموعه فعالیت هایی که بتوان فشارهای ناشی از فاجعه را محدود نموده و سیستم را در مسیر بهبودی هدایت نموده تا در حداقل زمان ممکن بتوان وضعیت را عادی نمود. به عنوان مثال تعویض دیسک سخت و برگرداندن فایل پشتیبان مربوط به شب گذشته که تا حدی از مشکلات خواهد کاست ، هر چند که اطلاعاتی که بعد از آخرین فایل پشتیبان وارد شده است دیگر وجود ندارد.... بروز فاجعه فقط با خراب شدن دیسک سخت به وجود نمی آید و ممکن است بلایای طبیعی مانند سیل و زلزله و رعد و برق و بلایای دیگر مانند آتش سوزی و سرقت باعث بروز فاجعه گردد. سوالی که مطرح می گردد این است که راهبر مرکز کامپیوتر در زمان بروز این گونه اشکالات چه کند.

- آیا دستورالعملی برای این گونه مواقع به او داده شده است؟
- آیا می داند با کجا بایستی تماس بگیرد؟
- آیا شماره تلفن مراکز مهم (مثل 125) را می داند؟
- آیا شماره تلفن سرپرست مستقیم و یا مدیران ذیربخط خود را دارد؟
- آیا می داند که فایل پشتیبان در کجا قرار دارد؟
- آیا می داند با کدام ISP تماس بگیرد؟
- آیا به او به اندازه کافی قدرت داده شده است که تصمیم گیری نماید؟
- و سوالاتی در این زمینه ...

در ایران فقط ایرانسل اینها را دارد که اگر مشکلی پیش آمد با چه کسانی باید تماس بگیرد. حتی بعضی جاها backup روی سرور گذاشته بودند یا زیرآبدار خانه که خراب شده بود. فاجعه چگونه به وجود می آید؟

- اشکال در برق 72 درصد
- اشکال سخت افزاری 52 درصد
- اشکال شبکه 46 درصد
- اشکال نرم افزار 43 درصد
- خطای انسانی و حملات هکرها ۲۰٪ درصد

دسته بندی فاجعه ها

حیاتی (vital) که کلاس 3 نیز خوانده می شود

سیستم کاملا خوابیده و اگر سیستم در عرض چند دقیقه عملیاتی نگردد امکان بروز بحران قطعی می باشد (به عنوان مثال کامپیوتر بانک جهانی که تمام داد و ستدتها از طریق آنجا انجام می گردد)

و خیم (critical) که کلاس 2 نیز خوانده می شود

سیستم خوابیده و اگر در عرض چند ساعت (کم تر از یک روز) راه اندازی نشود امکان بروز بحران می رود (مثل سازمان آب که کنترل فشار آب از طریق کامپیوتر مرکزی انجام می گیرد)

مهم (important) که کلاس 1 نیز خوانده می شود

سیستم خوابیده و اگر در عرض حدود یک هفته راه اندازی نشود امکان بروز بحران می رود (به عنوان مثال اگر سیستم رزرو یک هتل کوچک به هم بریزد در دراز مدت صاحب هتل ورشکست خواهد شد)

چگونه فاجعه را به حداقل برسانیم؟

- محل قرار گرفتن سیستم های کاربردی را دقیقاً بدانیم.(هارد و درایو و پارتیشن)
- کار سیستم های کاربردی را تا حدی بدانیم.
- محل نگه داری فایل پشتیبان را (در دو محل مجزا) دقیقاً بدانیم.

مثلا سیستم های حساس را در دو طرف خط زلزله قرار می دهیم. فایل های پیکربندی باید کپی شود (علاوه بر داده ها). در سازمانی که اتوماسیون دفتری دارد حتما فایل های conf را هم کپی بگیرید. البته این در حد کمال هم اتفاق نمی افتد مثلا 500 هزار نامه در گوگل گم شد و دلیل آن هم معلوم نیست.

- محل قرار گرفتن داده ها را بر روی سیستم دقیقاً بدانیم.
- محل قرار گرفتن نرم افزارهای پایه را بدانیم (سیستم عامل ، بانک اطلاعاتی و وصله ها (patches) و کپی فایل های پیکربندی)
- اطمینان کامل از صحت فایل های پشتیبان داشته باشیم.

حتما cpi را با -t (test) امتحان کنید. در ضمن در سایت های دولتی backup روی DVD معنی ندارد ؛ فقط سیستم املاک شهرداری 48GB حجم دارد و روی tape ذخیره می شود. یک بار در شهرداری منطقه 11 سیستم crash کرد و کل دیسک خرد شد فایل پشتیبان را که قرار دادیم خطای seize error داد (روی نوار زیگ زاگ خورده بود) یک بار هم در مخابرات موقع پشتیبان گیری taper تکان خورد و مجبور شدیم با اسمبلی کدی بنویسیم که کنترلر tape drive اطلاعات را بخواند.

- سیستم عامل و بانک های اطلاعاتی را بر روی پارتیشن جدا و حتی بر روی دیسک های سخت مجزا قرار دهیم. به همین خاطر است که الان از دیسک SCSI پرظرفیت 2TB استفاده نمی کنیم (مثال 4 تا 500GB قرار می دهیم).
- تمام مدارک لازمه (کاغذی) را در جائی محفوظ (گاوصندوقق) داشته باشیم.

- فایل README برای هر کدام از سیستم‌ها داشته باشیم.

- هر نرم افزاری را قبل از تست دقیق و کامل بر روی خادم اصلی نصب ننمائیم.

در یکی از بانک‌ها ادمین ساعت نماز نرم افزاری را نصب کرده و reboot کرده سیستم‌های جدی را اگر زمان کم است تغییر ندهید. مثلا سیستم مترو را اگر از ساعت 2 بامداد تغییر دهیم تا ساعت 5 و آغاز به کار مترو 20 دفعه می‌توانیم آن را restart کنیم.

- مرتب‌اوضعیت برق را و UPS را کنترل کنیم.

- اتاق کامپیوتر را از نظر آلودگی (خاک، دود ...) کنترل نمائیم.

- درجه هوای درصد رطوبت را کاملا کنترل نمائیم.

ماتریس Application Dependencies را داشته باشیم تا اگر یک سیستم کاربردی از کار افتاد، مشخص گردد که سیستم‌های دیگر از کار خواهد افتاد.

از بعد از شی گرایی مدد شده است که نرم افزارها از امکانات هم استفاده کنند (Reusability).

- در مورد امنیت دستورالعمل‌ها را دقیقاً اجرا کنیم.

امنیت بدون مدیریت اصلا فایده‌ای ندارد و نباید کار سلیقه‌ای باشد و حتماً به وسیله افرادی که تجربه زیادی در این زمین دارند انجام گیرد.

با داشتن یک برنامه مدون

- حداقل down time را خواهید داشت.

- زمان recovery به حداقل ممکن خواهد رسید.

- مشتریان را از دست نخواهید داد.

- مدیریت به شما اعتماد خواهد نمود.

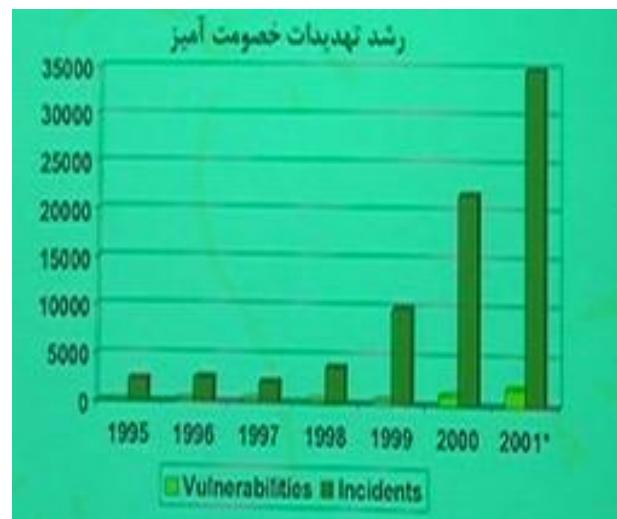
- شب‌ها راحت خواهید خوابید!

- از سیل و زلزله نخواهید ترسید!

Information Security Management System

معنی امن بودن چیست؟

- چه چیزی بایستی امن نگه داشته شود؟
- از چه کسی بایستی حفظ کرد؟
- برای چه مدتی بایستی امن نگه داشته شود؟
- تا چه حد بایستی در مقابل حملات مقاومت نمود؟



منبع: www.cert.org/stats نه ماهه اول سال

cert مثل آتش نشانی است؛ به هر کسی حمله شود به cert اطلاع می‌دهد. در ایران هم نیاز است که چنین سیستمی راه اندازی شود که کارشناسان و تجاربسان در یک پایگاه داده گردآوری شود.

امنیت در چه حد

فکر می‌کنید تا چه حد امنیت برای سازمان شما لازم است:

- با توجه به وقایعی که اطراحتان می‌گذرد
- با توجه به قبول ریسک
- با توجه به ارزش دارایی‌ها (داده‌ها)

- به خاطر داشته باشید هزینه به دست آوردن مجدد دارایی ها(اطلاعات) در صورت به هم ریختن جقدر خواهد بود.
- بیاد داشته باشید که امنیت مطلق حصول ناپذیر است.

یکی از موانع امنیت در سازمان ها محدودیت بودجه است چون خیلی هزینه بر است. هر جایی که خواستار امنیت شدند باید کتابچه سیاست امنیتی را از آنها بخواهید چون کار سلیقه ای نیست. نرم افزارهایی مثل IE یا Firefox آمن نیستند.

انواع تهدیدها

- هکرها
- حملات از نوع بازدارنده
- سازمان جاسوسی پ
- کارکنان سابق
- نامه های ناخواسته و بنجل (funk)
- کرم - ویروس - اسپ تروا
- خرابکاری غیرعمدی
- کارکنان فعلی!
-

3 رکن اصلی امنیت

محرمانگی (confidentiality)

جلوگیری از دسترسی افراد غیرمجاز و فاقد صلاحیت به اطلاعات را محرمانگی می نامند و این اصل از مهم ترین جنبه های امنیتی برای سازمان های نظامی و اطلاعاتی و وزارت خانه ها که عمدتاً وابسته به دولت می باشند بوده و یکی از بهترین روش ها برای محرمانه نگه داشتن اطلاعات که از زمان سزار تا کنون مورد استفاده قرار می گیرد. پ

صحت (integrity)

حفظ از داده ها و اطلاعات در مقابل تغییرات غیر مجاز که ممکن است عمده و یا سهوی باشد را صحت می نامند.

فراموش نگردد که سیستم های امنیتی کامپیوترا به تنها یعنی نمی توانند درستی داده های ورودی را که یک طرف آن نیروی انسانی باشد را ارزیابی نمایند ولی می توانند مسیری که داده در سیستم می پیماید و اثر گذاری هایی که بر روی کل سیستم می گذارد را کنترل نمایند (به عنوان مثال مکانیزم Auditing در oracle و یا بانک های اطلاعاتی مدرن دیگر)

دردسترس بودن (Availability)

صرف کننده داده ها (Users) همواره توقع دارد که اطلاعات صحیح به موقع در دسترس باشد و معمولاً هکرها سعی دارند به هر طریق ممکن کامپیوتر را در شرایطی قرار دهند که به سرویس های خواسته شده جواب منفی بدهد که اصطلاحاً به این مقوله نفی خدمت (Denial Of Service Attack) می‌گویند. به عنوان مثال در حدود اوایل اردیبهشت 1383 تعداد 13 خادم اصلی شبکه جهانی اینترنت را وادر به نفی سرویس نمودند و اختلال عظیمی در سرویس دهی شبکه جهانی بوجود آمد (نفی سرویس را اصطلاحاً DoS می‌نامند که مخفف Denial of service می‌باشد)

امنیت دو تا تعریف دارد؛ کسی اطلاعات من را بر ندارد (هکر) و دوم اینکه مرا از اطلاعاتم محروم نکند (سیل و هکر)

امنیت را می‌توان به طور خلاصه در شش جنبه اصلی زیر دید:

- امنیت دسترسی فیزیکی
- امنیت نیروی انسانی
- امنیت داده ها
- امنیت شبکه ها و ارتباطات
- امنیت سیستم عامل
- امنیت برنامه های کاربردی

لیست کلی تهدیدها

بستگی به نوع سازمان تهدید شونده دارد.

- خطاهای و غفلت ها به دلیل نبودن استراتژی معین امنیتی در سازمان
- کلاه برداری و دزدی (یه دلیل امن نبودن سیستم های پایه و کاربردی و شبکه)
- خرابکاری عمدى کارمندان (به دلایل متعدد از جمله انتقام جویی)
- فقدان پشتیبانی فیزیکی وزیرساختاری (سرقت کامپیوتر)
- نرم افزارهای مخرب (اجراهی نرم افزارهای مشکوک توسط کاربران کامپیوتر)
- هک کردن (به منظور آزاروادیت و اخطار)
- جاسوسی حرفه ای (بیشتر در مراکز نظامی و اطلاعاتی دولتی)

- عوامل طبیعی (زلزله ... سیل...)

- خطاهای سیستمی (تجهیزات و نرم افزارهایی که به اندازه کافی تست نشده اند)

- سازمان های جاسوسی

بعضی مدیران فکر می کنند به صرف نصب یک آنتی ویروس سیستم امن می شود در حالی که امنیت یک پروسه تکراری و همیشگی است چون خطرات هم تکرار می شوند وهم به روز.



ISMS چیست؟

بخشی از ساختار مدیریت سیستم می باشد که هدف آن طراحی و توسعه و کنترل و نگهداری سیستم هایی می باشد که به امن تر نمودن اطلاعات سازمان کمک نماید. شامل سیاست گذاری ها و تهیه دستورالعمل ها و کنترل مداوم و به انجام رسانیدن کار می باشد.

سیاست گذاری امنیتی

استانداردهای خاص از جمله:

BS7799 •

ISO17799 •

BS7799/ISO27001 چیست؟

مجموعه ای از توصیه های کنترل ایمنی ، استانداردهای بین المللی در زمینه امنیت

تجهیزات •

مدیریت سیاست ها •

نیروی انسانی •

ISO27001 چه حوزه هایی را تحت پوشش قرار می دهد؟

خط مشی امنیتی •

توسعه و نگه داری سامانه ها •

دسته بندی و کنترل دارایی ها •

امنیت سازمانی •

با دستور زیر تمام ارتباطات و پکیج هایی را که رو بدل می شود می بینیم:

```
[root@lpi ~]# tcpdump
```

ما کد ویندوز را نداریم و مشخص نیست که چه کار می کند شاید نرم افزار شنود داشته باشد! پرینتر کاخ صدام که ساخت شرکت اپسون هم بود هر پرینتی که انجام میداد یک کپی از آن را به آمریکا ارسال می کرد.

بررسی فعالیت های راهبر لینوکس

فلسفه راهبری لینوکس

- اتوماتیک نمودن فعالیت ها تا حد ممکن
- بدون نمودن فعالیت ها
- ارتباط هر چه بیشتر با کاربران و افراد مرتبه
- آشنایی با منابع در اختیار
- شناخت کاربران
- شناخت فعالیت های شغلی
- جاره اندیشی برای امنیت سیستم
- پیش بینی برای آینده (دور نگری)
- آمادگی برای پیش آمد اتفاقات غیرقابل پیش بینی

راهبر سیستم عامل لینوکس با به کارگیری نکته های فوق و اجرای دقیق آنها می تواند به نحو مطلوب مشتریان را راضی نگه داشته و باسرعت بیشتری ارتقاء شغلی خواهد داشت.

اتوماتیک نمودن فعالیت ها تا حد ممکن

- کنترل نمودن فضای آزاد دیسک و ارائه گزارش
- تهیه فایل پشتیبان
- جمع آوری اطلاعات به منظور تنظیم نمودن سیستم
- حساب کاربران (تولید-تغییر-حذف)
- تهیه گزارشات لازمه
- فعالیت های مرتبه

در شرکت های خارجی اگر کارمندی به صورت دستی برای گرفتن پشتیبان از یکی از دستورهای tar, cpio استفاده کند اخراج می شود چون ممکن است یکی از دایرکتوری ها را فراموش کند. به عنوان مثال الن در ایرانسل دستور را چند بار تست می کنند بعد برای آن یک job shell درست می کنند. به منظور جمع آوری اطلاعات می توانید از دستور sar استفاده کنید:

```
[n.pardis@lpi ~]$ man sar

SAR(1)                               Linux Users Manual                               SAR(1)

NAME
    sar - Collect, report, or save system activity information.

SYNOPSIS
    sar [ -A ] [ -b ] [ -B ] [ -C ] [ -d ] [ -h ] [ -i interval ] [ -m ] [
    -p ] [ -q ] [ -r ] [ -R ] [ -S ] [ -t ] [ -u [ ALL ] ] [ -v ] [ -V ] [
    -w ] [ -W ] [ -y ] [ -n { keyword [,...] | ALL } ] [ -I { int [,...] |
    SUM | ALL | XALL } ] [ -P { cpu [,...] | ALL } ] [ -o [ filename ] | -f
    [ filename ] ] [ -s [ hh:mm:ss ] ] [ -e [ hh:mm:ss ] ] [ interval [
    count ] ]

DESCRIPTION
    The sar command writes to standard output the contents of selected
    cumulative activity counters in the operating system. The accounting
    system, based on the values in the count and interval parameters,
    writes information the specified number of times spaced at the speci-
    fied intervals in seconds. If the interval parameter is set to zero,
    the sar command displays the average statistics for the time since the
    system was started. If the interval parameter is specified without the
    count parameter, then reports are generated continuously. The col-
    lected data can also be saved in the file specified by the -o filename
```

انواع گزارش ها را می گیرد از جمله آمار شبکه ها ، وقفه ها و... . در شهرداری ما با همین گزارشات فهمی دیدیم که ساعت 9:45 تا 10:30 پیک کاری است.

atop

بعضی، حاها 8 تا cpu دارند ولی، سیستم کند است چون نرم افزار لایسنس، یک Cpu دارد و بقیه idle هستند.

مدون نمودن

مدون نمودن فعالیت‌ها باعث خواهد شد که سیستم به نحو مطلوبی اداره شده و در غیاب راهبر بتوان با وجود منابع تدوین شده سیستم را سرپا نگه داشت. البته بعضی راهبران این کار را انجام نمی‌دهند که سیستم به آنها وابسته باشد که خیلی کار خطرناکی است.

بسیاری از راهبران سیستم عامل یکی اچند نکته زیر را مد نظر قرار می‌دهند:

- پس از پایان هر فعالیت گزارش مدونی راجه به فعالیت تهیه نمایم.
 - چراً ادادداشت کنم. همه چیز را به خاطر دارم!
 - اگر اطلاعات را در ذهن نگه دارم مرا اخراج نخواهند کرد!

کم رنگ ترین نوشه‌ها می‌تواند از پررنگ ترین حافظه‌ها موثرتر باشد.

راهبر بایستی در زمینه‌های زیر مدارکی را مدون نموده و در جای مناسبی نگهداری نماید:

- سیاست‌گذاری‌ها

نوشتن سیاست‌گذاری‌ها باعث می‌گردد که ارتباط رسمی و جدی بین افراد سازمان به وجود آمده و کاربران و افراد ذینفع دقیقاً می‌دانند چه تقاضاها و کمک‌هایی را می‌توانند از راهبر تقاضا کنند.

- دستورالعمل‌ها

تھیه دستورالعمل‌های قدم به قدم فعالیت‌ها باعث خواهد شد که راهبران سایت بتوانند با اطمینان فعالیت‌های حساس مانند تھیه فایل پشتیبان را انجام دهند و همین مسئله باعث خواهد شد که سلایق شخصی و غیرقابل اطمینان مطروح گردد.

فراموش نشود که اگر فعالیتی را چند بار انجام داده و مفید و مناسب برای استفاده در سایت تشخیص داده شد آنرا تبدیل به دستورالعمل نمایید.

در شرکت‌های خارجی اگر می‌خواهید نرم افزاری را روی سیستم نصب کنید حتماً باید دستور العمل‌های آن را بنویسید. در بانک برای پشتیبان گرفتن 15 صفحه دستورالعمل قرار داده ایم که اگر مثلاً هر خطای اتفاق افتاد نحوه برخورد با آن به چه طریقی باشد و... در لینوکس فایلی داریم که آخرین ورژن آن حدود 70MB حجم دارد که تمام howto‌ها با فرمت pdf یا html در آن موجود است؛ برای دانلود آن کافی است عبارت linux howto را در گوگل جستجو کنید

- تغییرات

یکی از فعالیت‌های مهم یک راهبر مدون نمودن تغییراتی است که در سایت انجام می‌گیرد و تغییراتی که برای بهینه کردن فعالیت سیستم و امنیت بالاتر انجام می‌گیرد. به عنوان مثال تغییراتی که در روش تعریف چاپگر و کاربر و غیره به وجود می‌آید بایستی مدون گردد و بدینیست که در فرمی که برای مدون نمودن تغییرات طراحی نموده اید عناصر زیر را مدنظر قرار دهید:

- نام و نام خانوادگی فردی که تغییرات را انجام داده است.

- تاریخ و ساعت تغییر انجام شده.

- علت تغییرات. (مثلاً ورژن قبلی همچین باگی داشته)

- تاثیرگذاری تغییرات بر روی سیستم.

سازمان‌هایی مثل مخابرات همیشه تغییراتی دارند که اگر جایی ثبت نشود مشکلات زیادی به وجود می‌آید.

ارتباط هرچه بیشتر و بهتر با کاربران

راهبر سایت با توجه به سیاست گذاری های سازمان تابعه بايستی با کاربران و افراد ذینفع ارتباط داشته باشد که می توان ارتباط از طریق Usenet News را نام برد. در هر حال بد نیست مسئول سایت از طرقی که صلاح می دارد به کاربران مطالب زیر را اعلام کند.

- چه فعالیتی را در نظر دارید که انجام دهید.
- چه فعالیتی را انجام می دهید.
- چه فعالیتی را انجام داده اید.

قبل از هرگونه فعالیتی بايستی کاربران را به تغییراتی که در سیستم انجام خواهد داد مستقیماً و یا غیرمستقیم به فعالیت کاربران مربوطه می گردد را آگاه نمائید که حداقل آگاهی عبارتند از:

- ماهیت تغییر
- زمان اعمال تغییرات
- علت اعمال تغییرات
- مدت دوره اعمال تغییرات
- اثر تغییرات بر روی فعالیت کاربران
- برقراری نقطه تماس به منظور پاسخ به سوالات مرتبط با تغییرات

در شهرداری هر کس login می کند یک صفحه خبر برایش ایمیل میکنیم که مثلا ساعت 2 سیستم down می شود. متن زیر عیناً کپی یکی از همین پیغام هاست که از شنبه صبح برای کاری که قرار بود پنج شنبه همان هفته انجام شود ایمیل کردیم:

از ساعت 12 نیمه شب مورخه 12/12/87 کامپیوتر MAH به مدت 5 ساعت و به منظور نصب آخرین گونه Oracle پائین خواهد بود.

برای اطلاعات بیشتر با داخلی 123 تماس بگیرید.

باتشکر

مسئول سایت

پس از پایان فعالیت های مربوطه بایستی به کاربران اعلام نمایید که چه عملیاتی انجام داده اید و آیا عملیات شما تاثیر مثبتی بر روی فعالیت آنان داشته است و اگر به دلایلی تغییرات مورد نظر شما از جمله کمبود جا و فضا و یا پردازش و غیره انجام نشده است آنرا صریحاً اعلام نمایید و بد نیست که قید نمایید؛ فعالیت در زمان مناسب دیگری انجام خواهد پذیرفت.

منابع خود را شناسایی نماید

راهبر بایستی منابع را به خوبی شناسائی نموده تا بتواند در موقع معینی به تنظیم سیستم و بطرف کردن نیاز کاربران بپردازد. آشنایی کامل با منبع موجود باعث خواهد گردید تا راهبر فشارهای روحی کم تر را داشته باشد و منابعی که بایستی راهبر به خوبی آنها را بشناسند عبارتند از:

- منابع سخت افزاری مانند حافظه، اندازه دیسک سخت، سرعت پردازش و پهنای باند شبکه.
 - بودجه اختصاص یافته شده برای فعالیت IT و البته توقع نداشته باشید که بودجه دقیق را به شما بگویند.
 - منابع انسانی در مجموعه ای که فعالیت می نماید.
 - زمان (مثلًا مدت زمان برگرداندن فایل backup)
 - منابع موجود شامل کتاب ها و یا بروشورها یا مدارک باقی مانده از افراد قدیمی تر.
- مثلًا دقیقاً در شهرداری می دانستیم فرمت کردن دیسک scsi 4 دقیقه و 18 ثانیه طول می کشد! و یا backup 6 ساعت زمان می برد؛ اگر بیس تر میشد می فهمیدیم که مشکلی به وجود آمده است.

کاربران خود را خوب شناسایی کنید

کاربران سیستم های کامپیوتری را می توان به دو دسته تقسیم نمود:

- کاربران جدی که مستقیماً با سیستم مرتبط بوده و مسئول پاسخگوئی به مراجعان با استفاده از اطلاعات دریافتی از سیستم هستند.
- کاربران اینترنتی که مستقیماً به سیستم متصل نبوده و مسئول پاسخگوئی نمی باشند و فقط به منظور دریافت اطلاعات با سیستم مرتبط می باشند.

راهبر بایستی درجه مسئولیت و پاسخگوئی کاربران را شناخته تا در موقعی که سیستم دچار وقفه و عدم سرویس دهی مناسب می گردد مشکلات و مسائلی به وجود آمده برای کاربران را بهتر درک نماید. مطمئن باشید که شناخت بیشتر و بهتر کاربران، شما را به اهداف خودتان که بهینه کردن سیستم می باشد نزدیک تر خواهد نمود.

شغل و موقعیت خود را بهتر بشناسید

- آیا شما در سازمانی کوچک و یا در سازمانی که در موقعیتی بین المللی قرار دارد مشغول به کار هستید؟
- آیا شما راهبر یک سایت دانشگاهی می باشید؟
- آیا می دانید برای چه هدفی شما را انتخاب نموده اند؟

دانستن اطلاعات فوق به شما کمک خواهد نمود که به نحو بینه‌ای فعالیت‌های لازمه را انجام داده و برنامه ریزی‌های مناسب را برای بهتر نگه داشتن سیستم انجام دهید. مطمئن باشید که شناخت موقعیت شغلی، شما را به اهداف خودتان که همانا سرویس دهی مناسب می‌باشد خواهد رسانید. بعضی از برنامه ریزی‌ها عبارتند از:

- مقاطعی که بایستی فایل پشتیبان تهیه نمود.
- موقعي که بایستی سیستم را maintenance نمود.
- ترغیب دوره‌ای مدیران برای آشنایی هر چه بیش تر با تکنولوژی جدید نرم افزاری و سخت افزاری.

امنیت نمی‌تواند کم اهمیت گرفته شود

راهبر نمی‌تواند به هیچ وجه امنیت را کم اهمیت در نظر بگیرد حتی اگر مسئولیت یک سیستم ساده و بدون اتصال به شبکه را داشته باشد. امنیت بالاترین اهمیت را برای یک سایت کامپیوتری دارد و راهبر بایستی همواره آگاهی‌های زیر را داشته باشد.

- حملات احتمالی از چه طریق انجام خواهد شد.
- محل قرار گرفتن اطلاعات و اهمیت آنها را در سیستم بداند.
- نوع و مجوز دسترسی کاربران و همکاران را بداند.

ساده نگری خواهد بود که راهبر تصور نماید، حملات فقط از طریق خارج از سازمان می‌باشد و آمار نشان داده است که بسیاری از حملات داخلی می‌باشند.

پیش‌بینی برای آینده (آینده نگری)

راهبر با توجه به وضعیت نرم افزار و سخت افزار و تعداد و تنوع کاربران و سیستم‌ها و با در نظر گرفتن جمیع جهات بایستی اقدامات لازمه را در جهت ارتقاء سیستم به عمل آورده و با تهیه گزارشات لازمه و مدلل مدیران را ترغیب به بروز در آوردن سیستم‌ها نمایند. فراموش نگردد که جدی نگرفتن آینده نگری، باعث خواهد گردید که پس از چند سال با سیستم فرسوده و غیر قابل پشتیبانی روبرو شوید. معمولاً مهاجرت نرم افزاری در سازمان‌ها خیلی مشکل است؛ سایت‌های داریم که برنامه‌هایشان تحت dos است!

آمادگی لازم برای مواجهه با اتفاقات غیرقابل پیش‌بینی

علیرغم تمام کنترل‌های لازمه این امکان وجود دارد که اتفاقاتی در سیستم به وقوع بپیوندد که غیرقابل پیش‌بینی بوده اند و از جمله این اتفاقات عبارتند از:

- فعال نشدن UPS بعد از قطع شدن برق شهر.
- خراب شدن یک دیسک سخت نو.
- پرشدن فضای دیسک که فکر می‌کردید به اندازه کافی فضای خالی دارد.

در مواقعي که اينگونه اتفاقات غيرقابل پيش بيني راهير بايستي با خونسردي تمام و اعتماد به نفس برای بروطوف نمودن اشکالاتي که به وي مربوط می شود پرداخته و برای رفع مشكلاتي که نياز به هماهنگي دارد (عمل نکردن UPS) سريعا با ساختارهای ذيربط برای رفع اشکال تماس حاصل نماید و با مهرباني و ملائمت به تلفن هاي کاربران گله متند از Down بودن سистем پاسخ دهد و بداند که حق با کاربران پاسخگو به ارباب رجوع می باشد. در شهرداري برای تست UPS برق شهر را قطع می کردیم یا يك بار يك سور گران قيمت از sun خريديم و با nfact تست کردیم ديسك سوخت. در اين بخش می خواهيم با دستور sar از سистем آمارگيري کنيم.

[n.pardis@lpi ~]\$ sar 2 10							
Linux 2.6.32-220.el6.i686 (lpi) 10/01/2012 _i686_(1 CPU)							
	CPU	%user	%nice	%system	%iowait	%steal	%idle
02:32:35 PM	all	0.00	0.00	1.00	0.00	0.00	99.00
02:32:37 PM	all	0.00	0.00	0.50	0.00	0.00	99.50
02:32:39 PM	all	0.00	0.00	1.00	0.00	0.00	99.00
02:32:41 PM	all	0.00	0.00	0.50	0.00	0.00	99.50
02:32:43 PM	all	0.00	0.00	1.00	0.00	0.00	99.00
02:32:45 PM	all	0.00	0.00	0.51	0.00	0.00	99.49
02:32:47 PM	all	0.00	0.00	0.50	0.00	0.00	99.00
02:32:49 PM	all	0.50	0.00	11.62	0.00	0.00	87.88
02:32:51 PM	all	0.51	0.00	0.50	0.00	0.00	99.50
02:32:53 PM	all	0.00	0.00	1.00	0.00	0.00	99.00
02:32:55 PM	all	0.10	0.00	1.80	0.00	0.00	98.10
Average:	all	0.10	0.00	1.80	0.00	0.00	98.10

در ترمinal بالا 10 مرتبه هر 2 ثانیه يك بار وضعیت سیستم را گزارش می کند. مثلا در بانکی با همین اطلاعات ثابت کرده می که خرید 32 تا CPU ضروري نبوده و با 2 تا CPU هم مشکلی نبوده است.

[n.pardis@lpi ~]\$ sar 1 1 -b					
Linux 2.6.32-220.el6.i686 (lpi) 10/01/2012 _i686_(1 CPU)					
	tps	rtps	wtps	bread/s	bwrtn/s
02:44:12 PM	102.00	72.00	30.00	17920.00	416.00
02:44:13 PM					

در ترمinal بالا وضعیت هارد نشان داده شده است.

[n.pardis@lpi ~]\$ sar 1 1 -B									
Linux 2.6.32-220.el6.i686 (lpi) 10/01/2012 _i686_(1 CPU)									
	pgpgin/s	pgpgout/s	fault/s	majflt/s	pgfree/s	pgscank/s	pgscand/s	pgsteal/s	%vmeff
02:48:15 PM	0.00	0.00	37.62	0.00	97.03	0.00	0.00	0.00	0.00
02:48:16 PM	0.00	0.00	37.62	0.00	97.03	0.00	0.00	0.00	0.00
Average:	0.00	0.00	37.62	0.00	97.03	0.00	0.00	0.00	0.00

به معنای page fault است. دلیل اینکه می‌گویند goto به راه دور خوب نیست این است که اگر برنامه به قسمتی که swap شده بپردازد page fault به وجود می‌آید و سیستم عامل مجبور است اطلاعات درخواست شده را از هارد بیاورد.

```
[n.pardis@lpi ~]$ sar 1 -I ALL | less
Linux 2.6.32-220.el6.i686 (lpi)           10/01/2012      _i686_   (1 CPU)

02:54:59 PM      INTR      intr/s
02:55:00 PM      0    1002.00
02:55:00 PM      1     4.00
02:55:00 PM      2     0.00
02:55:00 PM      3     0.00
02:55:00 PM      4     0.00
02:55:00 PM      5     0.00
02:55:00 PM      6     0.00
02:55:00 PM      7     0.00
02:55:00 PM      8     0.00
02:55:00 PM      9     0.00
02:55:00 PM     10     0.00
02:55:00 PM     11     0.00
02:55:00 PM     12     0.00
02:55:00 PM     13     0.00
02:55:00 PM     14     0.00
02:55:00 PM     15     0.99
```

pcی قدمی 15 تا بیش تر ندارند. ترمیнал بالا نشان می‌دهد که وقفه 0 در یک ثانیه 1002 بار اتفاق افتاده است. می‌خواهیم ببینیم این وقفه متعلق به چه وسیله‌ای است:

```
[n.pardis@lpi ~]$ cat /proc/interrupts
CPU0
0:        441  IO-APIC-edge      timer
1:        702  IO-APIC-edge      i8042
3:          1  IO-APIC-edge
4:          1  IO-APIC-edge
7:          0  IO-APIC-edge      parport0
8:          0  IO-APIC-edge      rtc0
9:          0  IO-APIC-fasteoi   acpi
12:       4319  IO-APIC-edge      i8042
14:          0  IO-APIC-edge      ata_piix
15:      10237  IO-APIC-edge      ata_piix
16:      2111  IO-APIC-fasteoi   vmci, Ensoniq AudioPCI
17:     19416  IO-APIC-fasteoi   ehci_hcd:usb1, ioc0
...
...
```

در هر ثانیه 4-3 هزار وقفه بیش تر به کارت شبکه نمی آمده ، گفتن که سیستم خیلی کند شده بررسی کردیم در هر ثانیه 100 هزار packet می آبد فهمیدیم که به ما حمله شده است.

با دستور زیر هر یک ثانیه (n1) محتویات interrupts را چاپ می کند و تغییرات را برجسته می کند(d).

```
[n.pardis@lpi ~]$ watch -n1 -d cat /proc/interrupts
```

هر نیم ساعت یک بار در شرکت‌هایی که مسئولیت آنها را بر عهده دارم این اطلاعات را برای خودم ایمیل می‌کنم.

اگر کرنل نداند باتری در حال ضعیف شدن است یا مادربرد مشکل دارد یعنی درد را حس نکند سیستم crash می‌کند. acpid را حتماً فعال کنید:

```
man acpid
```

<pre>acpid(8)</pre> <p>NAME acpid - Advanced Configuration and Power Interface event daemon</p> <p>SYNOPSIS acpid [options]</p> <p>DESCRIPTION acpid is designed to notify user-space programs of ACPI events. acpid should be started during the system boot, and will run as a background process, by default. It will open an events file (/proc/acpi/event by default) and attempt to read whole lines. When a line is received (an event), acpid will examine a list of rules, and execute the rules that match the event. acpid will ignore all incoming ACPI events if a lock file exists (/var/lock/acpid by default).</p> <p>Rules are defined by simple configuration files. acpid will look in a configuration directory (/etc/acpi/events by default), and parse all regular files that do not begin with a period (.) or end with a tilde (~). Each file must define two things: an event and an action. Any blank lines, or lines where the first character is a hash (#) are ignored. Extraneous lines are flagged as warnings, but are not fatal.</p>	<pre>acpid(8)</pre>
--	---------------------

```
cat /var/log/acpid
```

شارژ باتری که تمام می شود اطلاع می دهد ؛ battery.sh را اجرا می کند.

System Accounting

در خادم هایی که کاربران محلی زیادی دارند (مثل دانشگاه و بانک) نیاز به مجموعه ای فرمان و نرم افزار برای مدیریت و کنترل کاربران است که دلایل آن عبارتند از:

- هر کاربری در مدت معین، چه مدت زمانی را به سیستم متصل بوده است؟
- از چه منابعی و به چه اندازه استفاده نموده است؟
- از کدام آدرس به سیستم متصل شده است؟
- آیا کسی تلاش داشته با Userid غیرمعتبر وارد سیستم گردد؟
- تعدادی فرمان در زمینه به دست آوردن این گونه اطلاعات وجود دارد که ذیلا آمده است :

 - last
 - lastb
 - accton
 - dump-acct
 - dump-wtmp
 - ac & sa

در کلاس شل تمرینی داریم که هر ماه به userID ایمیل بزند که شما چند ساعت استفاده کردید و از چه IP استفاده کردید و ...

ما در کلاس هارد کوچکی داریم به همین خاطر مقداری از حجم Accounting را کم کرده ایم.

last

فرمان last فایل /var/log/wtmp را خوانده و گزارشات متنوعی را در رابطه با کاربران و زمان اتصال به سیستم و IP آدرس آنان را می دهد.

```
[n.pardis@lpi ~]$ last|less
```

User	Terminal	Date	Time	Duration	Message
n.pardis	pts/0	Fri Nov 2 09:07			still logged in
root	tty1	Fri Nov 2 09:07			still logged in
farshad	pts/0		:0.0		Fri Nov 2 09:06 - 09:06 (00:00)
farshad	tty1		:0		Fri Nov 2 09:06 - 09:06 (00:00)
reboot	system boot	2.6.32-220.el6.i			Fri Nov 2 09:03 - 09:08 (00:04)

last userin

اگر نوشته shutdown کردید ولی logout نکردید، يعني gone no logout.

lastb

فرمان lastb فایل /var/log/btmp را خوانده و گزارشات متنوعی را در رابطه با کاربرانی که سعی بر ورود به سیستم نموده ولی موفقیت نداشته اند را ارائه میدهد.

lastb|less

اگر کسی نام کاربری یا کلمه عبور را اشتباه بزند در اینجا نشان می دهد، سیسکو از این خیلی استفاده می کند؛ برای شهرداری نامه داد که یک نفر به زور می خواهد وارد سیستم شود.

ما در شهرداری DHCP استفاده نکردیم تا کاربرها IP استاتیک بگیرند و نام و نام خانوادگی کارمند در کنار IP در پایگاه داده کوچک نگه داری می شد و هر کسک اشتباهی مرتكب میشد یا قصد تخلف داشت ایمیل به او هشدار میدادیم.

سرویس arpwatch خیلی جالب است اگر ادمین هستید حتما آن را نصب کنید.

Address Resolution Protocol - arp

! arp با client سرور را پیدا می کند؛ داد میزند(broadcast) arp

این سرویس Mac Address هر کسی که به سیستم متصل می شود را ثبت می کند؛ اگر یک کاربر از Mac login غیر معمول خود کند به ادمین اطلاع میدهد.

```
[n.pardis@lpi etc]$ man arpwatch
```

ARPSNMP(8)

ARPSNMP(8)

NAME

arp snmp - keep track of ethernet/ip address pairings

SYNOPSIS

```
arp snmp [
```

NAME

arpwatch - keep track of ethernet/ip address pairings

SYNOPSIS

```
arpwatch [ -d ] [ -f datafile ] [ -i interface ]
[ -r file ]
```

DESCRIPTION

Arpwatch keeps track for ethernet/ip address pairings. It syslogs activity and reports certain changes via email. Arpwatch uses pcap(3) to listen for arp packets on a local ethernet interface.

The **-d** flag is used enable debugging. This also inhibits forking into the background and emailing the reports. Instead, they are sent to stderr.

Message 3:

From root@lpi.localdomain Fri Nov 2 10:00:35 2012

Return-Path: <root@lpi.localdomain>

X-Original-To: root

Delivered-To: root@lpi.localdomain

From: arpwatch@lpi.localdomain (Arpwatch)

To: root@lpi.localdomain

Subject: new station ()

Date: Fri, 2 Nov 2012 10:00:35 +0330 (IRST)

Status: RO

hostname:

ip address: 192.168.206.1

ethernet address: 0:50:56:c0:0:8

ethernet vendor: <unknown>

timestamp: Friday, November 2, 2012 10:00:35 +0330

در ترمینال بالا arpwatch در ایمیلی مشخصات سیستمی که login کرده است را نمایش می دهد. ethernet vendor مارک کارت شبکه کاربر است. تا کار شبکه را عوض کنند ایمیلی می آید مبنی بر اینکه این تا دیورز Toshiba بوده الان با 3com وارد شده است.

یک سری اسکریپت خیلی خوب هم زیر var دارد:

```
[n.pardis@lpi ~]$ cd /arpw/var/arpwatch/  
[n.pardis@lpi arpwatch]$ ls  
arp2ethers  arp.dat-  e.awk          euppertolower.awk  p.awk  
arp.dat      d.awk      ethercodes.dat  massagevendor
```

همه 48 بیتی هستند که 24 بیت اول کد کارخانه است که در یک فایل ذخیره شده است:

```
[n.pardis@lpi arpwatch]$ cat ethercodes.dat  
0:0:1SuperLAN-2U  
0:0:10Hughes LAN Systems (formerly Sytek)  
0:0:11Tektronix  
0:0:15Datapoint Corporation  
0:0:18Webster Computer CorporationAppletalk/Ethernet Gateway  
0:0:1aAMD (?)  
0:0:1bNovell (now Eagle Technology)
```

```
[n.pardis@lpi arpwatch]$ cat arp.dat  
0:50:56:c0:0:8  192.168.206.1  1351838708  
0:c:29:e7:29:e9  192.168.206.145  1351838586  
0:50:56:fd:81:b6 192.168.206.2  1351837923  
0:50:56:ea:d9:82 192.168.206.146  1351838268
```

یکی دیگر از فایل‌های مهم شاخه /var/arpwatch mac address است، فیلد اول IP و فیلد دوم time stamp می باشد.

accton

با اجرای فرمان accton تمام فعالیت‌های پروسس‌ها بر روی فایل داده شده ثبت می گردد که بعضی از آنها عبارتند از:

• نام فرمان

تاریخ و ساعت شروع •

مقدار پردازنده مصرف شده •

اطلاعات undocumented •

به شدت performance را پایین می آورد.

```
[n.pardis@lpi arpwatch]$ man accton  
  
ACCTON(8)  
  
NAME  
accton - turns process accounting on or off  
  
SYNOPSIS  
accton [ -V | --version ] [ -h | --help ] [ filename ]  
  
DESCRIPTION  
accton filename turns on process accounting. If called with no  
arguments, it will, by default, stop process accounting.  
  
OPTIONS  
-V, --version  
Print the version number of ac to standard output and  
quit.  
  
-h, --help  
Prints the usage string and default locations of system  
files to standard output and exits
```

باید فایلی درست کنیم که اطلاعات روی آن ذخیره شود:

```
[root@lpi ~]# cd /tmp  
[root@lpi tmp]# >t1  
[root@lpi tmp]# accton t1
```

این فایل به سرعت رشد می کند :

```
[root@lpi tmp]# ls -l t1  
-rw-r--r--. 1 root root 2432 Nov 2 11:04 t1
```

دستور بالا 2 ثانیه بعد از accton اجرا شده است!

فایل t1 یک فایل باینری است اما اسم دستوراتی که log کرده تقریبا مشخص است. با استفاده از این فایل می توانید منحنی رسم کنید که در 24 ساعت گذشته کدام دستورها و به چه تعدادی اجرا شده اند یعنی پس از مدتی فرهنگ سازمانی را متوجه می شوید.

برای stop کردن Accounting اگر فایل را (در اینجا t1) پاک کنید چون busy است واقعا پاک نمی شود . راه متوقف کردن این است که فرمان accton را بدون پارامتر اجرا کنید.

ac

```
[root@lpi tmp]# man ac  
AC(1) AC(1)
```

NAME
`ac` - *print statistics about users connect time*

SYNOPSIS

```
ac [ -d | --daily-totals ] [ -y | --print-year ]  
[ -p | --individual-totals ] [ people ]  
[ -f | --file filename ] [ -a | --all-days ]  
[ --complain ] [ --reboots ] [ --supplants ]  
[ --timewarps ] [ --compatibility ]  
[ --tw-lenienty num ] [ --tw-suspicious num ]  
[ -z | --print-zeros ] [ --debug ]  
[ -V | --version ] [ -h | --help ]
```

DESCRIPTION

`ac` prints out a report of connect **time** (*in hours*) based on the logins/logout in the current wtmp file. A total is also printed out.

The accounting **file** wtmp is maintained by `init(8)` and `login(1)`. Neither `ac` nor `login` creates the wtmp if it doesn't exist, no accounting is done. To begin accounting, create the **file** with a length of zero.

```
[n.pardis@lpi tmp]$ ac  
total 127.11  
[n.pardis@lpi tmp]$ ac -p  
farshad 77.93  
root 18.43  
n.pardis 30.77  
total 127.13
```

sa

مخفف system accounting است و اطلاعات جامعی را در باره منابع استفاده شده توسط پروسس‌ها نشان می‌دهد.

```
[n.pardis@lpi tmp]$ man sa
```

SA(8) **SA(8)**

NAME
sa - summarizes accounting information

SYNOPSIS

```
sa [ -a | --list-all-names ]
    [ -b | --sort-sys-user-div-calls ]
    [ -c | --percentages ] [ -f | --not-interactive ]
    [ -i | --dont-read-summary-file ]
    [ -j | --print-seconds ] [ -k | --sort-cpu-avmem ]
    [ -K | --sort-ksec ] [ -l | --separate-times ]
    [ -m | --user-summary ] [ -n | --sort-num-calls ]
    [ -r | --reverse-sort ] [ -s | --merge ]
    [ -t | --print-ratio ] [ -u | --print-users ]
    [ -v num | --threshold num ] [ --sort-real-time ]
    [ --debug ] [ -V | --version ] [ -h | --help ]
    [ --other-usracct-file filename ]
    [ --other-savacct-file filename ]
    [ --other-acct-file ] filename ]
```

DESCRIPTION
sa summarizes information about previously executed commands as

تمرین 1 : اگر یک userID سه روز وصل نشد غیر فعال شود.

تمرین 2 : از یک userID نتوان همزمان دو تا connection داشت (کاری که ispها می‌کنند).

Dynamic Host Communication Protocol (DHCP)

هر وسیله‌ای اعم از کامپیوت، مسیریاب، سویچ، خادم چاپگر و ... برای اتصال به شبکه به منظور سرویس گیری و یا سرویس دهنی نیاز به آدرسی دارند که اصطلاحاً آن را IP Address می‌نامند.

را می‌توان به دو گونه به وسیله اختصاص داد:

• آدرس ایستا (Static IP Address)

• آدرس پویا (Dynamic IP Address)

هر کدام از روش‌های آدرس دهنی فوق دارای مزایا و معایبی هستند که در ادامه توضیح می‌دهیم. به عنوان مثال در کلاس استاتیک استولی هر سازمانی که رفتید باید بررسی کنید که کدام مناسب‌تر است. وقتی سازمان بزرگتر می‌شود اگر IP را دستی بدھیم ممکن است مثل هم شوند و یا قسمت مشترک را اشتباه وارد کنیم.

مثلاً اگر کسی IP سرور کلاس را بدھد ممکن است کلاس را به هم بریزد. arp که بزنید نشان می‌دهد هر IP چه mac address دارد:

```
[n.pardis@lpi tmp]$ mna aan arp
```

ARP(8)	Linux Programmers Manual	ARP(8)
--------	--------------------------	--------

NAME
arp - manipulate the system ARP cache

SYNOPSIS

```
arp [-evn] [-H type] [-i if] -a [hostname]
arp [-v] [-i if] -d hostname [pub]
arp [-v] [-H type] [-i if] -s hostname hw_addr [temp]
arp [-v] [-H type] [-i if] -s hostname hw_addr [netmask nm] pub
arp [-v] [-H type] [-i if] -Ds hostname ifa [netmask nm] pub
arp [-vnd] [-H type] [-i if] -f [filename]
```

NOTE
This program is obsolete. For replacement check ip neighbor.

DESCRIPTION
Arp manipulates the kernels ARP cache in various ways.

```
[n.pardis@lpi tmp]$ arp
Address          HWtype  HWaddress          Flags Mask        Iface
ether            00:50:56:c0:00:08      C             eth0
```

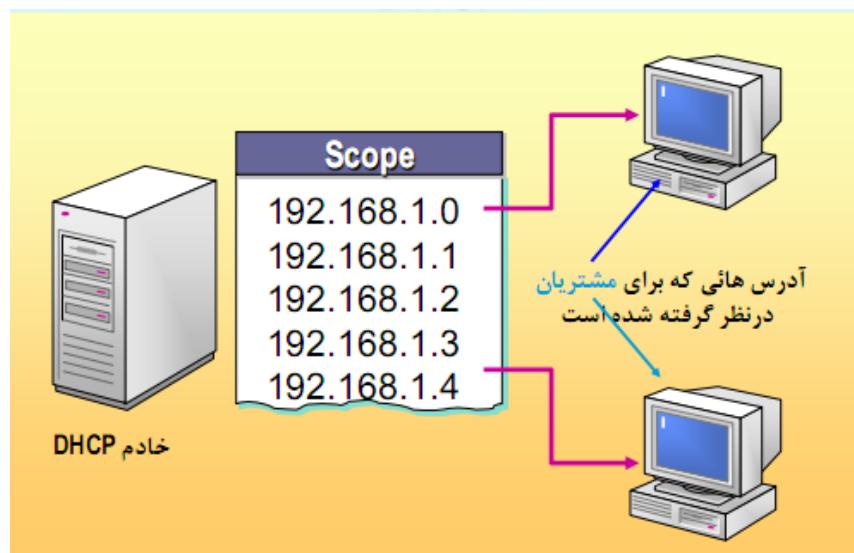
است اگر الان به این کامپیوتر 20 میلیون نفر وصل شوند جدول بالا همه حافظه را می گیرد. به همین خاطر کرنل مرتب ipنهایی را که قدیمی شده است بیرون می ریزد پس arp table دینامیک است و به همین دلیل است که اگر برای مدتی استفاده نکنید ارتباط قطع می شود.

کسانی که به کلاس وصلند اگر در ویندوز یا لینوکسشن arp بزنند مشاهده می کنند که mac متعلق به سرور را داده ولی دائمی نیست و در اتصال بعدی دوباره آن را broadcast می کند حال اگر به یک سیستم با کارت شبکه سریع تر IP سرور را بدھیم به جای سرور به آن وصل می شوند و مثلا اگر ssh نداشته باشد connection refused می گیریم.

در صدawسیما که همه سیستم هایشان مبتنی بر لینوکس بود به این نتیجه رسیدیم که Dynamic باشد که ip duplicate نداسته باشیم ولی در کلاس با استاتیک می فهمیم چه کسی چه اشتباہی انجام داده است.

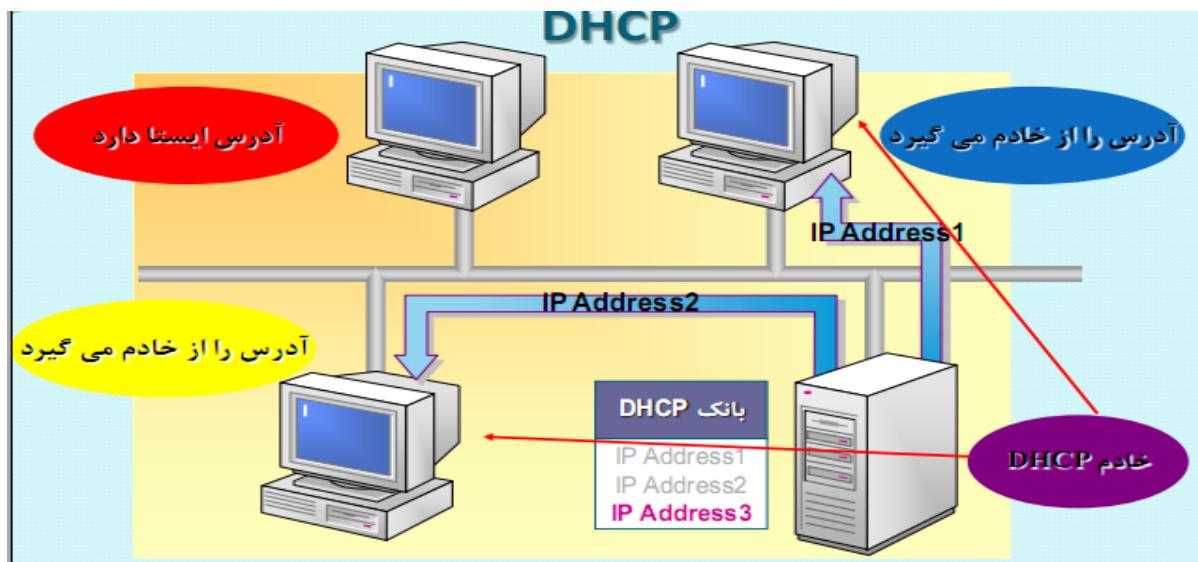
در شهرداری که استاتیک استفاده کردیم الگوریتمی استفاده کردیم که مثلا مناطق و یا پرینترسرور ها از روی IP قابل تشخیص بودند مثلا منطقه 22 با 192.168.220 شروع می شد.

اگر داینامیک هم گرفتیم باید به فکرمسائل امنیتی باشید چون هر کس از طریق واپرس یا کابل به شبکه وصل شود IP می گیرد بهتر است mac address را با arpwatch به دست آوریم بعد به شرطی IP بدھیم که mac Address آن معلوم باشد.



سروز DHCP باید یک پایگاه داده داشته باشد که حاوی آدرس هایی است که ادمین گذاشته است. دلیل استفاده از 192.168.1.198 این است که این آدرس برای کلاس C invalid است یعنی روترا آن را منتقل نمی کنند. کامپیوتر را که روشن می کنیم TCP/IP را کاربر قبلاً کرده بالا که می آید اعلام می کند که به من یک IP بدهید.

در سازمان می توانیم تعدادی از آدرس ها را ایستا بگیریم مثلاً مخابرات 118 را ایستا می گیرد.



TCP/IP در دستی

معایب

آدرس دهی به طور دستی برای هر Client انجام می گیرد

احتمال وارد نمودن آدرس آشتباه بعید به نظر نمی رسد

پیکربندی نادرست می تواند در بخشی از شبکه تاثیرگذار باشد

راهبری شبکه مشکل می باشد، علی الخصوص اگر جا به جائی زیاد باشد

TCP/IP در اتوماتیک

مزایا

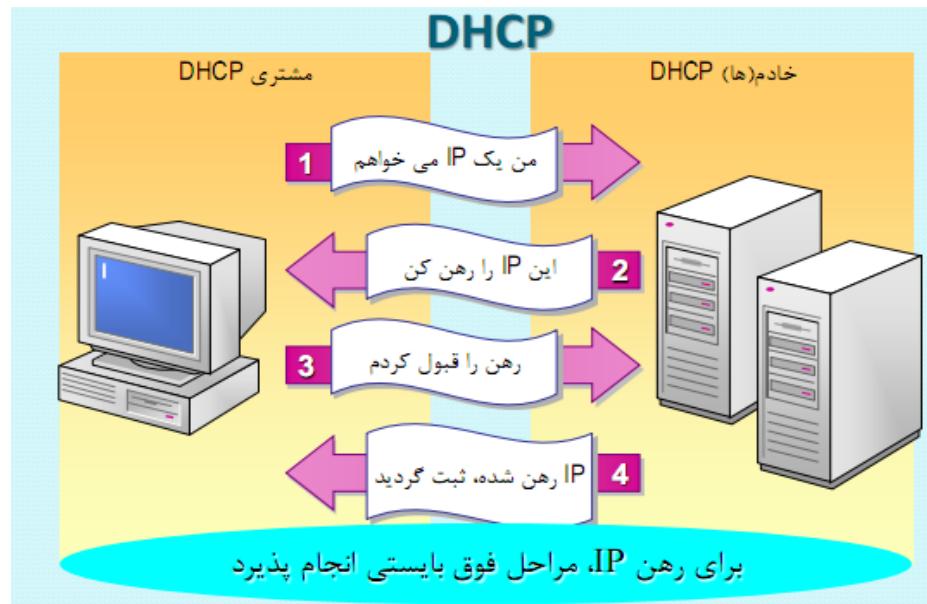
آدرس دهی به طور اتوماتیک برای هر Client انجام می گیرد

احتمال بروز اشتباه در آدرس دهی تقریباً غیر ممکن است

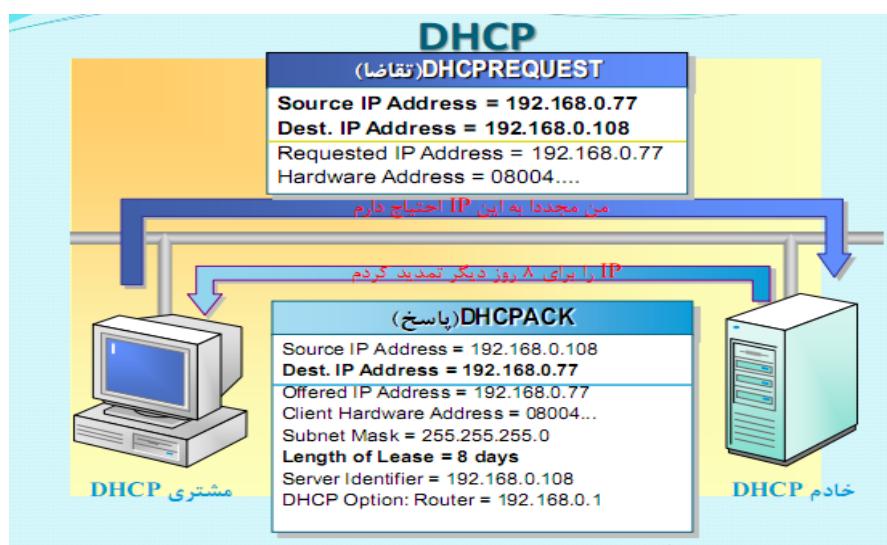
مشکلات کمتری در شبکه می گذارد

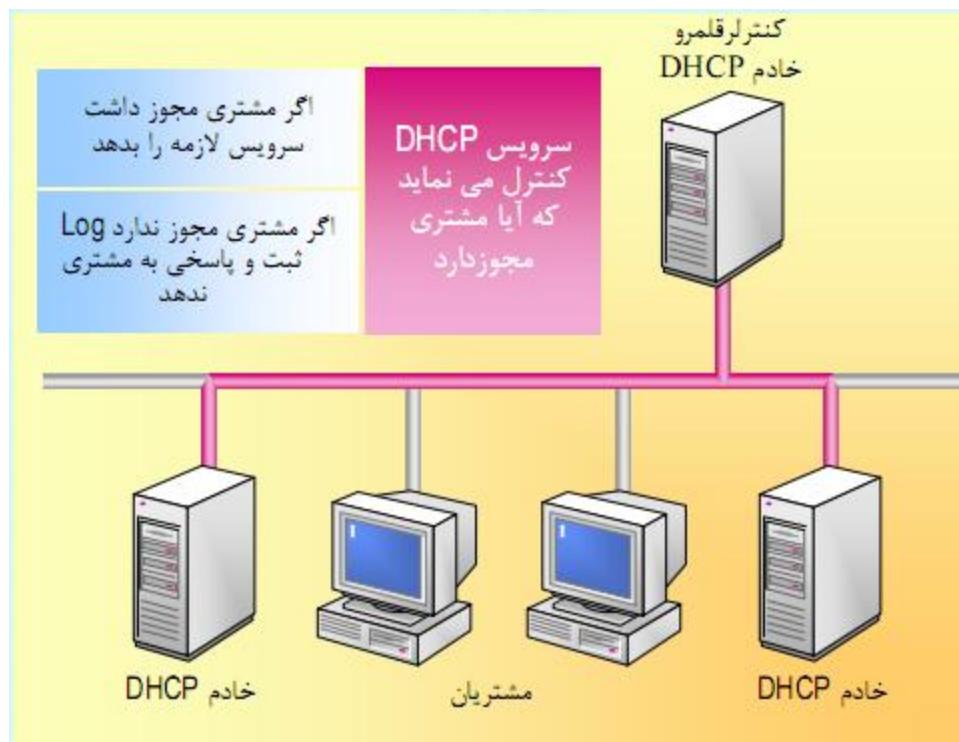
راهبری ساده تر و تغییرات کمتر مگر تغییر اساسی در شبکه بوجود آید

Ip را رهن کردن: leases یعنی leas



دلیلی ندارد که حتماً یک سرور DHCP داشته باشیم می توانیم از چند تا سرور استفاده کنیم. دلیل اینکه ممکن است client IP را قبول نکند این است که وقتی IP را برایش می فرستیم در شبکه اعلام میکند که این IP متعلق به من است کسی مشکلی ندارد؟ و اگر IP را قبلاً یکی به صورت استاتیک اختیار کرده باشد به اعلام می کند که این IP را نمی خواهم. خانه ای را که رهن می کنید مدت محدودی دارد که این مدت را در setup ماشین می توانید مشخص کنید. اگر 75 درصد مدت رهن گذشت باید تقاضای تجدید رهن کنید.





برای گرفتن IP تحت سیستم عامل لینوکس فرمان زیر را وارد کنید:

```
[root@lpi n.pardis]# dhclient
```

اگر خادم سرویس دهنده ای در شبکه باشد به فرمان فوق پاسخ داده و یک IP به کامپیوتر شما رهن می دهد.

در غیر این صورت ، پس از چند بار تلاش توسط dhclient این نرم افزار به پشت صحنه رفته و هرچندثانیه تلاش بر ارتباط و گرفتن IP دارد که گزارش آن بر روی کنسول ظاهر خواهد شد.

کسانی که مودم ADSL دارند اگر ps بزنند حتماً یک dhclient در آن می بینند. فرمان dhclient با پورت 67 درخواست می کند و با 68 جواب می گیرد:

```
[root@lpi n.pardis]# less /etc/services
bootps          67/tcp                                # BOOTP server
bootps          67/udp
bootpc          68/tcp                                # BOOTP client
bootpc          68/udp
```

همان طور که قبلا توضیح دادیم پورت 67 برای بوت شدن hethin client هم استفاده می شود. که حتی سیستم عاملی برای آن خریده ایم که ۵ مگابایت بیش تر حجم ندارد؛ فقط مانیتور و mouse و کیبورد را می خواهد کنترل کند.

```
[root@reza etc]# cat dhcpcd.conf
ddns-update-style interim;
ignore client-updates;
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0 ;
option broadcast-address 192.168.1.255 ;
#option routers 192.168.1.254 ;
#option domain-name-servers 192.168.1.1, 192.168.1.2 ;
#option domain-name "jalal.myself.org" ;

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.10 192.168.1.100 ;
    range 192.168.1.150 192.168.1.200;
}
[root@reza etc]#
```

مثال ساده ای از فایل پیکربندی DHCP

البته 600 خیلی کم است چون مثلا بعد از 400 ثانیه باید دوباره تقاضای رهن کند. عین اجاره‌ای که هر 20 روز یک بار باید تمدید شود که کلی از وقت ما در مشاور املاک تلف می‌شود. همچنین بعد از 7200 ثانیه ارتباط قطع می‌شود خیلی ایمیل داشتیم که؛ در حال کار بودیم که یک دفعه سیستم قطع شد.

```
[root@jalal root]# dhclient
Internet Software Consortium DHCP Client V3.0pl2
Copyright 1995-2001 Internet Software Consortium.
All rights reserved.
For info, please visit http://www.isc.org/products/DHCP

Listening on LPF/lo/
Sending on LPF/lo/
Listening on LPF/eth0/00:80:c7:ef:ea:da
Sending on LPF/eth0/00:80:c7:ef:ea:da
Sending on Socket/fallback
DHCPREQUEST on eth0 to 255.255.255.255 port 67
ip length 328 disagrees with bytes received 332.
accepting packet with data after udp payload.
DHCPACK from 1.1.1.2
bound to 192.168.1.200 -- renewal in 301 seconds.
[root@jalal root]#
```

اگر تحت سیستم عامل Linux بوده و میخواهید IP Address از خادم DHCP بگیرید

راه های زیادی وجود دارد که یکی از این روش ها وارد نمودن فرمان dhclient می باشد.

IP داده شده توسط خادم ۱۹۲.۱۶۸.۱.۲۰۰ می باشد

البته حدود ۳۰۰ ثانیه دیگر بایستی تمدید رهن نمود

بررسی Raid و Disk Striping در لینوکس

به منظور بالا بردن سرعت و حجم اطلاعات نگه داری شده و تضمین بیشتر برای از بین نرفتن اطلاعات می‌توان از نوعی پیکربندی دیسک استفاده نمود که با نام RAID شناخته می‌گردد.

استفاده از تکنولوژی RAID هزینه سخت افزار را تا حدی بالا برده است ولی سرعت بالاتر و اطمینان خاطر بیشتری را فراهم می‌نماید.

در سیستم عامل لینوکس می‌توان از امکانات Software Raid نیز استفاده نمود و در این حالت RAID تحت نرم افزار شبیه سازی شده و اگر دیسک سخت با ظرفیت بسیار نصب باشد ولی از همه فضای آن استفاده نمی‌گردد می‌توان با روش نرم افزاری نیز از امکانات RAID استفاده نمود (با سرعت وظیری اطمینان کم تر)

در این روش اطلاعات را در دو جا می‌نویسیم که به ندرت هم استفاده می‌شود.

RAID 0

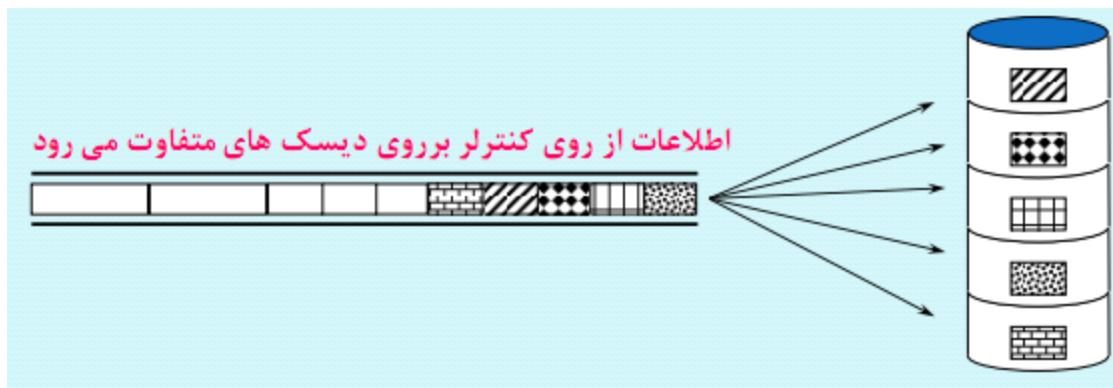
اصطلاحاً به آن Disk Striping می‌گویند.

فرض کنید برنامه‌ای داریم که می‌خواهد یک آرایه 100 تایی از کاراکترها روی دیسک بنویسد. برنامه به سیستم عامل اعلام می‌کند که می‌خواهیم این اطلاعات روی دیسک قرار بگیرد، سیستم عامل پارامترها را چک می‌کند اگر درست بود یک سری دستور درست می‌کند و آنها را به کنترلر دیسک ارجاع می‌دهد. شما سیستم را که بالا آوردید در setup بایوس دیدید که هر کنترلری یک آدرس حافظه دارد (interrupt) هر دستگاهی یک جایی از حافظه را برای ارتباط با سیستم عامل در اختیار دارد که اصطلاحاً به آن mailbox می‌گویند:

```
[n.pardis@lpi ~]$ cat /proc/iomem
00000000-0000ffff : reserved
00010000-0009f7ff : System RAM
0009f800-0009ffff : reserved
000a0000-000bffff : Video RAM area
000c0000-000c7fff : Video ROM
000ca000-000cbfff : reserved
    000ca000-000cafff : Adapter ROM
000dc000-000e3fff : reserved
000e8000-000fffff : reserved
    000f0000-000fffff : System ROM
00100000-3fefffff : System RAM
    00400000-008364e2 : Kernel code
    008364e3-00a917c7 : Kernel data
    00b15000-00c3af6f : Kernel bss
```

سیستم عامل در رجیسترها کنترلر جدولی شبیه جدول زیر قرار می دهد کنترلر اینها را می خواند و می فهمد که باید از آدرس 100 به اندازه 2k بخواند.

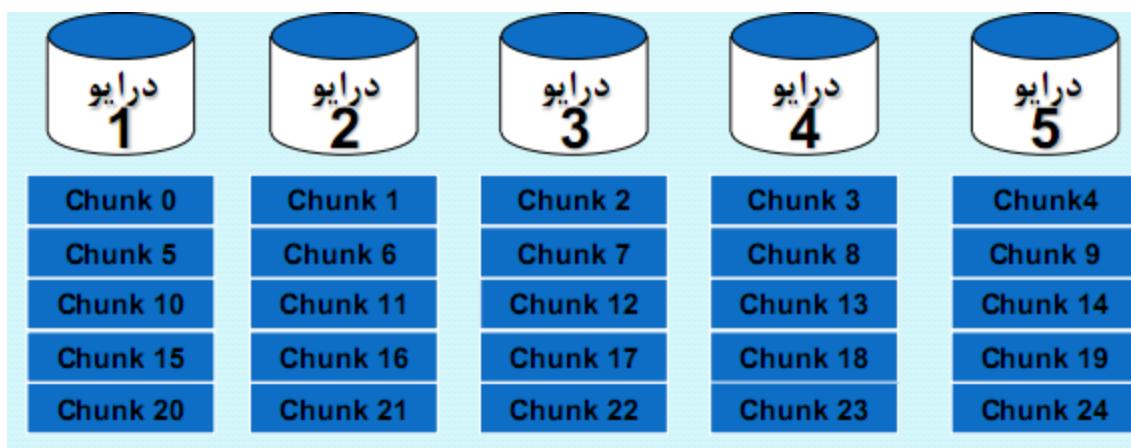
100	2
-----	---



کنترلر بلاک های 512 بایتی را خیلی سریع روی قسمت های مختلف دیسک می نویسد.

کاری که سیستم عامل با وسایل I/O انجام می دهد شامل دو بخش است؛ یکی initial interrupts که ناخدای کشته به سکاندار فرمان می دهد 2 درجه به چپ!

و بخش دوم T1 است مثل حالتی که سکاندار دستور ناخدا را اعمال کرده و اعلام می کند که 2 درجه به چپ (انجام شد!)



پس در RAID 0 فقط یک کپی از اطلاعات داریم.

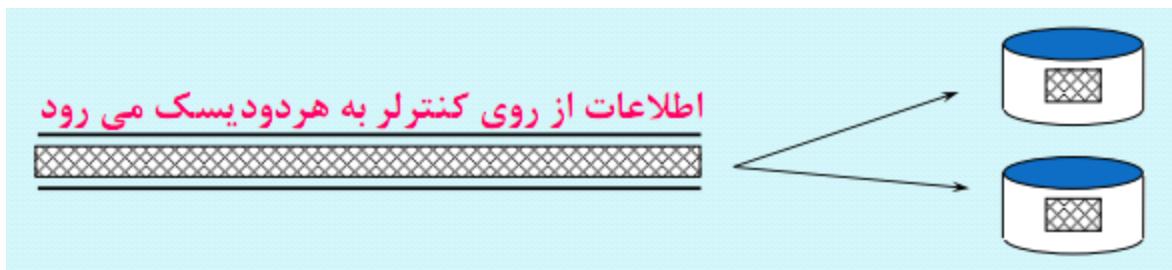
chunk یعنی یک فضایی از حافظه.

انتقال اطلاعات با سرعت بالا

خطراز دست دادن اطلاعات

(Disk Mirroring) RAID 1

در این روش ما یک رونوشت از اطلاعات نگه داری می کنیم. دیسکهای جدید خیلی کم بدستور می دهد ولی می سوزد! این روش برای سیستم های کوچک و متوسط مناسب است ولی در سازمان های بزرگ هزینه زیادی دارد. معمولا در سازمان ها هزینه نگه داری 1/12 هزینه تکنولوژی است پس اگر دیسک ها را زیاد کنیم هزینه پشتیبانی هم زیاد می شود از طرف دیگر مصرف برق هم بالا می رود. همچنین ما تجربه کرده ایم که این روش 15 الی 30 درصد سیستم را کند می کند.



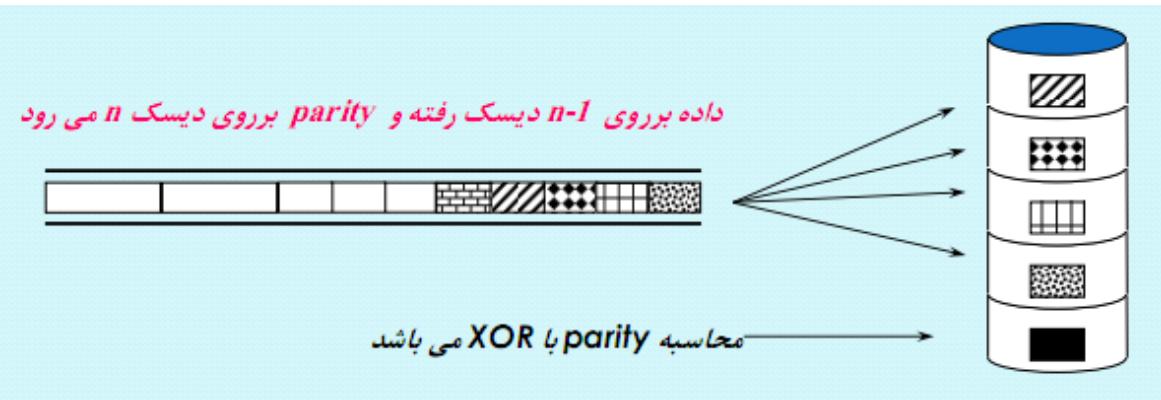
در RAID 0 حتی دو تا کنترلر می گذاریم روی دو تا باس جدا که اگر یک کنترلر باشد و بسوzd باز اطلاعات ذخیره نشده است.

RAID 3(Disk Striping with dedicated parity drive)

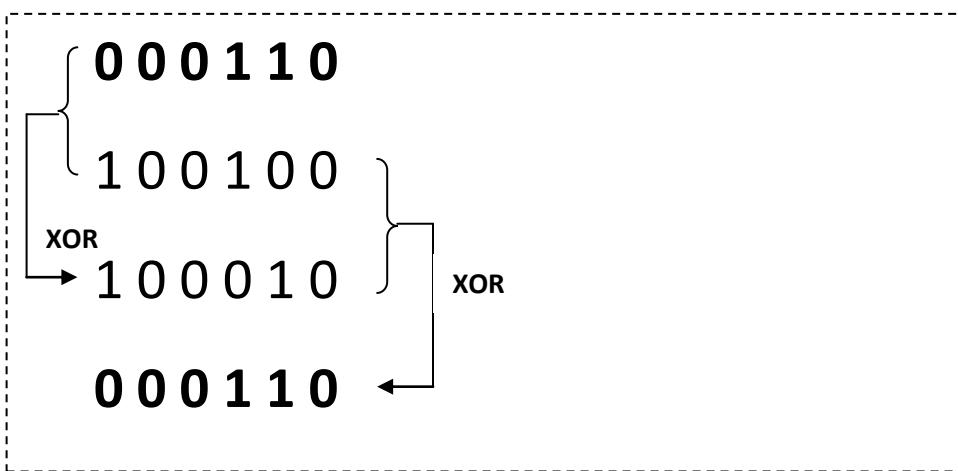
بهترین روش برای دسترسی به اطلاعاتی است که به صورت متوالی یا sequential خوانده می شود. حتی با خرابی کامل یک دیسک سخت اطلاعات از دست نمی رود ولی دیسک اضافه تحمیلی بر سیستم خواهد بود.

استفاده از این روش هزینه بالایی را خواهد داشت ولی اطلاعات با تضمین بیشتری در دسترس خواهد بود. همچنین این روش در ساختمان هایی که لرزش زیاد است با مشکل مواجه می شود و الان اکثر مراکز داده را در چندین متر زیر سطح زمین احداث می کند.

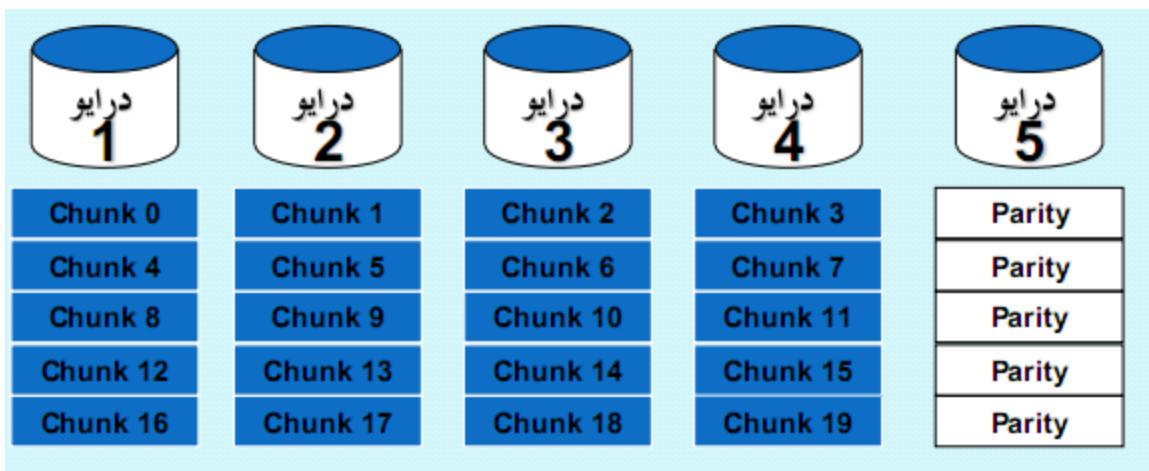
در RAID 3 هر سکتور را روی یک دیسک می نویسند و سپس Parity را روی دیسک آخر می نویسند به این ترتیب اگر هر کدام از دیسک ها بسوzd با کمک دیسک Parity و دیگر دیسک ها قابل بازیابی است.



خاصیت XOR این است که اگر نتیجه را با یکی از عملوندها دوباره XOR کنید(هردو بیت مثل هم بودند جاب 0 و اگر مخالف هم بودند جواب 1 است) عملوند دوم به دست می آید:



یکی از دلایل استفاده از XOR این است که از لحاظ سخت افزاری سریع است و از طرف دیگر AND در چنین شرایطی همیشه 0 و OR اکثرا 1 برمی گرداند.



برای اطمینان از صحت اطلاعات در پورت سریال هم از parity و هم xor استفاده می کنیم چون احتمال اینکه در این مسیر بیشتر از یک بیت تغییر کند، هست در این روش معادل دهدی کد باینری که می خواهیم منتقل کنیم حساب می کنیم و آن را بر عدد مشخصی که با محاسبات ریاضی حساب شده است تقسیم می کنیم سپس باقیمانه حاصل از این تقسیم را در جلوی اطلاعات اصلی قرار می دهیم در مقصد با این دو عدد و عدد مقسوم علیه مشخص چک می کنند که اطلاعات به درستس منتقل شده باشد. روی کارت های شبکه یک IC ریز قرار دارد که این چک را انجام میدهد.

همین کار روی tape انجام می شود با این تفاوت که مقسوم علیه در network از درجه 32 می باشد ولی در tape از نوع polynomial درجه 16 می باشد. ولی در این روش داده ها قابل برگشت نیست فقط مقصد NACK می فرستد و مبدأ فریم را retry می کند.

حداقل 3 تا دیسک می خواهد ولی اگر دو تا از دیسک ها بسوزد گاری نمی توان کرد.

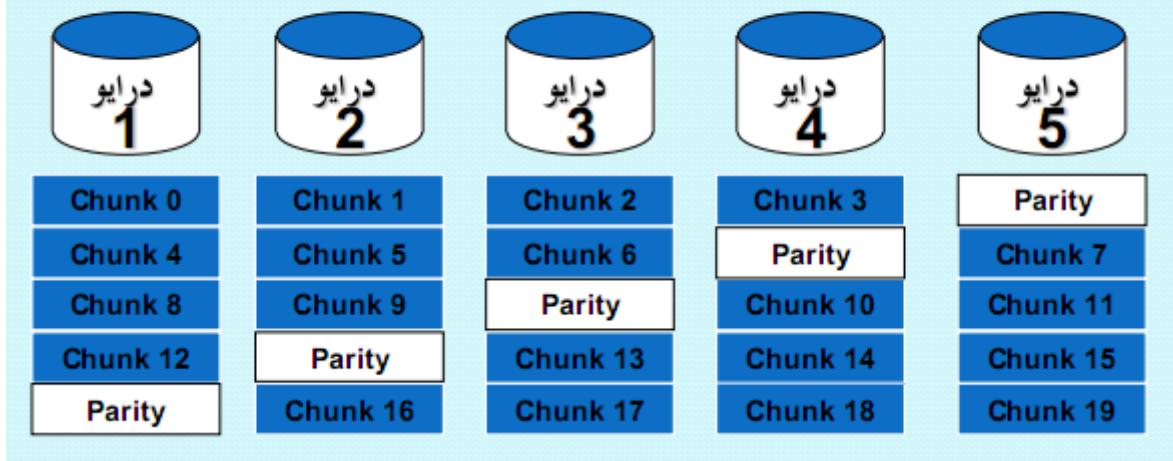
RAID 5(Disk striping with rotating parity block)

اینروش چون به سختی به سخت افزار بستگی دارد در عمل نمی توان آن را تست کرد. در این روش به جای اینکه همه parity ها روی یک دیسک قرار دهیم آنها روی همه دیسک ها پخش می کنیم. از این روش خیلی استفاده می شود و منطق آن هم به آمار و احتمالات و خرابی و عمر مفید دیسک برمی گردد.

- بهینه و بالا بودن تعداد عملیات ورودی/خروجی بر ثانیه

- حتی با خرابی کامل یک دیسک سخت اطلاعات از دست نمی رود.

نوشتن Parity بر روی دیسک های متفاوت برای بالا بردن سرعت می باشد



استفاده از این روش هزینه بالایی را خواهد داشت ولی اطلاعات با تضمین و سرعت بیشتری در دسترس خواهد بود. در raid سختافزاری همه زحمات را منترler متحمل میشود و به همین خاطر کنترلر و درایورش خیلی گران قیمت هستند.

دستوری داریم که سرعت لینوکس را خیلی بالا می برد:

```
[n.pardis@lpi ~]$ man hdparm
```

HDPARM(8)

NAME
hdparm - get/**set** SATA/IDE device parameters

SYNOPSIS
hdparm [flags] [device] ..

DESCRIPTION
hdparm provides a command **line** interface to various kernel interfaces supported by the Linux SATA/PATA/SAS "libata" subsystem and the older IDE driver subsystem. Many newer (**2008** and later) USB drive enclosures now also support "SAT" (SCSI-ATA Command Translation) and therefore may also work with hdparm. Eg. recent WD "Passport" models and recent NexStar-3 enclosures. Some options may work correctly only with the latest kernels.

OPTIONS
When no flags are given, **-acdgkmur** is assumed. For Get/Set options, a

این دستور پارامترهای کنترلر هارد دیسک را عوض می کند ولی خیلی خطروناک است به کنترلر می گوید که سریع تر کار کن! مثلا به جای 16 بیت اطلاعات را در بسته های 32 بیتی جایه جا کند.

همچنین با این دستور می توان سرعت خواندن از بافر یا دیسک را برابر یک فلاش مموری به دست آورد:

```
[root@lpi dev]# hdparm -Tt /dev/sdb
```

/dev/sdb:
Timing cached reads: 1716 MB **in** 2.00 seconds = 858.47 MB/sec
Timing buffered disk reads: 36 MB **in** 3.03 seconds = 11.87 MB/sec

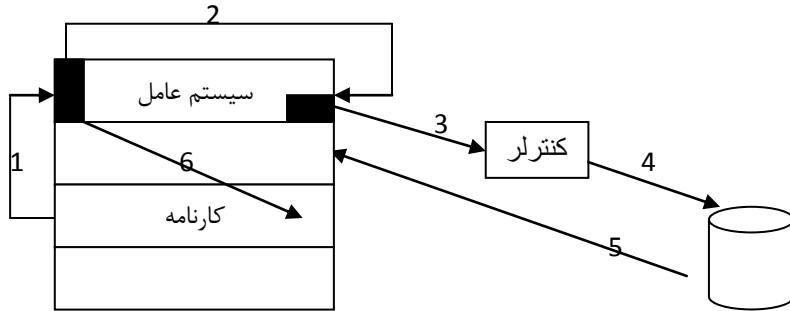
که sdb اسم فلاش در سیستم است و ممکن است در سیستم شما متفاوت باشد (sda یا sdc و ...)

Cooked And Raw Devices بررسی

در تبادل اطلاعات به صورت **cooked** ، تمام تلاش سیستم بر آن است که اطلاعات را پس از بررسی از نظر کامل و سالم بودن به پروسس ارسال نماید و به عنوان مثال وقتی یک فایل معمولی را با فرمان های متعارف می خوانید سیستم عامل تمام کنترل های لازمه را انجام می دهد و اگر سکتوری مشکل داشته باشد پیام لازمه ارسال و پروسس را قطع می نماید.

در حالت **raw mode** پروسس خودموظف است که بسیاری از کنترل های لازمه را در مورد **I/O** داشته باشد و سیستم عامل در صورت وجود هر اشکال در زمینه **I/O** کنترل را به پروسس داده و پروسس می تواند فعالیت های لازمه را انجام دهد.

بعضی اعتقاد دارند که این امکان زودتر در ویندوز بوده است ولی به هر حال کسانی که می خواهند ادمین لینوکسی باشند که پایگاه داده روی آن نصب است حتما باید این مطلب را تعقیب کنند.



- ۱- کارنامه به سیستم می گوید که می خواهم اطلاعاتی را از دیسک بخوانم.
- ۲- سیستم عامل پارامترها و مجوز ها را بررسی میکند اگر همه چیز درست بود به پارامترها را به درایور دیسک منتقل می کند.
- ۳- درایور پارامترها را برای کنترلر آماده می کند و آنها را برای استفاده کنترلر در حافظه مخصوص رجیسترها کنترلر قرار می دهد.
- ۴- کنترلر رجیسترها را خوانده و داده های موردنظر را از دیسک میخواند.
- ۵- این داده ها اگر سالم بود به بافر کارنامه منتقل می شوند.
- ۶- در نهایت داده ها به بافر کارنامه منتقل می شود.

این پروسه ای است که در **cooked mode** اتفاق می افتاد . بدی روشن بالا این است که اگر سکتور داخل دیسک خراب باشد هیچ اطلاعاتی به بافر کرنل منتقل نمی شود و به سیستم عامل می گوید نمی توانم سیستم عامل هم برنامه را **cancle** می کند (**I/O Error** می دهد) حسنه که مدد آمده (**cooked**) دارد این است که اگر نرم افزار دیگری مثل ثبت نام همان داده ها را بخواهد می تواند آنها را مستقیما از بافر کرنل بخواند و نیازی به خواندن دوباره دیسک نیست پس در **cooked mode** عملکرد سیستم بهبود پیدا می کند.

الان کیبورد در ویندوز و لینوکس **cooked** است و اگر **backspace** بزنید کاراکتر از بافر پاک می شود ولی در **vi** این طور نیست و از بیرون آورده و خودش چک می کند. 99 درصد نرم افزارها **cooked** هستند.

در واقع در raw mode کنترلر اطلاعات را مستقیماً به بافر کارنامه ارسال می کند. به همین خاطر فایل هایی که نمی توانیم آنها را بخوانیم نرم افزاری مثل norton commander می خواند چون در raw mode اجرا می شود و هرچه کنترلر بخواند را دریافت می کند البته کنترلر می گوید که اطلاعات مناسب نیست ولی خود نرم افزار آن را مدیریت می کند و هر چقدر که قابل دیدن باشد نمایش می دهد.

بعضی مواقع شکایت می کنند که مثلاً نرم افزاری را نصب کرده این و 16 گیگابایت رم پر شده است حتی با ارتقا به 32 گیگابایت پس از مدتی رم سیستم پر شده است دلیل این است که لینوکس اطلاعاتی را که بافر کرده تا حافظه دارد نگه می دارد و گرنه قدیمیترین را پاک می کند نرم افزار مورد بحث مشکلی نداشته و بافر را پر کرده است.

اگر بخواهید رم را خالی کنید باید اشاره گرهای بافر را دستکاری کنید که کار خیلی فنی و خطرناکی است. از لحاظ کار با رم دو نوع لینوکس داریم؛ اول جای برنامه ای که می خواهد از رم بیرون بریزد و برنامه دیگری در آن قرار دهد /dev/zero / می زند سپس برنامه جدید را منتقل می کند.

در نوع unsecure آدرس برنامه قدیمی را به عنوان فضای خالی اعلام کرده و به برنامه جدید می گوید در آنجا مستقر شود. در گذشته سیستم عامل همه برنامه ها را به یک قسمت رم منتقل می کرد و فضای خالی پیوسته به وجود می آورد.

خوبی حالت raw این است که برنامه دیگر kill نمی شود. در پایگاه داده دیسک اصلی را به صورت raw می خواند اگر خراب بود را می خواند و برنامه کنسل نمی شود. در حالت raw mode حتی backspace هم ثبت می شود.

raw mode نرم افزاری به نام check disk (fsck) داریم که مثل ندارد و اگر سیستم carsh کرد راهبر نگران این نیست fsck تصمیمی بر روی فایل گرفته و احتمالاً آن را حذف کند.

به همین خاطر سرور های پایگاه داده عمدها raw mode هستند چون حتی اگر پایگاه داده خراب شد application نباید از کار بیفتند.

```
[n.pardis@lpi ~]$ man raw
```

RAW(8)

RAW(8)

NAME

raw – bind a Linux raw character device

SYNOPSIS

raw **/dev/raw/raw<N>** <major> <minor>

raw **/dev/raw/raw<N>** **/dev/<blockdev>**

raw -q **/dev/raw/raw<N>**

raw -qa

DESCRIPTION

raw is used to bind a Linux raw character device to a block device. Any block device may be used: at the **time** of binding, the device driver does not even have to be accessible (it may be loaded on demand as a kernel module later).

raw is used **in** two modes: it either sets raw device bindings, or it queries existing bindings. When setting a raw device, **/dev/raw/raw<N>** is the device name of an existing raw device node **in** the filesystem.

```
[n.pardis@lpi dev]$ ls -l
total 0
crw-rw----. 1 root video      10, 175 Nov  4 15:35 agpgart
crw-rw----. 1 root root       10,  56 Nov  4 15:35 autoofs
drwxr-xr-x. 2 root root      640 Nov  4 16:09 block
drwxr-xr-x. 2 root root      80 Nov  4 16:09 bsg
drwxr-xr-x. 3 root root      60 Nov  4 15:35 bus
lrwxrwxrwx. 1 root root      3 Nov  4 15:35 cdrom -> sr0
drwxr-xr-x. 2 root root     2860 Nov  4 16:09 char
crw-----. 1 root root      5,   1 Nov  4 15:35 console
lrwxrwxrwx. 1 root root     11 Nov  4 15:35 core -> /proc/kcore
drwxr-xr-x. 3 root root      80 Nov  4 15:35 cpu
crw-rw----. 1 root root     10,  61 Nov  4 15:35 cpu_dma_latency
crw-rw----. 1 root root     10,  62 Nov  4 15:35 crash
drwxr-xr-x. 6 root root     120 Nov  4 15:35 disk
brw-rw----. 1 root disk     253,   0 Nov  4 15:35 dm-0
brw-rw----. 1 root disk     253,   1 Nov  4 15:35 dm-1
```

آنهايي که سمت چپشان b نوشته شده يعني block device هستند و آنهايي که c نوشته شده يعني در raw mode کار می کنند.

Iptables In Linux

یک نرم افزار امنیت شبکه است که در سطح کرنل اجرا شده و ترافیک ورودی و خروجی را کنترل می کند.

سیستم های packet filtering به دو دسته تقسیم می شوند دسته اول بدون حالت که بسته های جدآگانه را فیلتر می کنند و دسته دوم مبتنی بر حالت؛ مثل proxy server ها که وقتی مثلا می خواهید به یاهو وصل شوید IP شما invalid است و نمی توانید وصل شوید پس فریم شما به پروکسی سرور می رود و او خودش را به جای شما جا می زند اینها را یک جا باید یادداشت کند که اگر جواب برگشت بداند که به چه کسی انتقال دهد، خیلی وقت ها هکرها packet های بی ربط می فرستند.

این نوع امنیت بالاتری دارد و در عین حال حافظه بیش تری می گیرد و سرعت کم تری دارد. از هر 100 تا فایروال 99 تا لینوکسی است.

```
[n.pardis@lpi ~]$ rpm -qi iptables
Name        : iptables                               Relocations: (not relocatable)
Version     : 1.4.7                                  Vendor: Red Hat, Inc.
Release    : 4.el6                                 Build Date: Fri 07 Jan 2011
09:17:49 PM IRST
Install Date: Sat 22 Sep 2012 11:34:23 PM IRST      Build Host: x86-
003.build.bos.redhat.com
Group       : System Environment/Base               Source RPM: iptables-1.4.7-
4.el6.src.rpm
Size        : 719035                                License: GPLv2
Signature   : RSA/8, Tue 19 Apr 2011 03:31:10 PM IRDT, Key ID
199e2f91fd431d51
Packager   : Red Hat, Inc. <http://bugzilla.redhat.com/bugzilla>
URL        : http://www.netfilter.org/
Summary    : Tools for managing Linux kernel packet filtering
capabilities
Description :
The iptables utility controls the network packet filtering code in the
Linux kernel. If you need to set up firewalls and/or IP masquerading,
you should install this package.
```

و در زیر فایل های این سرویس را می بینیم که تعداد زیادی share object دارد که در زیر lib قرار دارند :

```
[n.pardis@lpi ~]$ rpm -qil iptables
Name        : iptables
Version     : 1.4.7
Release    : 4.el6
09:17:49 PM IRST
Install Date: Sat 22 Sep 2012 11:34:23 PM IRST      Build Host: x86-
003.build.bos.redhat.com
Group       : System Environment/Base      Source RPM: iptables-1.4.7-
4.el6.src.rpm
Size        : 719035                      License: GPLv2
Signature   : RSA/8, Tue 19 Apr 2011 03:31:10 PM IRDT, Key ID
199e2f91fd431d51
Packager    : Red Hat, Inc. <http://bugzilla.redhat.com/bugzilla>
URL         : http://www.netfilter.org/
Summary     : Tools for managing Linux kernel packet filtering
capabilities
Description :
The iptables utility controls the network packet filtering code in the
Linux kernel. If you need to set up firewalls and/or IP masquerading,
you should install this package.
/bin/iptables-xml
/etc/rc.d/init.d/iptables
/etc/sysconfig/iptables-config
/lib/libip4tc.so.0
/lib/libip4tc.so.0.0.0
/lib/libip6tc.so.0
/lib/libip6tc.so.0.0.0
....
```

iptable را نمی توان متوقف کرد فقط می توان آن را غیرفعال کرد . اگر webserver یا mailserver یا ps غیر فعال کنیم از معلوم می شود ولی iptables را نمی بینیم . بعضی از سرویس ها فقط آماده سازی می کنند و می روند . لینوکس که بالا می آید در کرنل ما table نداریم که بسته ای را فیلتر کند اینها در حافظه قرار دارد و کامپیوتر را که خاموش کنیم می رود. در بالا آمدن سیستم سرویس iptables می آید فایلی را می خواند ، در کرنل تغییراتی اعمال می کند و خارج می شود.

```
[root@lpi n.pardis]# less /etc/sysconfig/iptables-config|grep -v "#"
IPTABLES_MODULES="nf_conntrack_netbios_ns"

IPTABLES_MODULES_UNLOAD="yes"

IPTABLES_SAVE_ON_STOP="no"

IPTABLES_SAVE_ON_RESTART="no"

IPTABLES_SAVE_COUNTER="no"

IPTABLES_STATUS_NUMERIC="yes"

IPTABLES_STATUS_VERBOSE="no"

IPTABLES_STATUS_LINENUMBERS="yes"
```

این فایل توضیحات زیادی دارد که با grep فقط کدها را بیرون کشیده ایم.

اگر در ps -aef دقت کرده باشید syslog با آپشن m-آمده است در حالی که ما آن را با -m- اجرا نکرده ایم:

less /etc/sysconfig/syslog

سرویسی دچار مشکل شد در گوگل جستجو می کنیم و مثلا باید با -k- آن را بالا بیاوریم. برای این کار سرویس در sysconf یک فایل متناظر دارد.

```
[root@lpi n.pardis]# less /etc/sysconfig/rsyslog

# Options to syslogd
# syslogd options are deprecated since rsyslog v3
# if you want to use them, switch to compatibility mode 2 by "-c 2"
SYSLOGD_OPTIONS="-c 4"

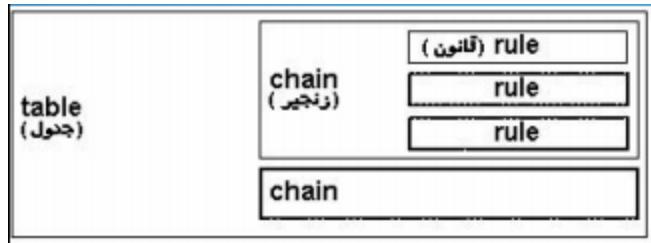
/etc/sysconfig/rsyslog (END)
```

less /etc/sysconfig/*

ممکن است در سازمان ها بگویند که iptables را که راه اندازی کردید کاری کنید که هر 5 دقیقه restart شود.

زیر sysconfig پیکربندی سرویس ها قرار دارد نه کرنل.

مجموعه ای از جدول هاست که خود شامل chain هایی هستند که هر rule شامل تعدادی iptables است.



فرمت دستورات iptables خیلی پیچیده است و بهتر است آنها را همراه خود داشته باشید!

به عنوان مثال الان در فرودگاه انگلیس یک طرف نوشتند خروجی اروپایی ها و طرف دیگر مخصوص سایر کشورها ، حالا در مسیر خروج اول بازرسی می کنند بعد پاسپورت را می بینند و ... و هر روز ممکن است این زنجیره تغییر کند و چیزی از آن کم و یا به آن اضافه شود. این دقیقاً اتفاقی است که ممکن است برای iptables packet در حالی که همان مسافران هستند.

فرق iptables با ACL (access control) است که acl بیشتر برای درون کامپیوتر استفاده می شود و شبیه squid است.

با استفاده از فرمان iptables می توان وضعیت تمامی اتصالات دیواره آتش را همزمان با تغییر آنها و به صورت لحظه ای مشاهده نمود. در واقع این فرمان وضعیت جدول فیلتر ، قانون های ایجاد شده در این جدول و هدفهای اعمال شده روی بسته ها را نمایش می دهد.

را در زیر لیست کرده ایم:

```
[root@lpi n.pardis]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source          destination
ACCEPT     all  --  anywhere        anywhere         state
RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere        anywhere
ACCEPT     all  --  anywhere        anywhere
ACCEPT     all  --  anywhere        anywhere
ACCEPT     tcp  --  anywhere        anywhere        state NEW tcp
dpt:nfs
ACCEPT     udp  --  anywhere        anywhere        state NEW udp
dpt:netbios-ns
ACCEPT     udp  --  anywhere        anywhere        state NEW udp
dpt:netbios-dgm
ACCEPT     tcp  --  anywhere        anywhere        state NEW tcp
dpt:netbios-ssn
ACCEPT     tcp  --  anywhere        anywhere        state NEW tcp
dpt:microsoft-ds
ACCEPT     udp  --  anywhere        anywhere        state NEW udp
dpt:netbios-ns
ACCEPT     udp  --  anywhere        anywhere        state NEW udp
dpt:netbios-dgm
ACCEPT     tcp  --  anywhere        anywhere        state NEW tcp
dpt:ssh
REJECT    all  --  anywhere        anywhere        reject-with
icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source          destination
ACCEPT     all  --  anywhere        anywhere         state
RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere        anywhere
ACCEPT     all  --  anywhere        anywhere
ACCEPT     all  --  anywhere        anywhere
REJECT    all  --  anywhere        anywhere        reject-with
icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
```

less /etc/init.d/iptables

این سرویس iptables است و مثلا در اوایل فایل قید شده است که اگر execution permision نداشته باشد اصلا بالا نمی آید. در تابع گفته اگر اصلا start نشده است stop نکند.

توصیه می شود که راجع به این فایل و محتویاتش تحقیق کنید چون اگر در جایی که نمی توانند هزینه فایروال را تامین کنند استخدام شوید گرفتار این دستور می شوید.

اول option خودش را می خواند می فهمد که باید از یک جایی اطلاعات را بخواند؛ خط به خط خوانده و در کرنل قرار می دهد.

اگر یک بسته به خادم برسد می توان تصمیمات زیر را گرفت:

- بسته را دور انداخت و پیام خطا را به مبداء اعلام نمود. Reject
- بسته را دور انداخت و به مبداء اعلام ننمود (سکوت) Deny
- بسته را قبول نموده و فعالیت مورد نظر را انجام داد. Accept
- بسته را به جای دیگر ارسال نمود. Forward

تذکر: توجه کنید که مثال های زیر ناقص بوده و فقط جنبه آموزشی دارد.

مثال 1:

به راهبر سایتی ماموریت داده شده که ترافیک ورودی از آدرس 100.100.100.1 را کنترل نموده تمام پاکت های دریافتی از این آدرس را دور بریزد.

راهبر سایت می داند که آدرس مبداء چیست و چون به او اعلام نشده بعد از دور انداختن بسته چه کند دو راه حل در پیش دارد:

1. بسته را دور ریخته و در پاسخ به مبداء سکوت کند

iptables -s 100.100.100.1 -j Drop

j- در اینجا به معنی DROP است. DROP به معنی حذف پیام بدون اطلاع به فرستنده می باشد.

2. بسته را دور ریخته و به مبداء پیام خطایی را ارسال نماید

iptables -s 100.100.100.1 -j Reject

به معنی حذف پیام و اطلاع به فرستنده می باشد. Reject

Linux Kernel

در آمریکا اجازه نمی دهنده یک تجارت و یک شرکت تک قطبی شود به همین خاطر کنگره آمریکا به تیم لینوز توروالدز و کسانی که هسته لینوکس را توسعه می دهند کمک مالی می کنند حتی سیستم های کاخ سفید هم بر پایه لینوکس است.

دیگر شرکت هایی که نرم افزار open source تولید می کنند اکثرا از راه پشتیبانی کسب درآمد می کنند مثلاً شرکت داده پردازی لینوکسی به نام کارآمد دارد که به قیمت 3500 تومان به فروش می رسد ولی از راه پشتیبانی آن کسب درآمد می کند.

روز به روز در حال ضعیف تر شدن است و یونیکس کارها کم کم لینوکس کار می شوند. البته خیلی از آدم های فنی عقیده دارند یونیکس از لینوکس پایدارتر (stable) است.

لینوکس کرنل یک پارچه دارد (monolithic) ولی در این بخش الان دو جناح فعالیت می کنند یکی توروالدز و همکارانش که معماری یک پارچه را دنبال میکنند و دیگری tanenbaum که بر استفاده از معماری micro-kernel اصرار دارد.

در نسخه 2.2 global spilock از هسته لینوکس حذف شد در این روش اگر پروسسی وسیله ای را احتیاج داشت و وسیله lock بود پروسس وارد حلقه می شود که هر چند وقت یک بار چک کند که وسیله آزاد شده یا نه.

یک سری کامپیوتر رومیزی وارد ایران شده که وقتی لینوکس یا حتی ویندوز روی آن نصب کنید با panic ، spin lock می دهد بررسی کردیم یکی از Ic های آن مشکل داشت.

کرنل معمولی آدرس فیزیکی بالای 4GB را پشتیبانی نمی کند که از ورژن 2.6 به بعد این مشکل مرتفع شده است.

وقتی شما نسخه ای مثل ubuntu را دانلود می کنید در واقع بسته ای را نصب می کنید که سازندگانش هر چیزی که فکر کرده اند لازم است کامپایل کرده اند و به صورت فایل قابل اجرا در بسته گنجانده اند.

برای دریافت کد هسته لینوکس باید به سایت kernel.org مراجعه کنید که حجمی زیر 100MB دارد ولی مثلاً توزیعی مثل دبیان 7 تا DVD است! و لینوکسی هم داریم که یک فلاپی است.

با دستور ls pci می توان لیست اکثر سخت افزارهای سیستم را دید ولی دلیلی ندارد همه را نشان بدهد چون لینوکس که بالا می آید Roll Call می کند یعنی می گوید که همه بردها خودشان را معرفی کنند بردهی هم خودش را معرفی می کند ولی درایور ندارد.

PANIC دلایل به وجود آمدن

به مجرد اینکه در حافظه fault اتفاق بیفتند یکی از روتین های کرنل چک میکند که از خودی ها بوده (kernel mode) یا غریبه ها (user mode) . اگر غریبه باشد نرم افزار را بیرون می گذارد ولی اگر در کرنل باشد چاره ای نیست جز اینکه سیستم را down کند.

می توان این روتین را برداشت تا سیستم panic ندهد که البته کار درستی نیست. کرنل خیلی وقت ها مثل گریه عمل می کند گریه هر قطعه گوشته که می خورد چک می کند که گوشت باشد! کرنل اکثر اوقات که می خواهد جدول هایش را به روز کند به گذشته اش نگاه می کند مثلاً می بیند 40- نفر login کرده اند؛ یعنی جدول ها به هم ریخته اند، خیلی وقت ها که ادامه دادن، وضع را بدتر می کند سیستم crash می کند.

crash سیستم را آنالیز می کند که رفع مشکل بستگی به دانش سخت افزاری شما دارد که مثلا وقفه ها را بدانید و یا با دستورهای زیر آشنا باشید. که در ایران دو سه نفر بیشتر نیستند که اطلاعات crash سیستم را بتوانند دنیا کنند.

حالا می خواهیم بررسی کنیم که چگونه عملا یک کرنل را up کنیم. ابتدا کد منبع کرنل را دانلود کرده و با دستور tar آن را در جایی extract کنید. بهتر است پس از کامپایل به انتهایی اسم کرنل نام خودتان را اضافه کنید که دیگران بدانند این کرنل استاندارد نیست و تغییراتی در آن اعمال شده است و گرفته ممکن است برنامه ای را اجرا کند که کار نکند.

پیگر بندی هسته

ابتدا فایل فشرده حاوی کد هسته لینوکس را دانلود کنید سپس با دستور tar آن را باز کنید. در مرحله بعد زیر دایرکتوری ساخته شده که فایل های هسته در آن قرار گرفته اند رفته و دستورات زیر را اجرا کنید:

```
[root@lpi linux-2.6.9]# make config
HOSTCC scripts/basic/fixdep
scripts/basic/fixdep.c: In function traps:
scripts/basic/fixdep.c:368: warning: dereferencing type-punned pointer
will break strict-aliasing rules
scripts/basic/fixdep.c:370: warning: dereferencing type-punned pointer
will break strict-aliasing rules
...

```

در ادامه صدها سوال از شما پرسیده می شود که از قبیل با توجه به نیازهایتان باید جواب آنها را در نظر بگیرید البته خیلی از اینها فنی است. مثل امکان IPC که برای موقعی استفاده می شود که دو نرم افزار بخواهند در داخل سیستم با هم صحبت کنند.

[n.pardis@lp1 ~]\$ man ipcrm

IPCRM(1) Linux Programmers Manual **IPCRM(1)**

NAME

ipcrm - remove a message queue, semaphore **set** or shared memory **id**

SYNOPSIS

ipcrm [**-M** *key* | **-m** *id* | **-Q** *key* | **-q** *id* | **-S** *key* | **-s** *id*] ...

deprecated usage

ipcrm {**shm|msg|sem**} *id...*

DESCRIPTION

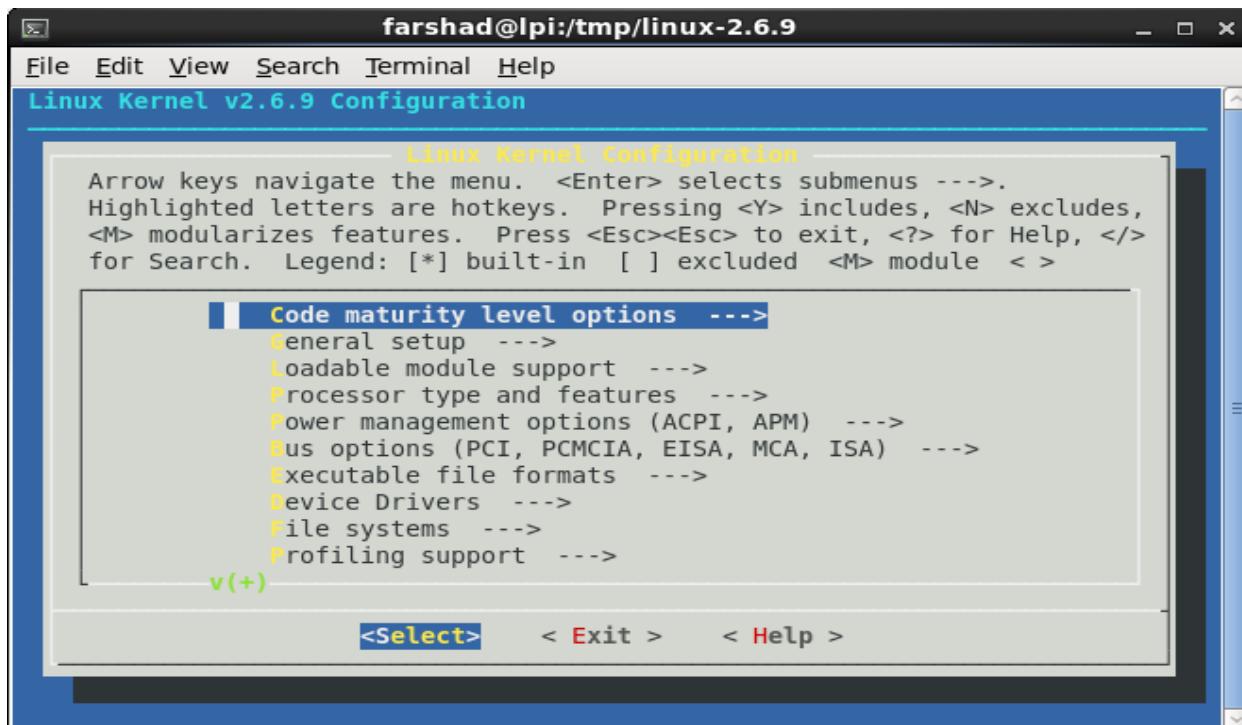
ipcrm removes System V interprocess communication (IPC) objects and associated data structures from the system. In order to delete such objects, you must be superuser, or the creator or owner of the object.

System V IPC objects are of three types: shared memory, message queues and semaphores. Deletion of a message queue or semaphore object is immediate (regardless of whether any process still holds an IPC identifier **for** the object). A shared memory object is only removed after all currently attached processes have detached (**shmdt(2)**) the object from their virtual address space.

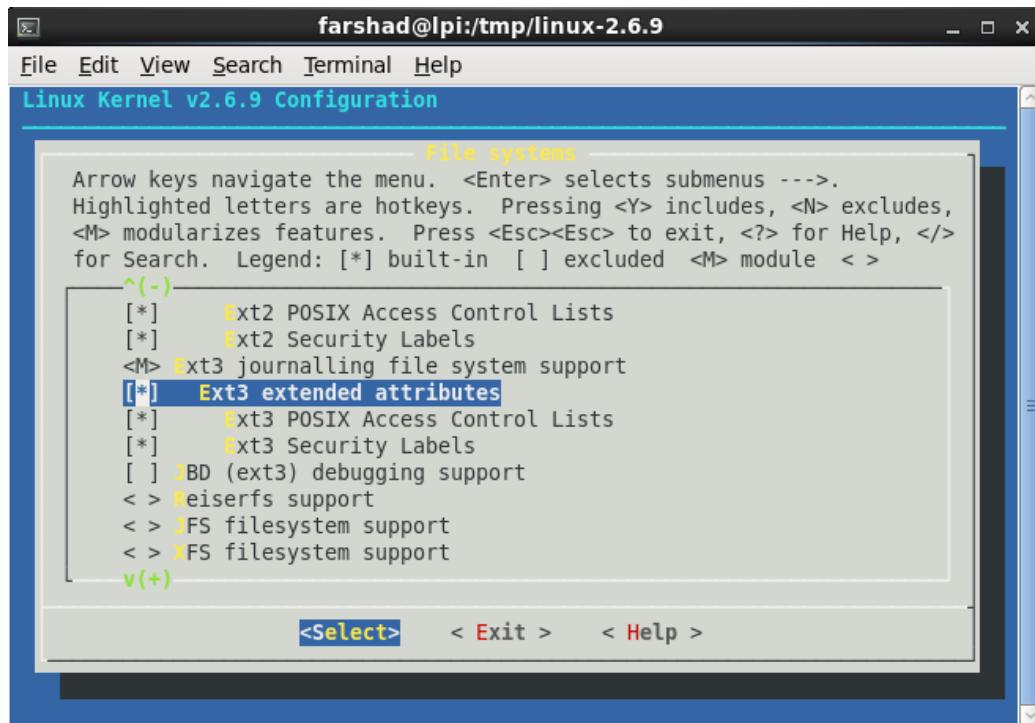
انتخاب یا عدم انتخاب امکاناتی که در این مرحله از شما پرسیده می شود در کاهش یا افزایش حافظه مورد نیاز سیستم نهایی دخیل است.
یکی از امکانات `sysctl support` که جدول های کرنل را نشان می دهد:

```
[n.pardis@lpi ~]$ sysctl -a
kernel.sched_child_runs_first = 0
kernel.sched_min_granularity_ns = 1000000
kernel.sched_latency_ns = 5000000
kernel.sched_wakeup_granularity_ns = 1000000
kernel.sched_tunable_scaling = 1
kernel.sched_features = 3183
kernel.sched_migration_cost = 500000
kernel.sched_nr_migrate = 32
kernel.sched_time_avg = 1000
kernel.sched_shares_window = 10000000
kernel.timer_migration = 1
kernel.sched_rt_period_us = 1000000
kernel.sched_rt_runtime_us = 950000
kernel.sched_compat_yield = 0
kernel.sched_autogroup_enabled = 0
kernel.sched_cfs_bandwidth_slice_us = 5000
```

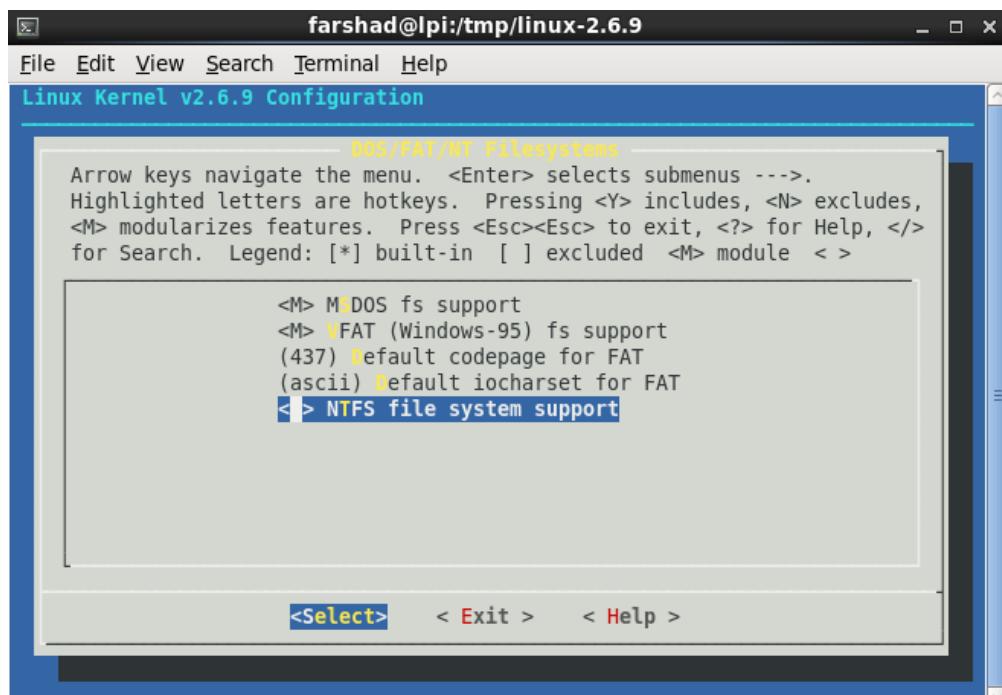
به دلیل زیاد بودن سوالات از ذکر همه آنها خودداری می کنیم. امکان دیگر استفاده از نصب گرافیکی است که با دستور `make menuconfig` پنجره زیر ظاهر می شود: اگر بخواهید دفعه اولی که لینوکس را نصب می کنید با سورس هسته آن را بالا بیاورید (در واقع از اول هسته را خودتان کامپایل کنید) باید در گوگل عبارت `linux from scratch LFS` را جستجو کنید



ما در lsattr و chattr و lpi1 را درس ندادیم چون هر لینوکسی اینها را ندارد:



گزینه پشتیبانی NTFS را نمی زند ولی ubuntu redhat می زند:



دلیل ردهت هم این است که می گوید ما ntfs را نمی شناسیم و با مهندسی معکوس فهمیدیم و اگر بگذاریم و ntfs کاربری دجاج مشکل شود از شرکت ناراضی می شود و چند سال پیش هم نامه ای به ردهت رسید که در لینوکس ntfs استفاده کردم ntfs به هم ریخته و ویندوز بالا نمی آید. ردهت هم این امکان را از سیستم عامل خود برداشت و گفت هر کس می خواهد خودش اضافه کند. البته دستور make xconfig هم محیط گرافیکی نصب را بالا می آورد.

مايكروسفت برای خودش فایل سیستم دارد؛ اگر fat و vfat ضعیف است، NTFS فوق العاده قوی است لینوکس هم آن را ساپورت می کند ولی منبع و layout آن را نداریم. لینوکس کارها با مهندسی معکوس پیدا کرده اند و به صورت read only روی لینوکس گذاشته اند و در گام بعدی read/write گذاشتند.

پس از آن که مشکلاتی با ntfs اعلام شد چند تا از کمپانی های بزرگ هم read و write آن را حذف کردند ولی desktop ادامه دادند.

کم کم گروهی تشکیل شد و NTFS-3G (نسل سوم) نوشته شد که می توانید دانلود کنید؛ این سیستم در user mode کار می کندو پارتیشن را در raw mode می بیند.

نکته: هم برای بالا رفتن اطلاعات عمومی و هم اعلام آخرین نسخه های نرم افزارها در گروه های آنها عضو شوید.

ntfs-3g مثل samba و wine متن باز است ولی نسخه پولی هم دارد که امکانات بیشتری دارد.

نرم افزار wine می تواند نرم افزارهای ویندوزی را اجرا کند:

man wine

نیزی که در کامپایل کرنل انتخاب می کنید read only است و اصلاً احتیاجی به کامپایل کرنل نیست چون مازول جدایی است.

init

وقتی برای تغییر runlevel دستور init 3 را اجرا می کنید در رم علاوه بر init اولیه 3 هم قرار می گیرد که این وضعیت را به هم می ریزد. روالی که init برای حل این مشکل انجام میدهد بعداً اگر بخواهید نرم افزار تولید کنید در طراحی آن به کار می آید که اگر یک کپی دیگر از برنامه داخل رم شود قبلي از کجا بفهمد و این در طراحی نرم افزار های سنگین استفاده می شود.

این تیپ نرم افزارها وقتی اجرا می شوند چک می کنند که پدرشان کیست . پدر init اولیه سیستم عامل با pid برابر با 0 است . وقتی init 3 را اجرا می کنیم می بیند که پدرش سیستم عامل نیست (bash fork کرده است) متوجه می شود که یک init از قبل وجود دارد به او اطلاع می دهد که مرا که یک کپی از تو هستم! با پارامتر 3 صدا کرده اند و خارج می شود سپس init اصلی runlevel را عوض می کند.

نرم افزارهایی که در حافظه مستقر هستند اول سیگنال ارسال می کنند و بعد اطلاعات رو بدل می کنند . بدون سیگنال سیستم عامل وجود ندارد و سیگنال با داده دادن اشتباه است چون ممکن است سیگنال نرسد و اطلاعات لوس شود .

init به نرم افزارهایی که در runlevel جاری در حال اجرا هستند ولی در runlevel بعدی حضور ندارند با سیگنال اعلام می کند و 5 ثانیه بعد kill می شود.

در فایل inittab ، برچسب هایی که با حروف الفبا شروع می شوند runlevel را عوض نمی کنند بلکه در ترکیب با init مدخلی را که با آن مطابقت دارد را اجرا می کند.

این کار در شهرداری شده بود که دستوراتی مثل تعریف روتر خیلی سخت است پس در inittab مدخلی برای آن درست می کنیم و به اپراتور می گوییم که مثلاً برای تعریف روتر init a را اجرا کند.

این نرم افزارها فرزند init هستند نه bash ، پس اگر logout کنید kill نمی شود مثلا اگر در حال پشتیبان گیری از سیستم باشید و logout کنید kill نمی شود در حالی که batch file پس از خروج kill می شود.

init q : به init می گوید که فایل inittab را دوباره بخواند.

single user : برو به init s

تفاوت آن با 1 این است که اگر به 1 runlevel بروید خیلی از فایل سیستم ها (کاربرها) umount می شوند ولی در init s چیزی umount نمی شود.

s برای maintain به کار می رود که کاربرهای دیگر نفهمند و اختلال ایجاد کنند.

یک سوال گمراه کننده: ماشین با چه runlevel یی بالا آمده است؟

who -r و runlevel ، جاری را می دهد و جواب نیست. بلکه باید در inittab initdefault را نگاه می کنیم.

معمولًا در سازمان ها default را 1 میگذارند سیستم که بالا آمد چک لیستی را پر می کنند و بعد به runlevel مورد نظر می رود.

که این چک لیست سوالاتی از قبیل اینکه آیا همه backupها را برگرداندید یا سیستم خنک کننده روشن است.

نکته: سیگنال poweroff را حفظ کنید که 27 است.

تمرین: کاربرد دکمه sys rq در لینوکس چیست؟

می توانیم با کلید های ترکیبی بین ترمینال های مختلف لینوکس switch کنیم 6 تا محیط text داریم و هفتمنی هم برای ترمینال گرافیکی به کار می رود (ترکیب Alt و کلید های F1 تا F7) ولی در اصل 72 تا tty داریم که با دیگر کلیدها قابل دسترسی هستند.

[n.pardis@lpi rc3.d]\$ cd /etc/rc3.d			
K01certmonger	K50snmptrapd	S10network	S26udev-post
K01matahari-host	K60nfs	S11auditd	S28autofs
K01matahari-network	K69rpcsvcgssd	S11portreserve	S50bluetooth
K01matahari-service	K73ypbind	S12rsyslog	S55sshd
K01matahari-sysconfig	K74ntpd	S13cpuspeed	S56xinetd
K01smartd	K75ntpdate	S13irqbalance	S80postfix
K02oddjobd	K75quota_nld	S13rpcbind	S82abrt-ccpp
K03rhnsd	K80kdump	S15mdmonitor	S82abrted
K05wdaemon	K80sssd	S22messagebus	S82abrt-oops
K10psacct	K84wpa_supplicant	S23NetworkManager	S85qpidd
K10saslauthd	K87restorecond	S24avahi-daemon	S90crond
K12mailman	K89rdisc	S24nfsllock	S91nmb
K15httpd	K95firstboot	S24rpcgssd	S91smb
K15matahari-broker	S01sysstat	S24rpcidmapd	S95atd
K30spice-vdagentd	S021vm2-monitor	S25cups	S97rhsmcertd
K50dnsmasq	S03vmware-tools	S25netfs	S99local
K50netconsole	S08ip6tables	S26acpid	
K50snmpd	S08iptables	S26haldaemon	

اول kها اجرا می شوند بعد sها . همچنین نرم افزارهایی مثل postgresSQL و mysql (در واقع خیلی از برنامه های پایگاه داده)

را که نصب می کنیم اکثرا 99s می شوند اینقدر منتظر می شوند تا همه سرویس ها بالا بیایند.

دستور service روی بعضی از توزیع ها (fedora و redhat) وجود دارد ، به عنوان تمرین اسکریپتی بنویسید که این دستور را وارد سیستم کند.

راهنمایی:

```
[root@lpi ~]# /etc/rc.d/init.d/httpd restart
```

یکی از سرویس های تحت /etc/init.d سرویس NIS یا Network Information Service است که قبلا با نام YP شناخته می شد. اما چون کلمه yellow pages توسط مخابرات انگلیس ثبت شده است لینوکس مجبور است این کلمه را استفاده نکند ولی هنوز در دستوراتش هست.

برنامه getty مسئول گوش دادن به فعالیت ها در پورت و ارائه یک login prompt در صورت اطلاع از فعالیت است:

```
[n.pardis@lpi ~]$ man mingetty

MINGETTY(8)                               Linux Programmers Manual
MINGETTY(8)

NAME
    mingetty - minimal getty for consoles

SYNOPSIS
    mingetty [--noclear] [--nonewline] [--noissue] [--nohangup] [--nohost-
    name] [--long-hostname] [--loginprog=/bin/login] [--nice=10]
    [--delay=5] [--chdir=/home] [--chroot=/chroot] [--autologin
    username]
    [--loginpause] tty

DESCRIPTION
    mingetty is a minimal getty for use on virtual consoles.
Unlike
    agetty(8), mingetty is not suitable for serial lines. I
recommend
    using mgetty(8) for this purpose.

OPTIONS
    --noclear
        Do not clear the screen before prompting for the login name
(the
        screen is normally cleared).
```

جایگزین init

سرویس ها را به محض این که نرم افزارهای وابسته آنها (dependencies) اجرا شدند، شروع می کند. این سیستم موجب بالانس بهتر CPU و I/O می شود.

در سرورها و سازمان هایی که جدی که تعداد زیادی سرویس بالا می آید سیستم را که boot می کند خیلی طول می کشد عملیاتی شود و یکی از دلایل عمدۀ آن init است که فایل inittab را پشت سرهم (sequential) می خواند؛ این فایل در شهرداری منطقه 4 بدون توضیحات 500 خط بود. اما در initng سرویس ها را دسته بندی می کنند و آنها باید که به هم وابسته نیستند به صورت موازی (parallel) اجرا می شوند ولی وابسته ها sequential اجرا می شوند. اگر سیستم میل سرور و network داشته باشد بهبود عملکرد ملموس تر است.

خیلی وقت ها error روی کنسول می آید ادمین نمی داند از کجا آمده است، چند سال پیش این خطاهای جمع آوری شد و در غالب کتابی به چاپ رسید و شل اسکریپتی هم نوشته شد که خطای را در آن paste می کردیم و می گفت متعلق به کجاست.

```
[root@lpi ~]# type init
init is /sbin/init
[root@lpi ~]# cd /sbin/
[root@lpi sbin]# strings -15 15 init|less

/lib/ld-linux.so.2
_Jv_RegisterClasses
nih_timer_add_timeout
nih_alloc_real_set_destructor
nih_error_raise_error
nih_child_add_watch
nih_signal_add_handler
nih_str_array_copy
nih_log_set_priority
nih_hash_lookup
nih_str_array_add
nih_signal_reset
nih_config_parse_block
nih_hash_string_hash
nih_log_set_logger
nih_error_push_context
nih_io_shutdown
nih_signal_set_handler
nih_tree_next_post_full
nih_main_loop_add_func
nih_error_raise_printf
nih_hash_string_key
nih_signal_handler
```

اینها init و بهتر است دسته بندی شده و پرینت شود.
سیستم را که down می کنیم init به همه سیگنال 15 (SIGTERM) می دهد (نگهبان از بلندگو اعلام می کند که دانشگاه را ترک کنید) و پس از چند ثانیه سیگنال kill را می فرستد (9).

گوگل در سال یک دقیقه هم down نیست خطا که می آید نباید دنبال این بگردید که از کجا آمده است بلکه باید یک پایگاه داده درست کنید ولی نمی توان آن را کتاب کرد چون کرنل ها با هم تفاوت دارند در حالی که در data center ها معمولاً هر روز کرنل عوض نمی شود. یکی از خطاهای init، %s respawning too fast, stopped می باشد این خطا را می توان به عمدتاً به وجود آورد به این صورت که یک مودم اکسترنال به سیستم متصل کنید و به سرعت آن را power on و power off کنید این خطا را مشاهده می کنید. برای مودم نرم افزاری را در حافظه می گذارد. مودم هایی که دارند callerID را گرفته و شماره تلفن را بر می گردانند.

```
[root@lpi sbin]# man agetty

AGETTY(8)
AGETTY(8)

NAME
    agetty - alternative Linux getty

SYNOPSIS
    agetty [-8ihLmnUw] [-f issue_file] [-l login_program] [-I init]
    [-t timeout] [-H login_host] port baud_rate,... [term]
    agetty [-8ihLmnw] [-f issue_file] [-l login_program] [-I init]
    [-t timeout] [-H login_host] baud_rate,... port [term]

DESCRIPTION
    agetty opens a tty port, prompts for a login name and invokes
the
    /bin/login command. It is normally invoked by init(8).

    agetty has several non-standard features that are useful for hard-
wired
    and for dial-in lines:

    o      Adapts the tty settings to parity bits and to erase, kill,
end-
name.
    or
    following
    #,
```

به مجرد اینکه شما مودم را وصل می کنید نرم افزار مودم به مودم می گوید که power on شود . سیگنال به respawn می رود و تا خاموش می کنید نرم افزار kill می شود اگر این کار را خیلی تکرار کیم init روی کتسول می نویسد که من خیلی اینها را کرده ام و 5-7 دقیقه آن را disable می کند. کسانی که data center را اینکه مودم های زیادی به لینوکس متصل است زیاد با این خطای مواجه می شوند.

حتی موس را اگر به سرعت تکان دهید error می دهد نرم افزاری باید به برد vga بگوید که فلاش را تکان بددهد برای اطلاعات بیشتر عبارت linux buffer overflow mouse را در گوگل جستجو کنید.

```
[n.pardis@lpi dev]$ ls tty*|less
```

```
tty
tty0
tty1
tty10
tty11
tty12
tty13
tty14
tty15
tty16
tty17
tty18
tty19
tty2
...
tty6
tty60
tty61
tty62
tty63
tty7
tty8
tty9
ttyS0
ttyS1
ttyS2
ttyS3
```

از ترمینال های متعدد لینوکس می توانیم به وسیله کلیدهای ترکیبی مختلف با F₁ ها استفاده کنیم از جمله Alt یا Ctrl+Alt و

این S₁ ها در tty ها مودم است . ما در لینوکس هر تعداد که بخواهیم مودم وصل کنیم S مخفف سریال است . بردهایی قبلا به فروش می رسید که بانک صادرات هم استفاده می کرد همچنین این برد را dial up سرویس می دادند داشتند . برد را داخل ماشین می گذارند که 200 تا سیم تلفن مانند از آن خارج می شود که هر کدام یک سریال پورت است.

لینوکس به صورت پیش فرض ، 80-70 تا مودم ساپورت می کند. در شهرداری اینترنت مستقل و پرسرعت ماهواره ای داشتیم که بعد از ظهرها که اداره تعطیل بود 1000 تا تلفن هم داشتیم که بعد از ظهر به برد سریال وصل می شد و کارمندان شهرداری می توانستند به صورت رایگان از منزل به لینوکس شهرداری متصل شوند.

الان خیلی از نرم افزارها رویکرد event base دارند یک برنامه که kill می شود یک event است در fedora و ubuntu که فدایی های redhat و debian و initrd کم رنگ شده است مثلا در فدورا یک خط است و آن هم کامنت است نوشته که این را گذاشته ایم تا بدانید باید جای دیگری بروید اوبونتو حتی حتی این کار را هم نکرد.

جایگزین هم تقریبا به این صورت است که هر خط فایل initrd یک فایل جداگانه چند خطی شده است . دلیل هم این است که اگر initrd خراب شود احتمال بالا نیامدن سیستم زیاد است حالا اگر یکی از این فایل ها خراب شود مشکلی پیش نمی آید.

دلیل اینکه ما این فایل ها را درس ندادیم این است که او لا امتحان lpi بین المللی مد روز نیست ثانیا این هنوز از بوته آزمایش در نیامده است و ما فدورا و اوبونتو را به عنوان سرور قبول نداریم.

بررسی پروتکل FTP

برای حل مسائل مربوط به نشر اسناد و نرم افزار به نحوی که افراد بتوانند به آنها به سادگی از طریق سایر سیستم های کامپیوتری دسترسی پیدا نمایند ارائه شد.

برنامه های به اشتراک گذاری فایل ها نظیر SAMBA و NFS ابزارهای فوق العاده ای برای به اشتراک گذاری فایل ها و دایرکتوری ها بر روی یک شبکه خصوصی می باشند . برای سازمان هایی که به اشتراک گذاری تعداد فراوانی فایل برروی شبکه عمومی نیاز دارند نرم افزار سرور FTP ابزاری مستحکم تر برای به اشتراک گذاری فایل ها و محافظت از آنها می باشد.

خدماتی ftp را می توان به صورت زیر عملیاتی نمود:

- مستقل (standalone) و تحت نظر init
- تحت نظر و مدیریت xinetd

این سرویس از دو پورت استفاده می کند ؛ پورت 20 برای رد و بدل نمودن اطلاعات (فایل) و پورت شماره 21 برای فرمانها (help) و cd از طریق پورت 21 و put و get از پورت 20 انجام می شود)

وقتی سرویس ftp را راه اندازی می کنید در سیاست گذاری های سازمان باید تکلیف کنترل هویت (identification) را مشخص کند. کنترل هویت در ابتدای کار و از طریق پورت 21 انجام می شود.

```
[n.pardis@lpi dev]$ less /etc/services

# /etc/services:
# $Id: services,v 1.48 2009/11/11 14:32:31 ovasik Exp $
...
ftp-data      20/tcp
ftp-data      20/udp
# 21 is registered to ftp, but also used by fsp
ftp          21/tcp
ftp          21/udp      fsp  fspd
```

استفاده نمی کند . این را گذاشته اند که کسی نتواند از 20 udp برای نرم افزارش استفاده کند ؛ محدود نرم افزارهایی داریم که آنها فرق کند:

```
shell      514/tcp      cmd      # no passwords used
syslog    514/udp
```

بعد از نصب و اجرای ftp با دستور `ftp 127.0.0.1` آن را تست می کنیم اگر connection refused داد یعنی طرف مقابل هست و ارتباط برقرار است ولی ارتباط را قبول نمی کند. در یک دعوای قضایی بین پست بانک و پیمانکار سخت افزار این خطأ در وصل نشدن می آمد که نشان میدهد شبکه و سخت افزار درست است.

پس از نصب نرم افزاری VSFTP فایلی با نام `/etc/vsftpd.conf` تحت دایرکتوری `vsftpd.conf` تولید می شود که نیاز به اصلاحاتی دارد:

آیا کاربر Anonymous می تواند سرویس بگیرد؟

- `anonymous_enable=no`

آیا کاربر می تواند فایل بر روی سرور قرار دهد؟

- `write_enable=yes`

آیا می توان سرور را به صورت standalone راه اندازی کرد؟

- `listen=yes`

در `log` این سرویس ها ثبت می شود که چه id ای چه فایلی را خوانده یا نوشته است.

مجوزهای فایل و دایرکتوری استاندارد به عنوان یک محدود سازی دسترسی به فایلهای خاص ، حتی در سیستمهای فایل در دسترس به کار برده می شوند.

شما تنظیمی را به فایل `vsftpd.conf` اضافه نمایید تا بر روی چگونگی دانلود شدن فایل ها تاثیر بگذارد . برای فعال نمودن دانلودهای اسکس می توانید این ویژگی را به صورت زیر فعال نمایید:

- `ascii_download_enable=yes`

بدون اعمال این تغییر تمامی دانلودها در مد باینری انجام می شوند. در شهرداری روی یک sun solaris پیش فرض انتقال اسکی بود و عکس و فیلم و فایل های exe اگر در این مد ftp شوند ممکن است ناقص منتقل می شوند. چون `ctrl+z` مشخص کننده آخر فایل است. یکی از کارمندان قرار بود که هارد را فرمت کند و سیستم پرسنلی شهرداری را روی آن نصب کند . سیستم پرسنلی را با ftp به یک کامپیوتر دیگر منتقل کرد هارد را فرمت کرد ، دوباره با ftp سیستم پرسنلی را برگرداند ولی سیستم کار نمی کرد!

چون اسکی دانلود کرده و فایل های پرسنلی پر از عکس بوده است و اتفاقا در یکی از همین عکس ها کد باینری `ctrl+z` وجود داشته است و از یک جایی به بعد اصلا منتقل نشده است.

توصیه می شود که اطلاعاتی که ماشین روی کنسول نمایش می دهد حتما مطالعه کنید ؛ به عنوان مثال در ورود به ftp نمایش می دهد که در مد باینری است یا اسکی:

```
[n.pardis@lpi ~]$ ftp 127.0.0.1
Connected to 127.0.0.1 (127.0.0.1).
220 (vsFTPD 2.2.2)
Name (127.0.0.1:n.pardis):
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

برای اینکه مشاهده کنید دسترسی چه کسانی به سروز vsftpd منع شده است ، فایل vsftpd.user_list را بررسی نمایید.

اچر قصد دارید تا به تمامی کاربران توانایی نوشتن بر روی سرور ftp را ارائه نمایید . مقدار umask=022 مجوز فایل پیش فرض را هنگامی که یک کاربر محلی یک فایل را بر روی سرور ایجاد می کند ، تنظیم می نماید. (مقدار 022 سبب می گردد فایل ها با داشتن مجوز 644 ایجاد شوند که به کاربر اجازه می دهند تا مجوز خواندن و نوشتن داشته باشند و تمامی افراد فقط مجوز خواندن را دارند)

توانایی upload برای کاربران ftp ناشناس غیرفعال می باشد. اگر شما تمایل دارید که این ویژگی را فعال کنید ، خط زیر را به فایل vsftpd.conf اضافه نمایید:

- anon_upload_enable=yes

شما باید همچنین مطمئن شوید که دایرکتوری /var/ftp شامل یک یا تعداد بیش تری دایرکتوری با داشتن مجوزهای نوشتن برای کاربران anonymous می باشد . برای مثال شما ممکن است بخواهید یک دایرکتوری incoming را ایجاد کرده و مجوزهای آن را باز نمایید . (chmod 777 /var/ftp/incoming)

777 برای دایرکتوری است فقط x دارد و می تواند cd کندو برای اجرا باید anonymous login کند که anonymous نمی تواند.

timeout های زیر به صورت پیش فرض در داخل vsftpd تنظیم می شوند:

- accept_timeout=60
- connect_timeout=60

این دو پارامتر مدت زمانی که client پیش از توقف مجبور به برقراری یک اتصال می باشد را تعیین می کند.

- idle_session_timeout=600
- data_connection_timeout=120

در اگر login idle کرده باشیم و تا ده دقیقه هیچ کاری نکنیم قطع می شویم ولی پارامتر دوم تعیین می کند که اگر حین انتقال داده روتر میانی خاموش شود یا hub switch خراب شود اگر تا دو دقیقه روش نشود وارتباط برقرار نشود سرور ارتباط را قطع می کند.

نرم افزار هایی هستند که به صورت parallel ، ftp می کنند و انتقال سریع تر انجام می شود . روند کار به این صورت است که در ابتدا client از سرور می پرسد که اندازه فایل چقدر است سپس 10 تا client fork می کند به اولی می گوید که یک دهم اول فایل را بگیرد ، دومی ؛ یک دهم دوم و الی آخر. وقتی دانلود به پایان رسید این فایل ها را merge می کند. اگر هم وسط کار قطع شد پس از اوصل مجدد client به سرور می گوید که به اندازه حجم دانلود قبلی جلو برود و از آنجا به بعد ادامه می دهد.

یکی دیگر از پارامتر ها dirmessage_enable می باشد که مثلا در بد و ورود به یک دایرکتوری می توان به کاربر گفت که به دایرکتوری بازی ها خوش آمدید و در اینجا 50 بازی قرار دارد.

جهت تست vsftpd واینکه متوجه شویم که آیا نرم افزار از پورت 21 استفاده می کند و یا خیر از دستورات زیر استفاده می کنیم:

```
[root@lpi ~]# netstat -nap|grep 21|grep ftp
tcp          0      0 0.0.0.0:21                      0.0.0.0:*
LISTEN      14873/vsftpd
[root@lpi ~]# ps a-ef|grep -i ftp
root    14873      1  0 Nov11 ?            00:00:00 /usr/sbin/vsftpd
/etc/vsftpd/vsftpd.conf
```

با ftp نمی توان دایرکتوری جابه جا کرد.

پورت های ftp را می توان عوض کرد ولی باشد به کاربران اطلاع دهد.

یک نرم افزار رابط گرافیکی برای کار با ftp است که از طریق مورگر قابل دسترسی است. وزارت بهداشت و درمان به صورت آزمایشی از این نرم افزار در یزد استفاده می کند به این صورت که بیماران کارتی دارند که دکتر در کارتخوان می کشد و سوابق بیماری شخص را مشاهده می کند سپس اطلاعات جدید را می کشد در وزارت بهداشت ftp

یک توزیع update

اگر لایسنس ردهت را داشته باشید یا عضو گروه های توزیع خود باشید به محض به روز شدن نسخه نرم افزارها با ارسال ایمیل مطلع می شوید و به صورت گام به گام سیستمان را به روز نگه دارید.

وقتی update cd را می گذارید یک پروسه طولانی انجام می شود اول تمام نرم افزارهای موجود را به دست آورده و در جایی نگه می دارد:

```
[n.pardis@lpi ~] $ rpm -aqal less  
libxml2-python-2.7.6-4.el6.i686  
elfutils-libs-0.152-1.el6.i686  
ibus-qt-1.3.0-2.el6.i686  
device-mapper-1.02.66-6.el6.i686  
dejavu-fonts-common-2.30-2.el6.noarch  
system-config-printer-libs-1.1.16-22.el6.i686  
libselinux-utils-2.0.94-5.2.el6.i686  
vim-enhanced-7.2.411-1.6.el6.i686  
postfix-2.6.6-2.2.el6_1.i686  
basesystem-10.0-4.el6.noarch  
pygtk2-2.16.0-3.el6.i686  
diffutils-2.8.1-28.el6.i686  
systemtap-runtime-1.6-4.el6.i686  
foomatic-4.0.4-1.el6_1.1.i686  
comps-extras-17.8-1.el6.noarch  
system-config-date-1.9.60-1.el6.noarch  
libtar-1.2.11-16.el6.i686  
mouse tweaks-2.28.2-1.el6.i686  
crda-1.1.1_2010.11.22-1.el6.i686  
dhclient-4.1.1-25.P1.el6.i686  
gnome-session-2.28.0-18.el6.i686  
dmidecode-2.11-2.el6.i686  
samba-client-3.5.10-114.el6.i686
```

سپس در cd هم همین دستور را اجرا می کند و قبلی را از این کم می کند. نرم افزارهایی که تغییر کرده اند dependecy های آن را پیدا می کند و یک ماتریس dependency درست می کند . در خیلی از سازمان ها صلاح نیست نصب مجدد داشته باشند ؛ خیلی وقت ها شما را می گذارید نرم افزار از کار می افتد . شرکتی که سیستم اتوماسیون نوشته که با کتابخانه ورژن خاصی کار می کند ولی کتابخانه upgrade شود نرم افزار می گوید که کتابخانه نیست.

```
[root@lpi lib]# ldd
total 21364
drwxr-xr-x. 3 root root 4096 Sep 22 23:36 alsa
lrwxrwxrwx. 1 root root 14 Sep 26 17:05 cpp -> ../usr/bin/cpp
drwxr-xr-x. 3 root root 4096 Sep 22 23:37 crda
drwxr-xr-x. 2 root root 4096 Sep 26 17:35 dbus-1
drwxr-xr-x. 2 root root 4096 Sep 22 23:37 device-mapper
drwxr-xr-x. 42 root root 4096 Sep 22 23:46 firmware
drwxr-xr-x. 3 root root 4096 Sep 22 23:33 i686
drwxr-xr-x. 6 root root 4096 Sep 22 23:32 kbd
-rw xr-xr-x. 1 root root 142480 Nov 3 2011 ld-2.12.so
lrwxrwxrwx. 1 root root 10 Sep 22 23:33 ld-linux.so.2 -> ld-2.12.so
lrwxrwxrwx. 1 root root 13 Sep 22 23:42 ld-lsb.so.3 -> ld-linux.so.2
lrwxrwxrwx. 1 root root 15 Sep 22 23:33 libacl.so.1 -> libacl.so.1.1.0
```

همین طور که مشاهده می کنید خیلی از این کتابخانه ها در جلوی خود symbolic table دارند که اسمی دیگری را برای آنها در نظر گرفته ایم.

از کجا می توان فهمید که یک نرم افزار به چه کتابخانه هایی احتیاج دارد؟

```
[n.pardis@lpi ~]$ ldd /bin/date
linux-gate.so.1 => (0x009a0000)
librt.so.1 => /lib/librt.so.1 (0x003bf000)
libc.so.6 => /lib/libc.so.6 (0x00209000)
libpthread.so.0 => /lib/libpthread.so.0 (0x003a2000)
/lib/ld-linux.so.2 (0x001e3000)
```

این دستور header یک فایل exe را می خواند. نرم افزارهای لینوکس در به روزرسانی ها از این لحاظ دچار مشکل نمی شوند بلکه فقط نرم افزارهایی که خودمان نوشته ایم حتی در موارد خیلی نادر پس از اینکه لینک کرده ایم چاپ کرده که این چیزی نیست که می خواستم!

در سازمان های بزرگ ادمین موظف است که در user group آن توزیع قرار بگیرد و به مرور تغییرات را اعمال کند چون به روزرسانی کلی خیلی وقت گیر است. از طرف دیگر احتیاجی نیست که همیشه به روز باشیم ولی به روز رسانی نرم افزارهای امنیتی در اولویت است.

در آغاز به کار همراه اول سیستم آن روی fedora core 4 بود چون آن زمان ردهت لاینسنس داشت و centos هم هنوز نیامده بود ولی centos که آمد سیستم ها روی آن منتقل شد چون اکثر سازمان ها 2 ورژن قبلي مخصوصاتشان را پشتيبانی می کنند مثلا الان فدورا 14 داريم فقط برای 13 و 12 patch می آيد ولی برای 11 نمی آيد و هر 4 تا 6 ماه ورژن جدید تولید می شود. مشکل اینجاست که سرور مخابرات همیشه up است و شما نمی توانید دائمًا ورژن عوض کنید.

ubuntu نسخه LTS (Long Term Support) 6-7 ساله support می کنند که این مدت هم اگر مشتری داشته باشد قابل تمدید است پس اگر فدورا 1000 برابر سریع تر از یک نسخه LTS باشد ما فدورا نصب نمی کنیم.

چیست؟ bin

پکیجها به چند طریق نصب می شوند؛ در دیبان به صورت rpm. هستند فایل های tar نیز وجود دارند که حاوی سورس برنامه می باشند و باید کامپایل شوند.

خیلی از سازمان ها هم فایل bin می دهند این فایل خاصی است که طریقه نصب استاندارد ندارد.

```
[root@lp1 share] cd /lib/file RealPlayer11GOLD.bin
RealPlayer11GOLD.bin: ELF 32-bit LSB executable, Intel 80386, version 1
(SYSV), dynamically linked (uses shared libs), for GNU/Linux 2.2.5, not
stripped
[root@lp1 share]# file RealPlayer11GOLD.bin .
Extracting files for Helix installation.....
```

Welcome to the RealPlayer (11.0.0.4028) Setup for UNIX
Setup will help you get RealPlayer running on your computer.
Press [Enter] to continue...

بعضی وقت ها نصب نمی کند و خودش اجرا می شود که چنین فایل هایی را در ویندوز هم داریم که اصطلاحا self extract می کند به این صورت که نرم افزار را به عنوان یک data در درون خودش قرار داده و وقتی آن را اجرا می کنیم به دایرکتوری مناسب منتقلش می کند. یک مدل دیگر گه خیلی استفاده می شود (درایور های nvidia) که قسمت اول آن shell است و قسمت دوم exe می باشد . همچنین جاوا روی لینوکس به این صورت ارائه می شود:

```
[root@lp1 share]# less jre-6u24-linux-i586-rpm.bin
```

```
#!/bin/sh
PATH=/usr/bin:/bin
umask 022

release_comp() {
    if [ "$1" = "$2" ] ; then
        echo "eq"
    else
        lrel=`printf "%s\n%s\n" $1 $2 | \
            sort -t . -k 1,1n -k 2,2n -k 3,3n -k 4,4n -k 5,5n | \
            head -1`
        if [ "$1" = "${lrel}" ] ; then
            echo "lt"
        else
            echo "gt"
        fi
    fi
}

install_JavaDB() {
    return 0
}
jre-6u24-linux-i586-rpm.bin
```

این شل با دستور dd از جایی که باینری است تا انتهای را جدا کرده و در جایی که باید اجرا شود قرار می دهد. چنین فایل هایی را به این طریق می سازند:

```
[root@lpi share]# cat shell.exe > a.bin
```

در ترمینال بالا cat دو فایل shell و exe را به هم می چسباند و در فایل a.bin ذخیره می کند.

```
[n.pardis@lpi ~]$ man dd
...
seek=BLOCKS
    skip BLOCKS obs-sized blocks at start of output
```

با پارامتر seek در dd می توانید مشخص کنید که از کجا جدا کند.
خوبی چنین فایل هایی این است که بر روی هم توزیع ها اجرا می شود مثل درایورهای nvidia

```
[n.pardis@lpi share]$ less NVIDIA-Linux-x86-304.64.run
#!/bin/sh
skip=973
CRCsum=934967149
MD5=88a9db313d9eb368e64dda8416204e3e
label="NVIDIA Accelerated Graphics Driver for Linux-x86 304.64"
version_string=304.64
pkg_version=0
script=./nvidia-installer
targetdir=NVIDIA-Linux-x86-304.64
scriptargs=""
keep=n
add_this_kernel=n
apply_patch=n
TMPROOT=${TMPDIR:=/tmp}
TARGET_OS="Linux"
TARGET_ARCH="x86"

#
# NVIDIA Accelerated Graphics Driver for Linux-x86 304.64
# Generated by Makeself 1.6.0-nv
# Do not edit by hand.
```

کاربرد Selinux چیست؟

skip=973 یعنی اینکه 973 خط پایین تر کدهای exe قرار دارند. اول شل محیط را می بیند ، سیستم عامل را می بیند و exe را به تناسب تغییر داده و در جای مناسب قرار می دهد. حتی file هم گول می خورد چون صد بایت اول را می خواند و اعلام می کند که این یک فایل shell script است:

```
[n.pardis@lpi share]$ file NVIDIA-Linux-x86-304.64.run
NVIDIA-Linux-x86-304.64.run: POSIX shell script text executable
```

در واقع دستور file ، احساسی (heuristic) کار می کند ؛ 100 بایت اول را می خواند اگر main داشت می گوید کد C است ، اگر end و begin داشت می گوید کد پاسکال است.

لینوز توروالدز خیلی روی امنیت لینوکس کار نکرده بود. گروهی 15 سال روی امنیت لینوکس کار کردند . باید در Selinux عنوان کنیم که منظورمان از امنیت چیست و گرنه الان Selinux را فعال کنیم میل سرور بالا نمی آید. Selinux می گوید که به من نگفته که است.

در ابتدای امر باید ببینیم selinux فعال است یا نه:

```
[n.pardis@lpi share]$ cd /etc/selinux/
[n.pardis@lpi selinux]$ less config
```



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

چون diasable نزدیک فعال است ، enforcing یعنی اینکه سیاست ها را اعمال کن! که در این صورت خیلی از نرم افزارها بدکار می کنند.

permisive یعنی نظرت را اعلام کن که مثلاً فلان نرم افزار امن نیست و این تنظیمات کتابی 150 صفحه ای دارد که یعنی بی جهت و به سادگی نمی توان selinux را enable کرد.

centos در ردیف ، فدورا و integrated به صورت centos وجود دارد و بقیه توزیع ها هم به مرور در حال گنجاندن آن هستند.

```
[n.pardis@lpi selinux]$ man selinux
selinux(8)          SELinux Command Line documentation      selinux(8)

NAME
    selinux - NSA Security-Enhanced Linux (SELinux)

DESCRIPTION
    NSA Security-Enhanced Linux (SELinux) is an implementation of a flexible mandatory access control architecture in the Linux operating system. The SELinux architecture provides general support for the enforcement of many kinds of mandatory access control policies, including those based on the concepts of Type Enforcement, Role-Based Access Control, and Multi-Level Security. Background information and technical documentation about SELinux can be found at http://www.nsa.gov/selinux.

The /etc/selinux/config configuration file controls whether SELinux is enabled or disabled, and if enabled, whether SELinux operates in permissive mode or enforcing mode. The SELINUX variable may be set to any one of disabled, permissive, or enforcing to select one of these options. The disabled option completely disables the SELinux kernel and application code, leaving the system running without any SELinux protection. The permissive option enables the SELinux code, but causes it to operate in a mode where accesses that would be denied by policy
```

قبل از flask ، selinux وجود داشت.

LDAP

در این کلاس دو تا سرور داریم ولی در یک سازمان عریض و طویل در هریک از طبقات چند تا سرور ویندوز، لینوکس و سولاریس است یکی خواست ftp کند برای هر ماشین user&pass می خواهد و اگر تغییر کرد باید روی همه ماشین ها تغییر کندمی توان این را مرکزیت داد که یک دفعه که وارد کردیم برای دیدن همه اطلاعات مربوط به خودمان نیازی به وارد کردن مجدد user&pass نباشد.

LDAP مدل پیاده سازی شده X.500 است و از دانشگاه میشیگان حمایت می شود و یک استانداردادست که برای دسترسی به اطلاعات و به روزرسانی در یک دایرکتوری استفاده می شود.

Lightweight Directory Access Protocol = LDAP

نوع خاصی از پایگاه داده است و بیش تر برای خواندن استفاده می شود مثل دفترچه تلفن. تفاوت پایگاه داده و دایرکتوری این است که دایرکتوری برای جستجو بهینه شده است.

در سازمانی که روزی 500 تا login دارد یعنی مرتب باید چک کند user&pass درست است ولی در پایگاه داده اتوماسیون اداری مرتب اطلاعات داخل و خارج می شود.

دایرکتوری برای کارهایی که سریع تغییر می کنند مناسب نیست و از rollback در تراکنش ها پشتیبانی نمی کند. تراکنش چیست؟

مثلا من مبلغی را از حسابم به حساب دیگری منتقل می کنم که این یک تراکنش است ولی اگر در این بین مشکلی به وجود بیاید از حساب من رفته ولی به حساب طرف مقابل نرفته است که در این مورد rollback اتفاق می افتد یعنی مبلغ موردنظر به حساب من برمی گردد.

البته rollback هم زیاد کارایی ندارد به عنوان مثال فرض کنید سه بانک مستقل در اصفهان، تهران و مشهد داریم و من در اصفهان یک چک می دهم که پول از مشهد به تهران برود. کامپیوتر اصفهان با tcp/ip به مشهد و تهران اعلام می کند که آماده باشید که از حساب این شخص کم کنید اگر جواب هر دو مثبت بود اعلام می کند که با هم صحبت کنید و تا پول را از مشهد به تهران منتقل کنید درست در همین لحظه کامپیوتر مشهد down می شود ولی دیگر rollback جواب نمی دهد چون اصلا ارتباط قطع شده؛ پول را برداشته ولی نمی تواند آن را به تهران بفرستد.

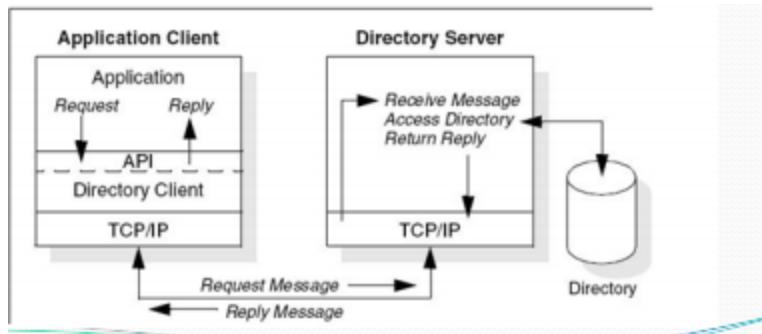
در عوض transaction partner آمده که به کامپیوترهای اطرافش گفته هر کس به تهران وصل شد اعلام کند که من کارم را انجام داده ام. ارتباط که قطع می شود اصفهان نمی داند که مشهد پول را در برنامه اش از حساب کم کرده یا نه!

تمام تلاش این است که داده سریع به کاربر برسد و گم نشود.

مدل اطلاعاتی LDAP براساس entry است که خود مجموعه ای از attribute می باشد که یک اسم منحصر به فرد جهانی دارد (DN Distinguished Name)

نوعی دسته بندی اطلاعات است به این صورت که کنترل می کند چه attribute در entry نیاز هستند (required) یا مجازند(allowed).

کاربرد LDAP در schema مثل کاربرد آن در پایگاه داده است و برای اعمال سیاست به کار می رود . در واقع entry و objectclass های حمایت شده و مجاز را تعریف می کند مثلاً کاربران باید user&pass داشته باشند یا مثلاً کاربران به پرینتر دسترسی داشته باشند.



براساس مدل client-server کار می کند. یک یا چند تا سرور LDAP حاوی داده ای هستند که directory information tree (DIT) را می سازد. client به سرورها متصل می شود و یک سوال از آن می پرسد. سرور با یک جواب به او پاسخ می دهد و یا با اشاره گری به جایی که client می تواند اطلاعات بیش تری دریافت کند.

یک سری اطلاعات در tableها پر می کنیم که مثلاً محدوده (bind) وصل شدن به سیستم چه باشد یا user&pass اشتباه وصل نشود (abandon) که خود client با سرورها اتصال برقرار می کند و ما اصلاً هیچ کاری نمی کنیم و همه کارها روی یک سیستم سنگین انجام می شود . در LPI2 فقط راه اندازی client آن ذکر می شود.

در ضمن باید در سازمان active directory هم راه اندازی کنید که از زیر مجموعهای ldap است و اینکه چطور این دو را به متصل کنید خودش یک پروژه سنگین است!

البته 3 تا موضوع که در LPI که با آنها اشتباه برخورد شده DNS ، DHCP و LDAP می باشد که حتماً باید کارگاهی تدریس شود.

LDAP Authentication

NSS و PAM با هم برای authenticate با استفاده از LDAP به کار می روند:

Pluggable Authentication Modules : PAM •

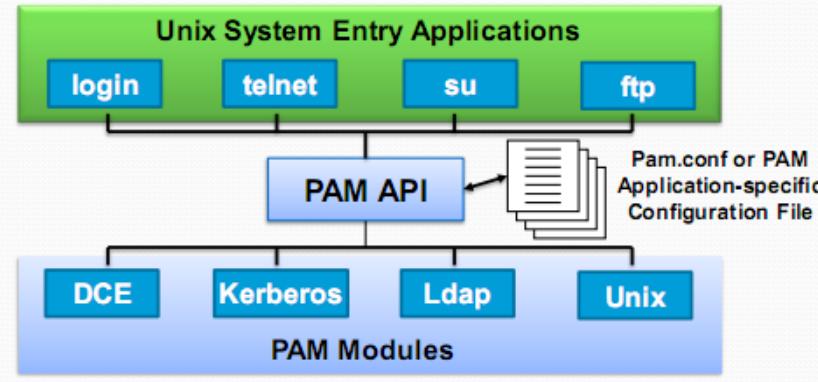
مسئول احراز هویت کاربران به جای برنامه صدازننده که از LDAP پشتیبانی می کند

Name Service Switch : NSS •

به وسیله برنامه ها برای به دست آوردن اطلاعات کاربری استفاده می شود:

/etc/{passwd,shadow,group,hosts,services,protocols,etc}

PAM Architecture

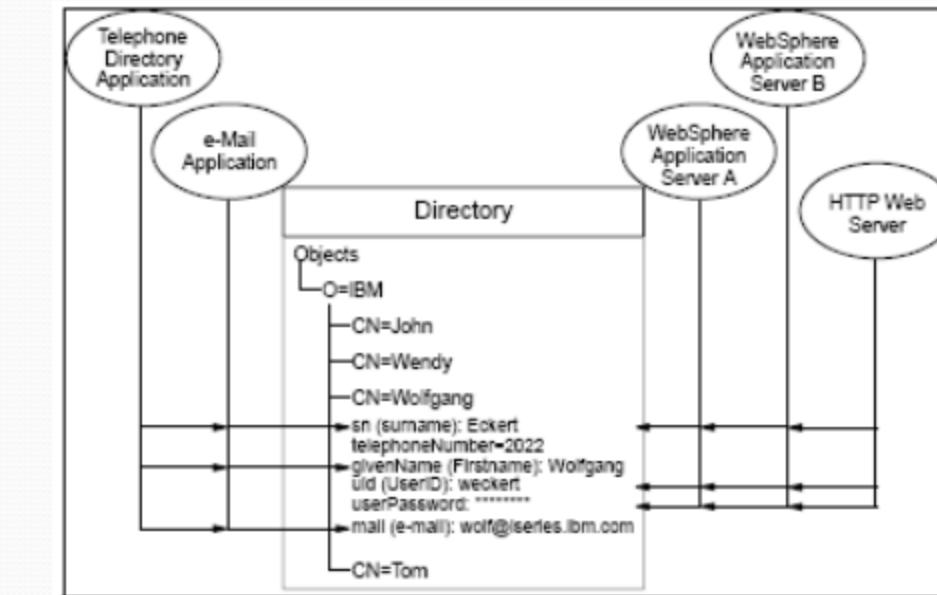


در شکل بالا به kerberos

معنای ازدهای دو سر است که یک نرم افزار فوق العاده پیچیده امنیتی است.

openLDAP یک نرم افزار تحت GPL است که در ویندوز و لینوکس پشتیبانی می شود. هم مثل خیلی از نرم افزارها از مرورگر وب برای تنظیمات استفاده می کندالبته از LDIF هم می توان استفاده کرد.

Directories advantages



در شکل قبل همه این application‌ها می‌توانند از LDAP برای Authentication و دسترسی استفاده کنند مثلاً دفترچه تلفن فقط به اسم و شماره تلفن شخص دسترسی داشته باشد. از طرف دیگر همه اطلاعات را یک بار و یک جا می‌نویسیم و به ندرت به روز می‌شود. مایکروسافت از LDAP استفاده کرد و Active Directory را نوشت.

در اینجا منظور ما از دایرکتوری، نوع ویندوزی و لینوکسی آن نیست بلکه منظور دفترچه راهنماست. LDAP مانند شخصی است که دفترچه راهنمایی در دست دارد؛ یک نفر زنگ می‌زند و پرینتر رنگی می‌خواهد. یکی login کرد، با چه شلی کار کند؟ دفترچه را نگاه می‌کند و مثلاً می‌گوید با cshell کار کند.

الان اطلاعات در لینوکس پراکنده است؛ در زیر etc چندین فایل داریم که مرتب مورد استفاده قرار می‌گیرد و حالا اگر چندین سیستم هم داشته باشیم و یک نفر بخواهد گروهش را عوض کند باید در همه این سیستم‌ها این کار را انجام دهد. در سازمان‌ها باید فرمی درست کرد و اسم، IP، ایمیل و شل مورد علاقه را از کارمندان بگیریم.

Network Information System

NIS امکاناتی را در سطح LAN می دهد تا تجمع اطلاعاتی مانند Account و اسمی کامپیوترها و شبکه ها و غیره را بر روی یک و یا چند کامپیوتر داشته باشیم. در این صورت لازم نیست هر کامپیوتری کلیه اطلاعات مربوط به کاربر و شبکه را دشته باشد و اگر نیاز به کنترل و به دست آوردن اطلاعاتی را داشت به خادم NIS مراجعه نموده و کسب تکلیف می نماید.

در شهرداری یک NIS server داشتیم و دو تا هم slave در کنارش برای جایگزین (در صورت از کار افتادن سرور اصلی) استفاده می کیم. هرجا را قرار دهیم ترافیک شبکه آنجا بالا می رود پس باید یک sub network جدا بگذاریم.

جدول NIS text base، NIS است و پایگاه داده ای نیست ولی LDAP از NIS خیلی قوی تر است.

همه کاربران زیر passwd ایمیل دارند:

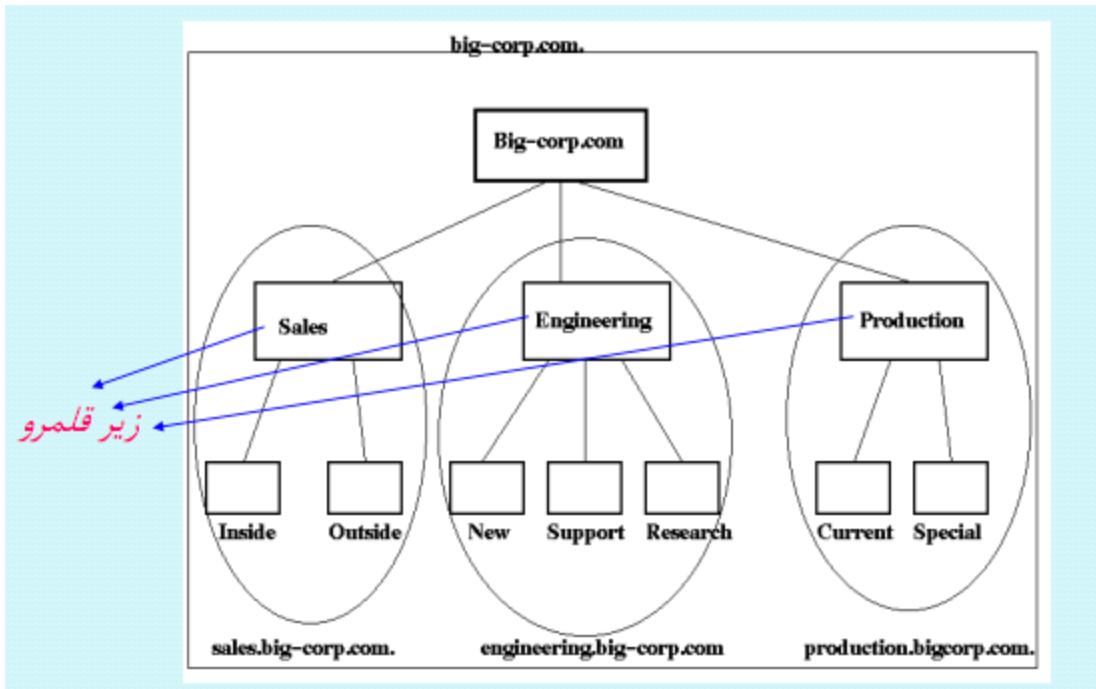
```
[n.pardis@lpi ~]$ less /etc/passwd

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
```

حال فرض کنید در کل ایران آدرس ایمیل بدنهاین فایل 70 میلیون خط می شود که تبعاً جستجو در آن کار مشکلی است و امكان درج تکراری (duplicate) در آن وجود دارد.

در واقع NIS همان کار LDAP را می کند. integrate، kerberos با kerberos احراز هویت می کند و بليطی می دهد که تا وقت خروج امكان دسترسی به بقیه امکانات متعلق به کاربر را می دهد.

سرور اصلی و slave هستند و در صورت تغییر دفترچه به روز می شوند.



برای تنظیم نرم افزارهایی مثل NIS باید چارت سازمانی را به دست آورید.

```
[n.pardis@lpi ~]$ cat /etc/nsswitch.conf
...
ethers:      files
netmasks:     files
networks:    files
protocols:   files
rpc:          files
services:    files

netgroup:    nisplus
publickey:   nisplus

automount:   files nisplus
aliases:     files nisplus
```

فایل بالا هم خیلی اساسی است که در آن مشخص می شود یک نرم افزار از LDAP استفاده کند یا از فایل ها (local) مثلا ethers: files یعنی اینکه کسی خواست به شبکه وصل شود فایل داریم : /etc/services هم در services /etc/host قرار دارد . البته در دیبلان و ردہت جنس فایل ها فرق می کند.

در شهرداری اسم کاربران خانم در alias بود و مثلا روز زن پیغام تبریک برای آنها ارسال می شد.(با مدیرها یا مردها) aliases

ایمیلی که قرار باشد ارسال باشد برای جستجوی اسم گیرنده اول زیر /etc/را می گردد و سپس به LDAP متصل می شود.

به عنوان مثال الان در مرورگر آدرس a.com را وارد می کنید کامپیوتر شما باید بداند IP متعلق به a.com چیست برای این منظور اول در سیستم شما می گردد و سپس به DNS (تبدیل اسم به IP) مراجعه می کند.

```
[n.pardis@lpi ~]$ cat /etc/hosts
127.0.0.1      localhost localhost.localdomain localhost4
localhost4.localdomain4
::1            localhost localhost.localdomain localhost6
localhost6.localdomain6
192.168.18.69
192.168.44.1
192.168.206.1
```

وزارت دارایی در کل ایران 80 تا سرور دارد حالا userID از مشهد رفت تبریز ، نباید مشکلی به وجود بیابد یعنی باید Authentication مجتمع باشد.

Pluggable Authentication Modules

در لینوکس این امکان وجود دارد که بتوان نرم افزارهای احراز هویت را به صورت داینامیک اجرا نموده و یا تغییر داد. نامه ای برای اداره می آید که پسورددها نمی توانند اسم گل باشد ولی `passwd` نمی تواند این کنترل را اعمال کند.

در لینوکس می توانیم به سیستم عامل بگوییم که اگر فلان نرم افزار را وارد حافظه کردی ماژولی را که نوشته ام را هم به آن بچسبان (Pluggable)

در سازمان ها برای الگوریتم خاص کنترل هویت باید ماژول بنویسید که این ماژول به `passwd` می چسبد و وقتی می خواهد پسورد عوض کنید کنترل را در دست می گیرد. نرم افزارهایی که می توانند از این ماژول ها استفاده کنند `pluggable` است.

یکی از دانشجویان دانشگاه الزهرا ماژولی نوشته است که اسمی `finnglish` را به عنوان پسورد قبول نمی کند که پایگاه داده آن را با اسمی که از ثبت احوال گرفته بود نوشت.

همچنین در لینوکس یک دیکشنری و بستر داریم که اگر لغت طولانی انگلیسی به عنوان پسورد بزنید ایراد می گیرد:

```
[n.pardis@lpi ~]$ passwd
Changing password for user n.pardis.
Changing password for n.pardis.
(current) UNIX password:
New password:
BAD PASSWORD: it is based on a dictionary word
```

خیلی از این امکانات استفاده می کند چون بخشی از کارهای احراز هویت است.

هر فایل تحت `/etc/pam.d` متناظر با یک نرم افزار `pam-aware` است به عنوان مثال اگر فایل `passwd` تحت این دایرکتوری باشد به این معنی می باشد که با اجرای فرمان `passwd` نرم افزار موردنظر شما به صورت داینامیک اجرا خواهد شد.

`pam-aware` یعنی نرم افزار `pam` آن را آگاه می کند و این باید یکی دیگه (ماژول) همراهش وارد حافظه می شود.

اولین بار `pam` روی سولاریس آمد ، از `sun6` آمده والان `sun11` داریم.

شما حتما باید نرم افزارتان را به صورت `.so` (shared object) بنویسید که `so` معادل `dll` در لینوکس است.

هم `pluggable` است و با استفاده از آن میتوان تعیین کرد که کسی که بخواهد سیستم را `halt` کند باید حتما پسورد دومی را وارد کند.

تمرین: اگر یک `userID` سه دفعه پسورد اشتباه وارد کرد نتواند وارد سیستم شود.

فایل های تحت pam.d در هر سطر سه فیلد دارند که به ترتیب libraryname و controlflaf ، moduletype است . که نام برنامه ای است که بایستی اجرا گردد فایل زیر این فیلد باید از نوع so باشد.

```
[n.pardis@lpi ~]$ lesscd /etc/pam.d/  
[n.pardis@lpi pam.d]$ ls  
atd                      newrole          ssh-keycat  
authconfig               other            su  
authconfig-gtk           passwd           subscription-manager  
authconfig-tui           password-auth   subscription-manager-gui  
chfn                     password-auth-ac sudo  
chsh                     polkit-1        sudo-i  
config-util              poweroff        su-1  
crond                   ppp              system-auth  
cups                     reboot           system-auth-ac  
cvs                      remote           system-config-authentication  
eject                    rhn_register  system-config-date  
fingerprint-auth        run_init        system-config-kdump  
fingerprint-auth-ac     runuser         system-config-keyboard  
gdm                      runuser-1      system-config-network  
gdm-autologin           samba           system-config-network-cmd  
gdm-fingerprint         setup            system-config-users  
gdm-password             smartcard-auth  vmtoolsd  
gnome-screensaver       smartcard-auth-ac vsftpd  
halt                     smtp             xserver  
ksu                      smtp.postfix  
login                   sshd
```

```
[n.pardis@lpi pam.d]$ cat passwd  
#%PAM-1.0  
auth    includesystem-auth  
account includesystem-auth  
password  substacksystem-auth  
-password optionalpam_gnome_keyring.so
```

web server

apache web server

گستردگی ترین و محبوب ترین سرور HTTP در دسترس روى اینترنت می باشد که زبان های php و perl را پشتیبانی می کند. یک برنامه رایگان و متن باز است که با سروزهای وب برای اداره کردن درخواست ها و تقاضاهای وب و منابع به کار می رود.

روی سیستم عامل خانواده یونیکس مانند لینوکس یا BSD اجرا می شود، همچنین می تواند روی ویندوز هم اجرا شود.

بزرگترین رقیب IIS ، Apache مایکروسافت است.

آپاچی در user ID 48 ، passwd در :

```
[root@lpi Packages]# less /etc/passwd
...
apache:x:48:48:Apache:/var/www:/sbin/nologin
```

اگر نرم افزاری را با root اجرا کنیم قدرت root را دارد و اگر کسی به آن حمله کند و نتواند مقاومت کند حمله کننده قدرت root را به دست می آورد. پس آپاچی به مجرد بالا آمدن userId خود را عوض می کندو از root خارج می شود.

```
[n.pardis@lpi Packages]$ ps -eaef |grep httpd
root    12153      1  0 22:40 ?          00:00:00 /usr/sbin/httpd
apache  12156 12153  0 22:41 ?          00:00:00 /usr/sbin/httpd
apache  12157 12153  0 22:41 ?          00:00:00 /usr/sbin/httpd
apache  12158 12153  0 22:41 ?          00:00:00 /usr/sbin/httpd
apache  12159 12153  0 22:41 ?          00:00:00 /usr/sbin/httpd
apache  12160 12153  0 22:41 ?          00:00:00 /usr/sbin/httpd
apache  12161 12153  0 22:41 ?          00:00:00 /usr/sbin/httpd
apache  12162 12153  0 22:41 ?          00:00:00 /usr/sbin/httpd
apache  12163 12153  0 22:41 ?          00:00:00 /usr/sbin/httpd
n.pardis 13428  7991  0 23:16 pts/0        00:00:00 grep httpd
```

سرور اصلی با uid=12153 با root بالا آمده ولی بقیه را با apache بر می گرداند و حداکثر این را هک می کنند که در این صورت هم کاری نمی توانند بکنند چون اصلاً فایل در passwd فایل nologin را می خوانند. آپاچی به پایگاه داده دسترسی ندارد و در سیستم های

جدی این دو تا از هم جدا هستند. حتی می توانیم آپاچی را با chroot زندانی کنیم! که هیچ کاری نتواند انجام دهد و فقط صفحات را نشان بدهد.

ترمینال زیر بخشی از فایل پیکربندی آپاچی را نشان می دهد:

```
[n.pardis@lpi ~]$ less /etc/httpd/conf/httpd.conf
...
#Listen 12.34.56.78:80
Listen 80

#
# Dynamic Shared Object (DSO) Support
```

یکی از پارامترهای این فایل bindAddress می باشد که بیش تر در دنیای IP می چرخد ولی listen بیشتر حول پورت است. مثلاً ماشین 5 تا IP دارد ولی می خواهید از یکی از آنها را برای وب سرور استفاده کنید.

AddCharset تعیین کننده کاراکترست های قابل پشتیبانی در آپاچی است که صفحات وب پردازش شده را تحت تاثیر قرار خواهد داد. باید عبارت زیر برای پشتیبانی از فارسی در این بخش وارد شود:

AddCharset UTF-8 .utf8

16 بیت جا می گیرد یعنی اداره ای که قبلاً 50 تا دیسک داشته باشد باید 100 تا شود.

آپاچی به جای اینکه نرم افزار تنومندی باشد تعداد زیادی SO دارد که در زمان نیاز به آن می چسبند.

با دستور زیر می توانیم ورژن آپاچی را پیدا کنیم :

```
[n.pardis@lpi ~]$ rpm -q httpd
httpd-2.2.15-15.el6.i686
```

این 2.2 است و لی هنوز 1.3 خیلی محبوب است.

فایرفاکس پلاگینی دارد که اگر به سایتی وصل شوید نشان می دهد که ورژن آپاچی سرور آن چند است که برای اکثر سایت های جدی 1.3 است. بعضی از سرورها نمی توانند برای نصب ورژن جدیدتر patch های امنیتی نصب می کنند

آپاچی را که صدا کنید به صورت پیش فرض 8 تا با خودش می آورد.

توجه داشته باشید که هیچ وقت اجازه ندهید پیغام خطای 404 ، 401 یا 402 در صفحه مرورگر کاربر ظاهر شود ؛ می توان یک صفحه HTML درست کرد که از کاربر معدتر خواهی کرده و اعلام کنیم که به زودی مشکل حل می شود.

معمولاً آپاچی را در سازمان‌ها به تنهايی راه اندازی نمی‌کنند و یک سری پایگاه داده و نرم افزار‌هی دیگری نیاز است. به طور کلی برای راه اندازی یک وب سرور به امکانات نرم افزاری زیر نیاز داریم:

- LINUX •
- APACHE •
- MySQL •
- php یا Perl •

لازم به ذکر است که نرم افزارهایی مانند lamp یا xamp سه مورد آخر را به صورت یک جا تامین می‌کنند و مثلاً احتیاجی نیست که در آپاچی ماژول‌های php را اضافه کنید. حتی با نصب xamp هنوز فایل‌های پیکربندی اجزای آن وجود دارند و می‌توان آنها را تغییر داد.

VLAN

فرض کنید تعدادی کاربر داریم که هر گروهی به switch‌های مختلف وصلند می‌خواهیم کاری کنیم که کارمندهای قسمت مالی هم دیگر را ببینند و بقیه بخشها نتوانند. مهم ترین مزیت گروه بندی است و اینکه می‌تواند ترافیک را محدود کند.

در واقع بدون کابل کشی اضافه و به صورت نرم افزاری کاربرها را جداسازی می‌کنیم.

دو نوع `vlan` داریم:

static •

که براساس پورت است.

dynamic •

براساس MAC address سیستم کاربراست و مزیتی که دارد این است که اولاً نیازی نیست همه ip‌ها را دستی بدھیم و ثانیاً اگر کاربران در شیکه جایه جا شوند مشکلی به وجود نمی‌آید.

VLAN را در لینوکس هم می‌توان تنظیم کرد ولی این کار بیشتر در در switch‌های است و در کارت شبکه هم می‌توان تنظیمات انجام داد.

دستور `vconfig add eth0.5` در لینوکس را می‌سازد و برای تنظیمات VLAN در لینوکس استفاده می‌شود.

با `ifconfig` هم می‌توان نصب کرد به این صورت که شما می‌توانید به کارت شبکه چند تا IP بدهید و اینها همه جور قابل استفاده هستند.

ابتدا با `ip`، `ifconfig` فعالی را می‌بینیم:

```
[root@lpi ~]# ifconfig eth0
eth0      Link encap:Ethernet HWaddr 00:0C:29:E7:29:E9
          inet addr:192.168.206.145 Bcast:192.168.206.255
          Mask:255.255.255.0
                  inet6 addr: fe80::20c:29ff:fee7:29e9/64 Scope:Link
                      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                      RX packets:50145 errors:0 dropped:0 overruns:0 frame:0
                      TX packets:14004 errors:0 dropped:0 overruns:0 carrier:0
                      collisions:0 txqueuelen:1000
                      RX bytes:6189108 (5.9 MiB) TX bytes:3168244 (3.0 MiB)
                      Interrupt:19 Base address:0x2024
```

حال یک ip دیگر هم تعریف می کنیم:

```
[root@lpi ~]# ifconfig eth0:1 1.2.3.4
[root@lpi ~]# ifconfig eth0:1 1.2.3.4
eth0:1      Link encap:Ethernet HWaddr 00:0C:29:E7:29:E9
            inet addr:1.2.3.4 Bcast:1.255.255.255 Mask:255.0.0.0
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  Interrupt:19 Base address:0x2024
```

ولی هیچ کدام از این ipها ، virtual نیست . برای این کار باید به جای ":" ، ":" قرار دهید . شکل زیر این کار را هم با استفاده از vconfig و هم با ifconfig نشان می دهد:

```
[root@myplanet ~]# vconfig add eth0 34
Added VLAN with VID == 34 to IF -:eth0:-
```

```
[root@myplanet ~]# ifconfig eth0.34
eth0.34    Link encap:Ethernet HWaddr 00:0C:29:FA:56:7D
            BROADCAST MULTICAST MTU:1500 Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

عمده فعالیت VLAN برای امنیت است که در یک سازمان هر کس به اطلاعاتی که مجاز است دسترسی داشته باشد. در VLAN tcpdump بزنید خیلی از فریم ها را نشان نمی دهد چون اصلا فریم ها با هم فرق می کند و عوض شده است. VLAN چیزهایی به یک فریم ساده اضافه می کند که هر کسی نمی تواند ببیند.

در واقع در ابتدای فریم کد 01111110 اضافه می کند:

0111111	Dest	Sourc	data	cac
---------	------	-------	------	-----

در واقع با این کد کارت شبکه می فهمد که فریمی رسیده است. شاید سوال پیش بیاید که اگر این کد ابتدایی عینا در قسمت data باید (مثالا در عکس یا فایل موسیقی) چه اتفاقی می افتد؟ جواب این است که در عمل شما هیچ وقت این کد را در داده نمی بینید چون اگر 6 تا 1 پشت سر هم رویت شود 5 تا 1 می گذارد و بعد 0 و در مقصد عکس این کار انجام می شود.

پایگاه داده

در بیشتر سازمان ها ادمین باید اطلاعاتی درباره پایگاه داده داشته باشد نه اینکه با آن application بنویسند بلکه مدیران همیشه می پرسند که برای نیازهای سازمان کدام پایگاه داده مناسب تر است؟

مايكروسفت خودش SQL SERVER نوشته و چون فقط خودش از ساختار ويندوز آگاهی داشته است performance آن روی ويندوز بالاست ولی اوراكل از system call استفاده کرده است ، IBM و اوراكل از مايكروسفت شکایت کرده اند که خودش از ويندوز آگاهی داشته و پایگاه داده ای هم که برای آن نوشته خيلي بهينه شده است.

یک سري پایگاه داده هم داريم که در دانشگاه برکلی نوشته شده است و در کارهای داخلی و دیکشنری ها استفاده می شود ولی خيلي معروف نیستند که مثلا در جایی مثل شركت نفت از آنها استفاده شود ولی در هر سازمانی که استخدام شوید با يكى از پایگاه داده های اوراكل ، MySQL یا postgres برخورد می کنيد. به عنوان مثال در ايرانسل هر سه تا استفاده می شود.

البته DB2 هم هست که البتهولی است . DB2 اول روی Main Frame بوده است و با اسمنبلر نوشته اند و خيلي سنگين است و آن را در لينوكس گنجاند و لينوكس کند شد!

خيلي از سازمانها اعتقاد دارند که لينوكس برای ميزبانی پایگاه داده مناسب نیت به عنوان مثال ايرانسل و همراه اول پایگاه های داده سنگين را روی sun نصب می کنند.

ORACLE

شرکت اوراكل تحت تمام سیستم های عامل بسته نرم افزاری ارائه می دهد و مدعی می باشد که تحت لينوكس بهترین سرويس را ارائه می نماید. انتقال اوراكل از روی ويندوز به لينوكس کار راحتی است.

اوراكل خواست ردهت را بخرد ولی نتوانست پس خودش Enterprise Linux را منتشر کرد که تفاوت چندانی با ردهت ندارد و سازمان های زیادی از آن استفاده می کنند. اوراكل خيلي به متن باز کمک کرده است ، Vmware هم خيلي کمک کرده و بخشی از روتین هايش را در هسته لينوكس جای داد که virtualization در لينوكس مدیون vmware است.

بانک تجارت می خواست به لينوكس مهاجرت کند برای اطمینان آمار به دست آورديم که از 100 بانک بزرگ دنيا 99تا ار لينوكس استفاده می کنند و بانک ها جرئت نمی کنند که روی بقیه سیستم ها کار کنند. از اين نظر متن بازها خيلي خوب است ولی ممکن است مثل MySQL لايセンس خود را عوض کند یا توسعه دهنده گانش کنار بکشند.

امكان اين بحران برای centos وجود دارد که توسط دانشگاه کارنگی ملون حمایت می شود و مدتی رئيس تیم کناره گيري کرده بود و patch نمي آمد ولی اين نگرانی ها در آپاچی وجود ندارد چون يك بنیاد است.

کامپيوترهای جدید و بزرگ اصطلاحا MultiPath هستند يعني مثلا به tape و ديسک راه های مختلفی هست ، برای استفاده از اين امكانات نرم افزار هوشمندی نياز است که اوراكل اين ويژگي را دارد.

یکی دیگر از قابلیت های اوراکل امکان اجرای موازی روی یک CPU است مثلاً اگر با دستور select جست و جویی انجام دهیم و پایگاه داده خیلی بزرگ باشد اوراکل این دستور را 1000 تا می کند و به هر کدام می گوید که یک قسمت را پیدا کند.

اوراکل سعی می کند تا جایی که می تواند در raw mode کار کند.

اوراکل از درایور Network Bonding پشتیبانی می کند که با استفاده از آن به راحتی می توانید در لینوکس پورت تعريف کنید به عنوان مثال اگر دو تا کارت شبکه داشته باشید و یکی از آنها قطع شود آدم هایی که به آن وصلند هم قطع می شوند ولی اگر bond تعريف کرده باشیم کارت شبکه دوم می تواند کار اولی را ادامه دهد و اوراکل می گوید که سعی کنم از همه کارت شبکه ها استفاده کنم. در واقع در این تکنیک سیستم اصلاً متوجه از کار افتادن کارت شبکه نمی شود و می توان با دستور ifconfig آن را تنظیم کرد.

اوراکل برای پایگاه داده های بزرگ بسیار مناسب است چون parallel است و روی دیسک های مختلف توزیع می شود. اوراکل خودش اعلام کرده که محدودیت ندارد و محدودیتش محدودیت سیستم عامل است. همچنین این شرکت برای اکثر کارها و سازمان ها حتی سدسازی پکیج دارد.

بزرگترین مشکل اوراکل این است که اگر crash کند restart و recovery نمی فهمد!

MySQL

این نرم افزار روی لینوکس به خوبی کار می کند.

MySQL خیلی برای Query مناسب است و به عنوان مثال در سازمان سنجش و برای کنکور سراسری استفاده می شود. اوراکل می گوید که برای Hybrid مناسب هستم (ترکیبی از تغییر و جست و جو)

مقایسه سریع پایگاه داده ها بدون در نظر گرفتن شرایط کار درستی نیست؛ پیکان یا BMW مهم نیست مهم این است که اگر خراب شد کسی هست آن را تعمیر کند، یک مدیر بیشتر نگران پشتیبانی است تا بهتر بودن.

البته اوراکل MySQL را خرید و این کار را IBM Informix با DB2 کرد، که خودش DB2 داشت و رقیب جدی اش infirmix بود؛ خصوصیات خوب informix را برداشت و روی DB2 قرار داد و پس از مدتی به کاربران informix ایمیل داد که نرم افزارهای مهاجرت را برای شما ارسال می کنم.

سیستم mySQL Accounting یا هو با MySQL است ولی از آن در بانک ها نمی توان استفاده کرد؛ MySQL در ورژن قبلی تا 4TB پشتیبانی می کرد، در همراه اول وقتی از این مقدار بیش تر شد بیت علامت یک شد و هد به اول پایگاه برگشت و پایگاه خراب شد. پشتیبانی هم فقط از طریق فروم هاست در حالی که اوراکل خودش آنلاین پشتیبانی میکند.

postgre

هواشناسی ایران مثل خیلی از کشورها روی postgre است که دقیقاً مثل اوراکل است ولی متن باز است. Hybrid است و بیشتر توسعه آن دانشگاهی است.

Linux Performance And Tuning

اگر شما وب سایتی راه اندازی کنید که به مرور زمان تعداد کاربران آن زیادشود نمی توانید کامپیوتر را عوض کنید بلکه باید tuning انجام دهید.

firmware : چیپ هایی که قبلا می ساختند نرم افزار در داخلش embedded بود دلیل اینکه الان مودم ها تحت لینوکس خوب کار نمی کنند این است که این نرم افزار را برای ویندوز نوشته اند و درایور تحت ویندوز که بالا می آید این firmware را در سخت افزار load می کند.

در cpu ، cpu ، firmware نصب می شود ؛ یکی از سرویس های cpu ، load کردن اینهاست.

thread یک واحد پردازشی ساخته شده در درون یک پردازش در بعضی از منابع مشترک است و با دیگر نخ ها در یک پردازش به صورت موازی اجرا می شوند.

linux memory architecture

برای اجرای یک پردازش هسته لینوکس بخشی از حافظه را برای پردازش متقارضی اختصاص می دهد و به دلیل تعداد زیاد پردازش ها کرنل باید حافظه را به صورت بهینه استفاده کند.

در معماری 32 بیتی بیشترین آدرس حافظه ای که یک پردازش می تواند به آن دسترسی داشته باشد 4GB است. این محدودیت از آدرس دهی مجازی 32 بیتی ناشی شده است. در یک پیاده سازی استاندارد فشاری آدرس مجازی به 3 گیگابایت فشاری کاربری و یک گیگابایت فضای کرنل تقسیم می شود ولی در معماری 64 بیتی چنین محدودیتی وجود ندارد.

اینتل این مشکل را به این صورت حل کرد که روی cpu ها تغییراتی با عنوان physical address extension pae یا اعمال کرد پس یا کرنلی که از pae پشتیبانی می کند استفاده کنید یا خودتان که کرنل را کامپایل می کنید pae را فعال کنید.

درک معیارهای عملکرد لینوکس

با دستور nice می توانیم اولویت یک پردازش را تغییر دهیم که البته کار اولویت بندی مجدد خود زمانی را از cpu تلف می کند. سعی کنید به ندرت از nice استفاده کنید چون بعضی مواقع مجبور می شوید سیستم را down کنید.

در سمینارها می گویند که لینوکس واقعا context switch دارد ، بعضی مواقع یک پروسس در loop می افتد و در ویندوز ctrl+alt+del هم کار نمی کند و سیستم هنگ می کند ولی در لینوکس تحت هر شرایطی می توان cpu را در اختیار گرفت.

یکی از معیارهای رابط شبکه ، تعداد بسته های دریافتی و ارسالی است:

```
[root@lpi ~]# ifconfig eth0
eth0      Link encap:Ethernet HWaddr 00:0C:29:E7:29:E9
          inet addr:192.168.206.145 Bcast:192.168.206.255
          Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fee7:29e9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:57289 errors:0 dropped:0 overruns:0 frame:0
          TX packets:15222 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7541532 (7.1 MiB) TX bytes:3356248 (3.2 MiB)
          Interrupt:19 Base address:0x2024
```

overrun یعنی اینقدر بسته برای درایور ارسال شود که تعدادی از آنها از دست درایور در می رودا! و اصلا چیز خوبی نیست چون نشان می دهد کارت شبکه و سیستم شما تحمل بارهای سنگین را ندارد.

Monitoring Tools

top

vmstat

```
[n.pardis@lpi root]$ vmstat -s
 1030888 total memory
 926852 used memory
 362208 active memory
 431324 inactive memory
 104036 free memory
 91724 buffer memory
 589556 swap cache
 2064376 total swap
     0 used swap
 2064376 free swap
 48722 non-nice user cpu ticks
   538 nice user cpu ticks
 89102 system cpu ticks
16519524 idle cpu ticks
 4570049 IO-wait cpu ticks
   162 IRQ cpu ticks
 1222 softirq cpu ticks
     0 stolen cpu ticks
 533199 pages paged in
 457290 pages paged out
     0 pages swapped in
     0 pages swapped out
 4154277 interrupts
 6431412 CPU context switches
1352574528 boot time
   62528 forks
```

uptime

```
[n.pardis@lpi root]$ uptime
00:03:41 up 3 days, 1:24, 4 users, load average: 0.00, 0.00, 0.00
```

درصد کارکرد cpu را (به ترتیب از چپ به راست) در 5 دقیقه، 10 دقیقه و 15 دقیقه اخیر نشان می دهد.

ps

[n.pardis@lpi root]\$ ps -elfL less															
F	S	UID	PID	PPID	LWP	C	NLWP	PRI	NI	ADDR	SZ	WCHAN	RSS	PSR	STIME
TT	Y				TIME	CMD									
4	S	root	1	0	1	0	1	80	0	-	716	?	1428	0	Nov10 ?
00:00:03		/sbin/init													
1	S	root	2	0	2	0	1	80	0	-	0	?	0	0	Nov10 ?
00:00:00		[kthreadd]													
1	S	root	3	2	3	0	1	-40	--		0	?	0	0	Nov10 ?
00:00:00		[migration/0]													
1	S	root	4	2	4	0	1	80	0	-	0	?	0	0	Nov10 ?
00:00:00		[ksoftirqd/0]													
1	S	root	5	2	5	0	1	-40	--		0	?	0	0	Nov10 ?
00:00:00		[migration/0]													
5	S	root	6	2	6	0	1	-40	--		0	?	0	0	Nov10 ?
00:00:00		[watchdog/0]													
1	S	root	7	2	7	0	1	80	0	-	0	?	0	0	Nov10 ?
00:00:00		[events/0]													
1	S	root	8	2	8	0	1	80	0	-	0	?	0	0	Nov10 ?
00:00:00		[cpuset]													
1	S	root	9	2	9	0	1	80	0	-	0	?	0	0	Nov10 ?
00:00:00		[khelper]													
1	S	root	10	2	10	0	1	80	0	-	0	?	0	0	Nov10 ?
00:00:00		[netns]													

وقتی نرم افزار در حافظه است اگر fork کند نرم افزار هم وزن خودش را می آورد ولی thread یا low weight process مشترکات را وارد حافظه نمی کند.

فرق بافر و کش چیست؟

شما جاهای دایرکتوری های مختلف می روید این جاهای کش می شود ولی بافر اطلاعات شمامست ، چیزی را که از فلاپی می خوانید روی بافر ذخیره می شود ولی کش کارهای خود سیستم است که مثلا فلان دایرکتوری کجای دیسک است.

iostat

```
[n.pardis@lpi root]$ iostat
Linux 2.6.32-220.el6.i686 (lpi) 11/14/2012 _i686_(1 CPU)
```

avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle
	0.23	0.00	0.43	21.12	0.00	78.22

Device:	tps	Blk_read/s	Blk_wrtn/s	Blk_read	Blk_wrtn
sda	0.26	4.72	4.28	1022418	927084
scd0	0.00	0.16	0.00	35660	0
dm-0	0.71	4.67	4.28	1010786	927048
dm-1	0.00	0.02	0.00	3936	0

sar

دستور زیر هر یک ثانیه و به تعداد سه بار آمار سیستم را چاپ می کند:

```
[n.pardis@lpi root]$ sar 1 3
Linux 2.6.32-220.el6.i686 (lpi) 11/14/2012 _i686_(1 CPU)
```

01:11:45 AM	CPU	%user	%nice	%system	%iowait	%steal	%idle
01:11:46 AM	all	0.00	0.00	2.08	0.00	0.00	97.92
01:11:47 AM	all	0.00	0.00	1.00	0.00	0.00	99.00
01:11:48 AM	all	1.98	0.00	10.89	0.00	0.00	87.13
Average:	all	0.67	0.00	4.71	0.00	0.00	94.61

کلا sar امکانات زیادی دارد و کار خیلی از دستورات دیگر را انجام می دهد.

mpstat

این دستور برای گزارش فعالیت های هر یک از cpu های در دسترس روی یک سرور چندپردازنده ای استفاده می شود. همچنین میانگین کلی (Global average) میان همه cpu ها گزارش می شود.

```
[n.pardis@lpi root]$ mpstat
Linux 2.6.32-220.el6.i686 (lpi) 11/14/2012 _i686_(1 CPU)
```

01:16:41 AM	CPU	%usr	%nice	%sys	%iowait	%irq	%soft	%steal	%guest
	%idle								
01:16:41 AM	all	0.23	0.00	0.42	21.07	0.00	0.01	0.00	0.00
		78.27							

در شهرداری در مانیتورهای بزرگ همه این دستورات را همزمان مانیتور می کنیم و مقدار نرمال پارامترها را داریم که اگر تغییرات مشکوکی به وجود آمد متوجه می شویم.

netstat

این دستور تعداد زیادی از اطلاعات مربوط به شبکه از قبیل network ، protocol ، interface ، routing ، socket usage و ... را نمایش می دهد:

```
[n.pardis@lpi ~]$ netstat -s|less
```

Ip:

```
41615 total packets received
20 with invalid addresses
0 forwarded
0 incoming packets discarded
41595 incoming packets delivered
28164 requests sent out
```

Icmp:

```
30 ICMP messages received
1 input ICMP message failed.
ICMP input histogram:
    destination unreachable: 4
    echo requests: 26
32 ICMP messages sent
0 ICMP messages failed
ICMP output histogram:
    destination unreachable: 6
    echo replies: 26
```

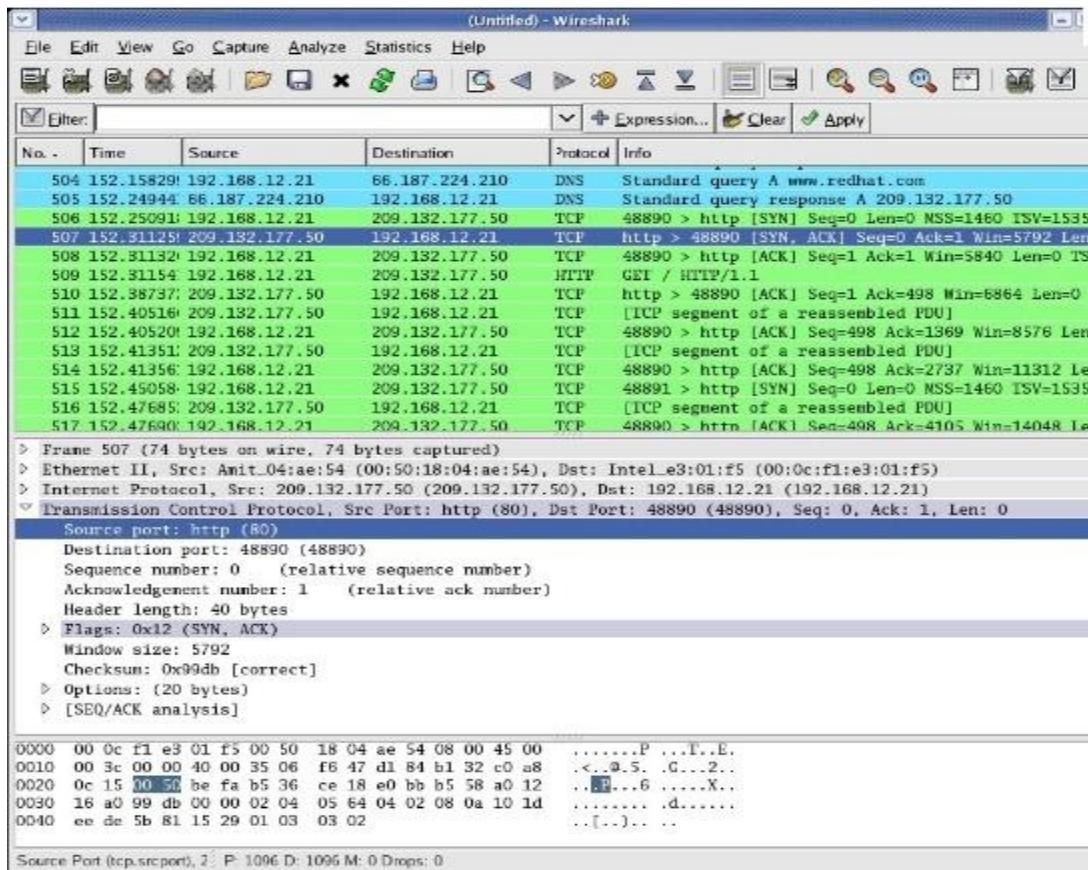
IcmpMsg:

```
InType3: 4
InType8: 26
OutType0: 26
OutType3: 6
```

تحلیل خروجی این دستور نیاز به دانش بالایی دارد. منظور از Icmp همان ping می باشد و forwarded نیز بسته های دور ریخته شده است.

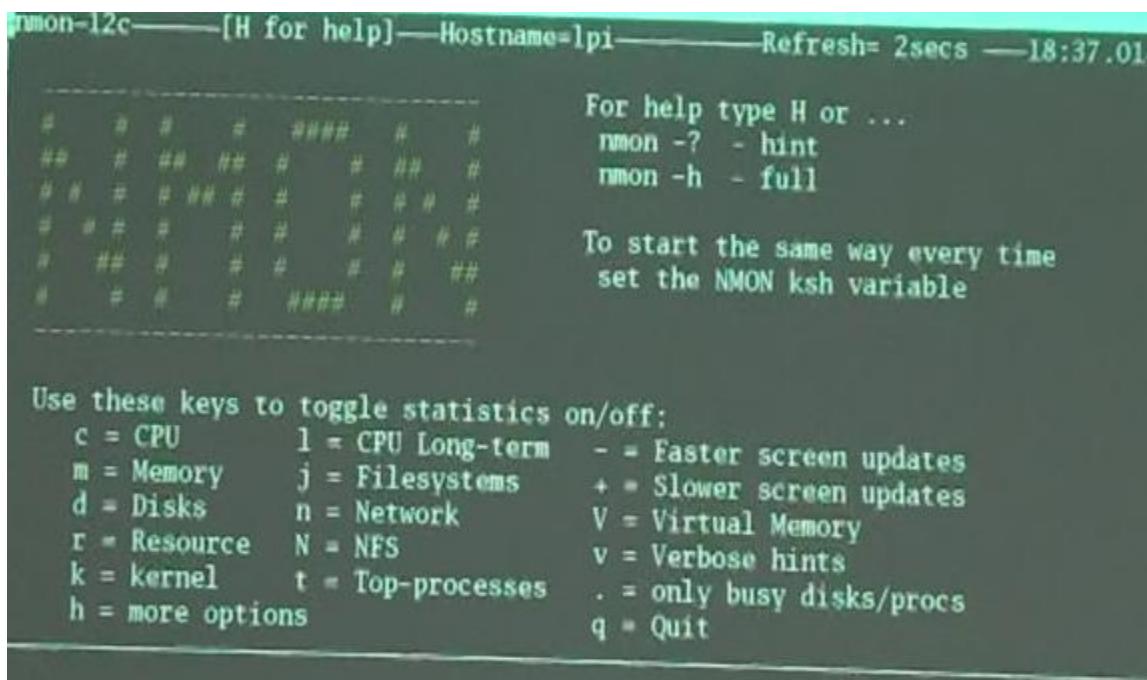
tcpdump/ethereal

این دو دستور برای گرفتن و آنالیز ترافیک شبکه مورد استفاده قرار می گیرند. tcpdump را فقط root می تواند اجرا کند. ethereal در ویندوز هم هست فقط اسمش عوض شده است. wireshark هم که بر روی همه سیستم عامل ها قابل نصب است نرم افزاری با رابط گرافیکی است که پکیج ها را آنالیز می کند و فریم ها را نشان می دهد.



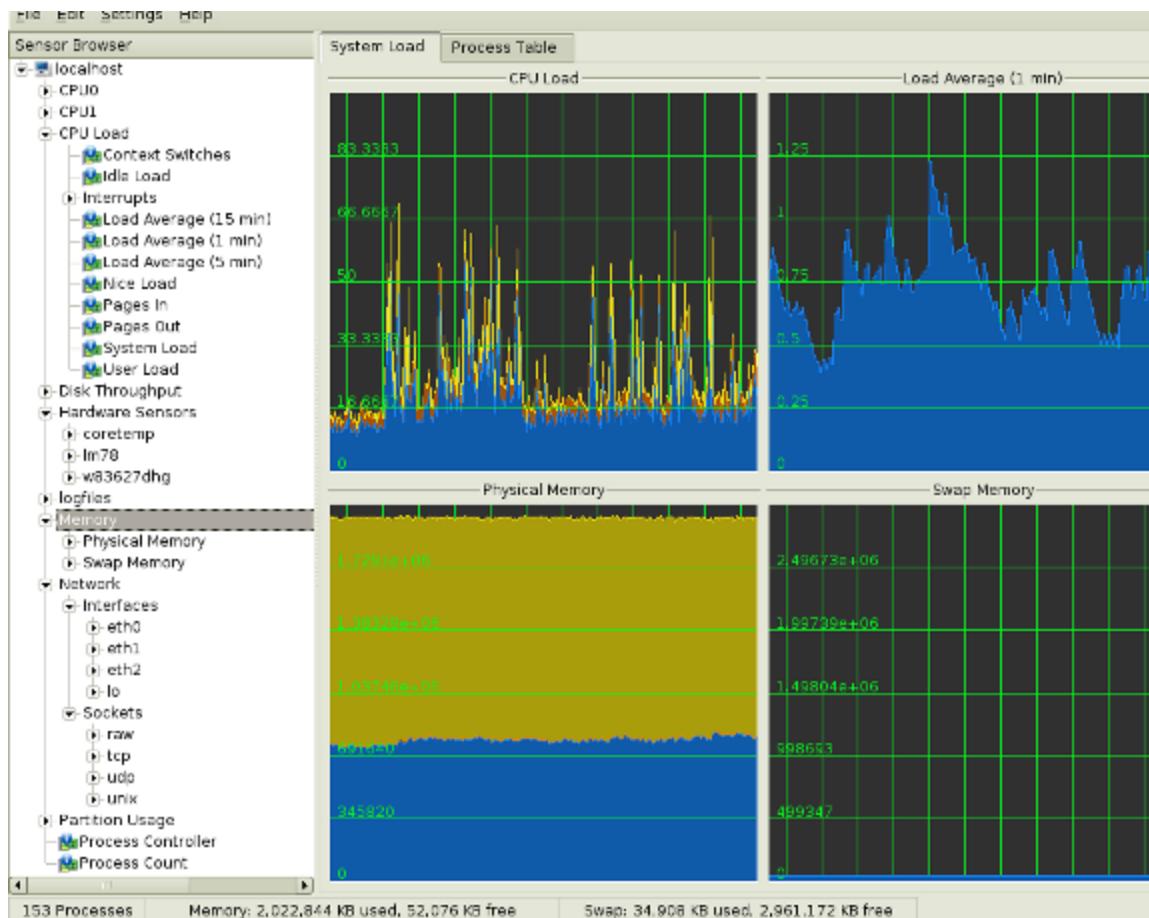
nmon

یکی از نرم افزارهای محبوب مانیتورینگ



KDE system guard

یک KDE task manager برای که گرافیکی بوده و همچنین قوی ترین سیستم مانیتورینگ می باشد.



tuning the operating system

پارامترهایی که بیشترین بهبود را در عملکرد سیستم دارند عبارتند از:

- linux memory management
- system clean up
- disk subsystem tuning
- kernel tuning using sysctl
- network optimization

اولین قدم در tuning این است که باید تنظیمات سیستم فعلی را بررسی کنیم.

مدیریت تغییرات شاید مهم ترین فاکتور برای tuning موفقیت آمیز باشد:

- یک بروزرسان مدیریت تغییرات مناسب پیاده سازی کنید
- هیچ وقت بیشتر از یک پارامتر را در حین پروسه tuning تغییر ندهید
- پارامترهایی را که حدس می زنید عملکرد را بهبود می بخشنند دوباره امتحان کنید
- پارامترهای موفقیت آمیز را مستندسازی کنید

در دستورهایی مثل sar هیچ وقت تعداد تکرار (interval) را یک ثانیه نگذارید چون خود این کار عملکرد سیستم را پایین می آورد

برای هر تلاشی در یک solid base tuning انجام دهید یعنی تضمین کنید که همه زیرسیستم ها به همان صورتی کار می کنند که برای آن طراحی شده اند و هیچ تناقضی وجود ندارد.

نمونه ای از تناقض کارت شبکه گیگابیتی است که سیستم نشان می دهد با 100MBIT/s کار می کند.

dmesg

هدف اصلی از dmesg نمایش پیام های کرنل می باشد و خروجی آن خیلی فنی می باشد:

```
[n.pardis@lpi Packages]$ dmesg
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 2.6.32-220.el6.i686 (mockbuild@x86-
003.build.bos.redhat.com) (gcc version 4.4.5 20110214 (Red Hat
4.4.5-6) (GCC) ) #1 SMP Wed Nov 9 08:02:18 EST 2011
KERNEL supported cpus:
Intel GenuineIntel
AMD AuthenticAMD
NSC Geode by NSC
Cyrix CyrixInstead
Centaur CentaurHauls
Transmeta GenuinetMx86
Transmeta TransmetaCPU
UMC UMC UMC UMC
Disabled fast string operations
BIOS-provided physical RAM map:
BIOS-e820: 0000000000000000 - 000000000009F800 (usable)
```

ulimit

کنترل روی منابع در دسترس shell و پردازشگری که به وسیله شل شروع شده اند را فراهم می کند. سیستم هایی که برای بیش ترین سطح عملکردی طراحی شده اند باید هر اتفاق منابعی را کمینه کنند.

مثلا برنامه ای می نویسید که برای هر کاربر حداقل 3 فایل باز می شود و 1000 نفر وصل می شوند پس 3000 فایل باید باز شود ولی قید کرده ایم که 1024 فایل بیشتر نتوان باز کرد:

```
[n.pardis@lpi Packages]$ ulimit -a
core file size          (blocks, -c) 0
data seg size            (kbytes, -d) unlimited
scheduling priority
file size                (blocks, -f) unlimited
pending signals           (-i) 7937
max locked memory        (kbytes, -l) 64
max memory size          (kbytes, -m) unlimited
open files               (-n) 1024
pipe size                 (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority        (-r) 0
stack size                (kbytes, -s) 10240
cpu time                  (seconds, -t) unlimited
max user processes        (-u) 1024
virtual memory             (kbytes, -v) unlimited
file locks                (-x) unlimited
```

اگر خطای too many files گرفتید یا هر خطای دیگری که از محدودیت های فایل بالا ناشی می شود ، برای رفع آن باید جمیع جهات را در نظر گرفته و پارامترهای این دستور را تغییر دهید.

با زدن دستور زیر متوجه می شوید که حتی برای تعداد کمی کاربر هزاران فایل در سیستم باز شده است:

```
[n.pardis@lpi Packages]$ lsof|less
COMMAND      PID      USER      FD      TYPE DEVICE SIZE/OFF      NODE NAME
init         1      root      cwd    unknown
Permission denied)
init         1      root     rtd    unknown
Permission denied)
init         1      root     txt    unknown
Permission denied)
init         1      root    NOFD
Permission denied)
kthreadd     2      root      cwd    unknown
Permission denied)
kthreadd     2      root     rtd    unknown
Permission denied)
kthreadd     2      root     txt    unknown
Permission denied)
...
...
```

Daemons

بعد از نصب پیش فرض یک سیستم لینوکس ممکن است چندین سرویس و daemon غیرضروری روی سیستم enable شود. غیرفعال کردن اینها حافظه مصرفی سیستم، زمان بالا آمد سیستم، تعداد پردازش ها و Context switch ها و از همه مهم تر وقوع تهدیدات امنیتی را کاهش می دهد.

Daemons	Description
apmd	Advanced power management daemon. apmd will usually not be used on a server.
arpTables_jf	User space program for the arpTables network filter. Unless you plan to use arpTables, you can safely disable this daemon.
autofs	Automatically mounts file systems on demand (for example, mounts a CD-ROM automatically). On server systems, file systems rarely have to be mounted automatically.
cpuspeed	Daemon to dynamically adjust the frequency of the CPU. In a server environment, this daemon is recommended off.
cups	Common UNIX Printing System. If you plan to provide print services with your server, do not disable this service.
gpm	Mouse server for the text console. Do not disable if you want mouse support for the local text console.
hpoj	HP OfficeJet support. Do not disable if you plan to use an HP OfficeJet printer with your server.
irqbalance	Balances interrupts between multiple processors. You may safely disable this daemon on a single CPU system or if you plan to balance IRQ statically.
isdn	ISDN modem support. Do not disable if you plan to use an ISDN modem with your server.
kudzu	Detects and configures new hardware. Should be run manually in case of a hardware change.

Daemons	Description
netfs	Used in support of exporting NFS shares. Do not disable if you plan to provide NFS shares with your server.
nfslock	Used for file locking with NFS. Do not disable if you plan to provide NFS shares with your server.
pcmcia	PCMCIA support on a server. Server systems rarely rely on a PCMCIA adapter so disabling this daemon is safe in most instances.
portmap	Dynamic port assignment for RPC services (such as NIS and NFS). If the system does not provide RPC-based services there is no need for this daemon.
rawdevices	Provides support for raw device bindings. If you do not intend to use raw devices you may safely turn it off.
rpc*	Various remote procedure call daemons mainly used for NFS and Samba. If the system does not provide RPC-based services, there is no need for this daemon.
sendmail	Mail Transport Agent. Do not disable this daemon if you plan to provide mail services with the respective system.
smartd	Self Monitor and Reporting Technology daemon that watches S.M.A.R.T. compatible devices for errors. Unless you use an IDE/ SATA technology based disk subsystem, there is no need for S.M.A.R.T. Monitoring.
xfs	Font server for X Windows. If you will run in runlevel 5, do not disable this daemon.

تغییر `runlevel` ها

معمولاً نیازی به رابط گرافیکی روی یک سرور لینوکس نیست . همه وظایف راهبری لینوکس می تواند از طریق خط فرمان یا مرورگر وب انجام شود. اگر شما رابط گرافیکی را ترجیح می دهید چندین نرم افزار مبتنی بر وب از قبیل `linuxconf` ، `webmin` و `swat` وجود دارند. `linuxconf` هر چند ماه یک بار به روز می شود ولی `webmin` حمایت نمی شود.

حتی اگر رابط گرافیکی روی به صورت محلی روزی سرور غیر فعال باشد هنوز می توان به صورت از راه دور (remotely) متصل شده و از GUI استفاده کرد برای این منظور از با دستور `ssh` پارامتر `-X` استفاده کنید.

تغییر پارامترهای کرنل

فایل سیستم `proc` رابطی به کرنل در حال اجرا فراهم می کند که می تواند برای هدف های مانیتورینگ و تغییرات در کرنل استفاده شود. به عنوان مثال برای تغییر پارامتر `shmmax` می توان به صورت زیر عمل کرد (`cat` برای نمایش و `echo` به منظور تغییر)

```
[root@lpi ~]# cat /proc/sys/kernel/shmmax
4294967295
[root@lpi ~]# echo 4094967295 >/proc/sys/kernel/shmmax
[root@lpi ~]# cat /proc/sys/kernel/shmmax
4094967295
```

دستور `sysctl` هم برای اعمال تغییرات در کرنل پارامترهای `/proc/sys` را تغییر می دهد.

یک برنامه هر چقدر کم تر باشد اولویت آن بالاتر است و مثلا 5 nice ، 5 تا از این کم می کند و برای زیاد کردن این مقدار باید - بگذاریم. از این دستور با احتیاط استفاده کنید.

```
[root@lpi ~]# man nice
Formatting page, please wait...

NICE(1)                               User Commands
NICE(1)

NAME
    nice - run a program with modified scheduling priority

SYNOPSIS
    nice [OPTION] [COMMAND [ARG]...]

DESCRIPTION
    Run COMMAND with an adjusted niceness, which affects process
scheduler-
ing. With no COMMAND, print the current niceness. Nicenesses
range
from -20 (most favorable scheduling) to 19 (least favorable).

-n, --adjustment=N
    add integer N to the niceness (default 10)

--help display this help and exit

--version
    output version information and exit
```

tuning network subsystem

زیرسیستم شبکه باید اولین باری که سیستم عامل نصب می شود tune شود . یک مشکل در این بخش می تواند روی دیگر زیرسیستمها تاثیر بگذارد.

به عنوان مثال مصرف cpu می تواند تحت تاثیر قرار گیرد مخصوصاً اگر اندازه packet ها خیلی کوچک باشد و اگر تعداد زیادی ارتباط tcp وجود داشته باشد حافظه افزایش پیدا میکند.

هر بسته ای که روی سیستم قرار می گیرد یک وقهه ایجاد می کند و مهم است که برنامه نویسان برنامه ها را به چه روشی طراحی کنند.

ما چقدر است؟ maximum transfer unit MTU

Iface	MTU	Met	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
eth0	1500	0	69150	0	0	0	29273	0	0	0	BMRU
eth0:1	1500	0	- no statistics available -								BMRU
lo	16436	0	12677	0	0	0	12677	0	0	0	LRU

اگر آن را 15000 کنیم وضع بهتر می شود؟

نه، بلکه کنترل می شود ؛ لینوکس بسته 15000 تایی می فرستد ولی 1500 router 1500 تایی منتقل می شود پس بسته ها 10 قسمت شوند و دوباره به صورت ترتیبی کد شده و فرستاده شوند.

شرکتی اینترنت گیگابیت گرفته بود ولی تغییری در سرعت ایجاد نشد معلوم شد که به اینترنت 128kbit متصل هستند و مودم ADSL نمی توانند بسته ها را ارسال کند و بسته ها را drop می کند در نهایت باید مودم ADSL را 1024 کند.

```
[root@lpi ~]# ifconfig eth0 mtu 9000 up
```

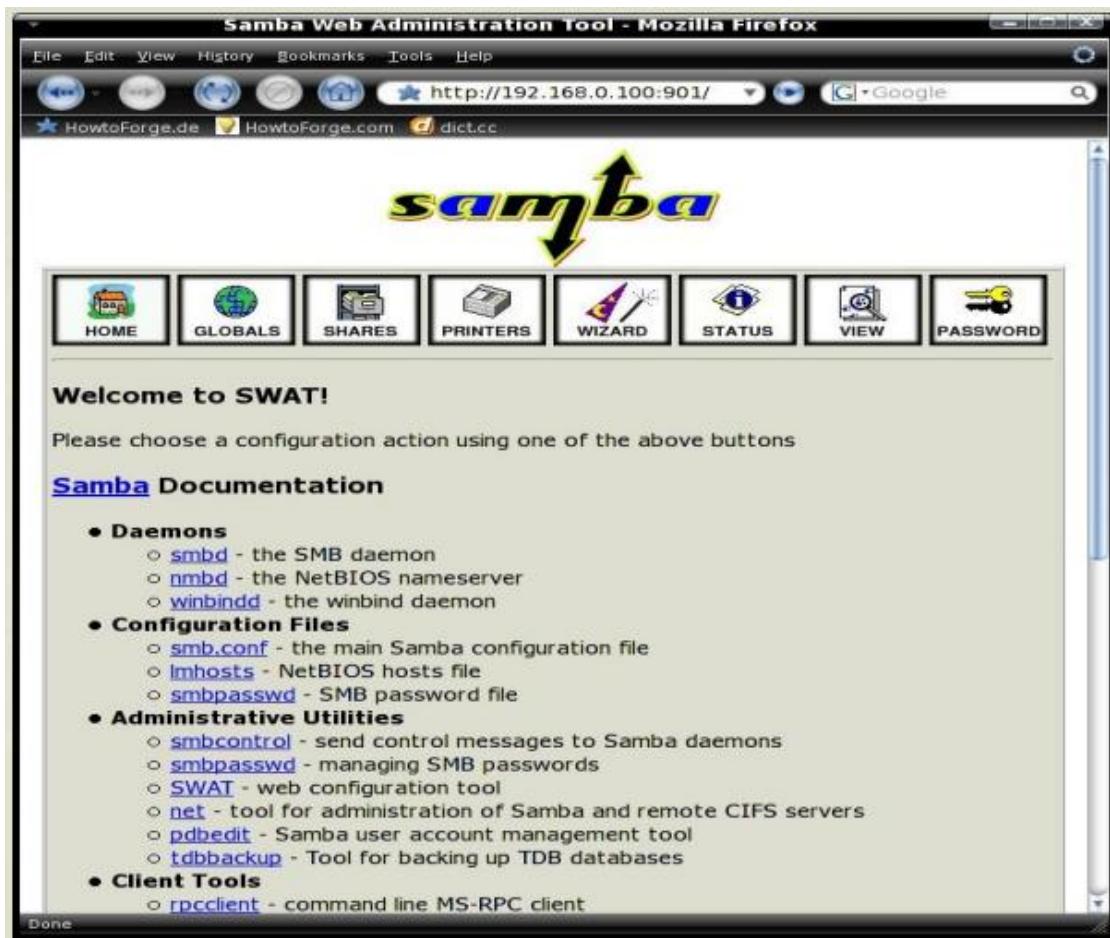
برای اینترنت و local مناسب است ولی اگر بخواهید بسته های 9000 بایتیه جای دور بفرستید از 10 روتر باید رد شود و مرتب باید در این مسیر شکسته شده و کد شود.

وقتی می خواهید اوراکل نصب کنید اعلام می کند که اگر کاربر زیادی دارد بافر شبکه را افزایش دهید.

سامبا

مايكروsoft سرويسى به نام SMB را برای ويندوز طراحى کرده که به کاربران اجازه می دهد تحت شرایطى منابع خود را به اشتراك بگذارند به همين دليل نام samba برای پروژه اشتراك فايل بين ويندوز و لينوكس انتخاب شد.

تنظيمات سامبا در /etc/samba/smb.conf قرار دارد که می تواند بهصورت دستی يا با swat يا linuxconf انجام شود.



- **Daemons**
 - [smbd](#) - the SMB daemon
 - [nmbd](#) - the NetBIOS nameserver
 - [winbindd](#) - the winbind daemon
- **Configuration Files**
 - [smb.conf](#) - the main Samba configuration file
 - [lmhosts](#) - NetBIOS hosts file
 - [smbpasswd](#) - SMB password file
- **Administrative Utilities**
 - [smbcontrol](#) - send control messages to Samba daemons
 - [smbpasswd](#) - managing SMB passwords
 - [SWAT](#) - web configuration tool
 - [net](#) - tool for administration of Samba and remote CIFS servers
 - [pdbedit](#) - Samba user account management tool
 - [tdbbackup](#) - Tool for backing up TDB databases
- **Client Tools**
 - [rpcclient](#) - command line MS-RPC client

برای دسترسی به منابع مشترک از دو دستور smbclient و smbmount استفاده می کنیم. سامبا می تواند ویندوزی یا لینوکسی باشد. برای مشاهده لیست منابع به اشتراک گذاشته شده از روی یک client لینوکسی باید دستور زیر را وارد نمایید:

- `smbclient -L samba_server_name`

با واردنمودن این دستور لیست تمام منابع به اشتراک گذاشته شده روی سرور مشاهده می شود . در client ویندوزی نیز با دستور net share می توان لیست منابع مشترک را دید.

توصیه می شود از راه دور samba را صدا نکنید چون clear text می فرستد و بهتر است تونل بزنید. راه اندازی در حالت معمولی ساده است ولی اگر LDAP و Active Control List هم در کنار آن باشد سخت می شود.

به عنوان تمرین می توانید سامبا را نصب کنید و زیر /tmp را به اشتراک بگذارید.

Squid Proxy Server

پروکسی سرور یک سخت افزار/نرم افزاری است که به عنوان یک فیلتر/اماسک بین شبکه های کامپیوتری محلی و شبکه های بزرگ تر مثل اینترنت عمل می کند.

به صورت پیش فرض کامپیوترهای خانگی برای استفاده از یک پروکسی تنظیم نشده اند ولی به راحتی می توانند برای استفاده از چندین نرم افزار پروکسی رایگان یا افزونه های مرورگر تنظیم شوند.

دلایل استفاده از پروکسی

• بهبود عملکرد

○ استفاده به عنوان کش سرور

○ bandwitch control

• فیلتر درخواست

○ جلوگیری از دسترسی به بعضی از وب سایت ها

○ جلوگیری از دسترسی به بعضی از پروتکل ها

○ تقسیم زمان

• وеб گردی به صورت ناشناس

○ مرور ووب بدون هیچ شناسایی

بعضی از IP ها می گویند که ما کش می فروشیم . در واقع خیلی از اطلاعاتی که می خواهید دریافت کنید روی کش آنها هست و با مقایسه tag ها و دیگر عناصر اگر تغییری نکرده بودند همان ها برای شما ارسال می شود.

گذاشتن کش سرور به شدت performance را بالا می برد . 18 سال پیش در شهرداری 8 تا هارد scsi گذاشتیم و بافرها را خیلی بزرگ تعریف کردیم آن زمان ورژن squid خیلی پایین بود ولی به خوبی رضایت را جلب کرد.

اگر 1 مگابایت پهنهای باند در شرکت دارید می توانید به یک کاربر 512 5 تا و به دو کارمند هر کدام 128 کیلو بایت بدھیم در واقع با پروکسی سرور می توان اینترنت را بین کاربران یک شبکه تقسیم کرد.

ما نمی توانیم با CNN invalid IP به CNN وصل شویم ، پروکسی سرور CNN را گول می زند ، خودش را به جای ما جا می زند درخواست ما را از CNN گرفته و به ما ارسال می کند.(واسطه است)

حتما باید در سایت ها آن را راه اندازی کنیم یعنی ممکن است در سازمان سامبا راه اندازی نکنیم ولی راه اندازی squid واجب است.

یک مطلب که برای cache می‌آید MD5 یا Message Digest version 5 آن را حساب می‌کند و نگه می‌دارد . دفعه بعد اگر همان فایل را بخواهید با مقایسه امضاها به جای کل فایل می‌فهمد که آن را دارد . به عنوان مثال اگر یک نقطه در عکس عوض شود آن MD5 عوض می‌شود:

```
[n.pardis@lpi ~]$ echo salam|md5sum
b17ee36a3a70c8373b415675d5ee21df -
[n.pardis@lpi ~]$ echo salam|md5sum A
4ca4db4fdded87178dfbe4cb50a0f24f -
[n.pardis@lpi ~]$ echo salAm|md5sum a
8907fcfae15f383f55ded8ac33c5684b -
[n.pardis@lpi ~]$ echo sal am|md5sum Aa..
a8d1c639deabb128ffe4064ed0c1d006 -
```

برای نصب پروکسی به چه چیزهایی احتیاج داریم؟

- نرم افزار پروکسی ○
- ... WinRoute ، MS ISA server ، squid ○
- سرور ○
- حداقل دو تا کارت شبکه ○
- ارتباط اینترنتی مستقیم (public ip address) ○
- switch/hub (اختیاری) ○
- آدرس IP خصوصی ○
- 10.0.0.1/8-172.16.0.1-192.168.0.1/24 ○

ویروس یاب ها ویروس ها را هم با امضایشان پیدا می‌کنند ، در غیر این صورت برای لینوکس یک میکریون ویروس داریم که کل آنها یک DVD حجم می‌گیرد ولی با امضا فقط یک میلیون 32 بایت را ذخیره می‌کنیم.

ایرانسل برای squid تعداد زیادی دیسک دارد و لیست مشتری پر مصرف و زمان نگهداری دارد در واقع چنین نرم افزارهایی باید سیاست گذاری داشته باشند.

نصب این نرم افزار ساده است ولی تنظیم Access Control List آن خیلی مشکل است .

بعضی مواقع سوال می‌شود که فلان فایل را نرم افزار در هر توزیعی در دایرکتوری های مختلفی قرار می‌دهد ؟ به راحتی می‌توان این محل را موقع نصب در فایل install قسمت prefix پیدا کرد:

```
[n.pardis@lpi squid-3.2.3]$ cat install  
To build and install the Squid Cache, type:
```

```
% ./configure --prefix=/usr/local/squid  
% make all  
% make install
```

To run a Cache, you will need to:

1. customize the squid.conf configuration file:

```
% vi /usr/local/squid/etc/squid.conf
```

2. Initialise the cache:

```
% /usr/local/squid/sbin/squid -z
```

3. start the cache:

```
% /usr/local/squid/sbin/squid
```

If you want to use the WWW interface to the Cache Manager, copy the cachemgr.cgi program into your httpd server's cgi-bin directory.

در بخش Access Control Lists می توانیم تعريف کنیم که چه کسی به چه چیزی می تواند دسترسی پیدا کند که به صورت پیش فرض http_access deny all در آن قرار دارم و باید همه قوانین را بالای آن تعريف کنیم. به عنوان مثال اول همه 20 ساله ها بروند سریازی ، بعد کسانی که مشکل پزشکی دارند معاف اند.

```
› acl INT_ACC src 192.168.1.0/255  
› acl WORKHOURS time SSMTWT 08:00-17:00  
› http_access deny INT_ACC WORKHOURS !INT_ACC  
› http_access allow INT_ACC
```

در مثال بالا اول قانون INT_ACC را تعريف می کنیم بعد دو خط پایین تر همه را به غیر از IP که در تعريف کردیم از دسترسی به اینترنت محروم می کنیم.

Redirectors

معمولا برای بستن تبلیغات روی صفحه های وب استفاده می شود. این کار را می توان در مرورگرها انجام داد ولی به هر صورت پهنهای باند گرفته می شود چون تا مرور گر می آید پس باید جلوتر این کار را انجام دهیم.

بسته های متعدد زیادی این کار را انجام می دهند که یکی از آنها adzapper است. می توان redirector را به squid اضافه کرد.

وقتی معاون اداره می گوید که همه با کیبوردها کار می کنند ولی کار پیش نمی رود ؛ می دانیم که می توانیم جلوی آنها را بگیریم.

Delay pool پاسخ squid برای مدیریت پهنهای باند است و به شما اجازه می دهد که مقدار پهنهای باندی که یک کامپیوتر خاص ، زیرشبکه یا پروکسی سرور ممکن است استفاده کند ، کنترل کنید.

خیلی وقت پیش در شهرداری ما با فیبر نوری به اکثر مناطق وصل بودیم و با مودم و ADSL به بقیه مناطق وصل بودیم ، منطقه 17 زنگ می زد که کسی سرعت پایین است چون منطقه ای که فیبر نوری داشت همه پهنهای باند را می گرفت . مقدار زیادی محاسبات انجام شد که برای هر منطقه چقدر delay بنویسیم . چنان این کارها زمان بر است و به آمارگیری نیاز دارد.

مکانیزم delay هم به این صورت است که فریم ها را که ارسال می کند بین آنها فاصله می گذارد.

در سازمان های بزرگ باید log های squid را نگه داریم چون ممکن است برای محاکم قضایی نیاز شود. ولی تماشای log کاربران از نظر اخلاقی کار درستی نیست.

در فایل cache.log آمار دانلود نرم افزار ، عکس و فیلم ثبت می شود. ISP ها باید حداقل 6 ماه log ها نگه دارند.

نصب squid

```
[root@lpi squid-3.2.3]# ./configure --help|less

'configure' configures Squid Web Proxy 3.2.3 to adapt to many kinds of
systems.

Usage: ./configure [OPTION]... [VAR=VALUE]...

To assign environment variables (e.g., CC, CFLAGS...), specify them as
VAR=VALUE. See below for descriptions of some of the useful variables.

Defaults for the options are specified in brackets.

Configuration:
-h, --help          display this help and exit
--help=short        display options specific to this package
--help=recursive   display the short help of all the included packages
-V, --version       display version information and exit
-q, --quiet, --silent do not print 'checking ...' messages
--cache-file=FILE  cache test results in FILE [disabled]
-C, --config-cache alias for '--cache-file=config.cache'
-n, --no-create    do not create output files
--srcdir=DIR        find the sources in DIR [configure dir or '..']

Installation directories:
--prefix=PREFIX      install architecture-independent files in PREFIX
                     [/usr/local/squid]
```

خیلی از اینها را باید disable یا enable کنید مثلا مدیر باید بگوید که کش سرور راه اندازی کنید و کلا باید صورت مسئله را کامل به ما بدهند.

امکان disable-optimization سرعت را بالا می برد ولی bug دارد.
squid همچنین گراف هایی از وضعیت شبکه می دهد که باید تحلیل شود چون مدیرها متوجه نمی شوند.
squid در نصب fault می دهد حتی اگر تمام مراحل نصب را به درستی انجام دهید.
squid نصب هست یا نه؟

```
[root@lpi squid-3.2.3]# ./configure
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /bin/mkdir -p
checking for gawk... gawk
checking whether make sets $MAKE... yes
```

که در نهایت خطای دهد و اعلام می کند یک دایرکتوری کم است.

یکی از مزیت های نصب سورس کد این است که ورژن های آن نسبت به پکیج ها جدیدتر است چون ردهت همیشه stable است و شما بهتر است سعی نکنید آخرین ورژن را نصب کنید.

NFS

این امکان را در اختیار کاربران قرار می دهد که بتوانند از فایل ها و دایرکتوری کامپیوترهای مختلف که به اشتراک گذاشته شده است بهره گیری نمایند.

در شهرداری یک سری Novell و یک سری ویندوز داشتیم و وقتی که لینوکس هم نصب کردیم باید به طریقی فایل سیستم های اینها را به هم وصل می کردیم؛ به جای سامبا از NFS استفاده کردیم چون در آن زمان ورژن سامبا پایین بود.

خادم NFS به منظور سرویس دهی بایستی سه نرم افزار ذیل را به همراه خود در حافظه داشته باشد:

portmap •

تقاضاهای سرویس گیرنده را به پروسس NFS ارسال نماید.

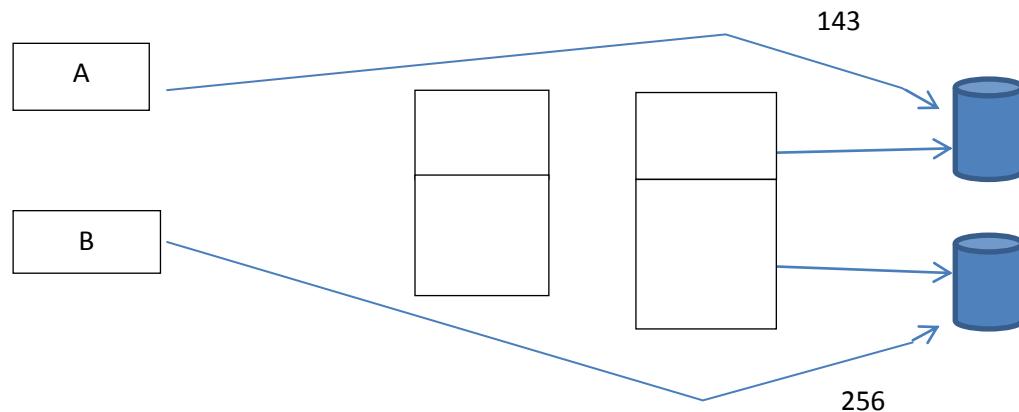
rpc.nfsd •

تقاضای فایل از راه دور را تجزیه و تحلیل نماید.

rpc.mountd •

وظیفه mount و umount کردن فایل سیستم را انجام دهد.

فرض کنید دو کامپیوتر داریم که روی یکی از آنها (در شکل، سمت راست) دو تا دیسک داریم. کاربر A می خواهد دیسک بالایی را و کاربر B دیسک پایینی را ببیند.



به عبارتی جلوی دیسک باید نرم افزاری باشد که داخلی ها را نگه دارد به عنوان مثال بداند که A با 143 به دیسک بالا و B با 256 به دیسک پایین متصل است . این نرم افزار همان portmap است.

: (remote procedure call) rpc

یک زمانی در نرم افزارها تابعی را صدا می کنید (مثل sin(x)) کامپایلر به لینکر می گوید که من نمی دانم sin(x) یعنی چه. لینکر از کتابخانه Math سیستم ، سینوس را آورده و به نرم افزار ما می چسباند. این rpc نیست!

ولی زمانی هم به ما می گویند روی یک ماشین دیگری نرم افزاری هست که جذر اعداد را تا 1000 رقم اعشار حساب می کند و شما در برنامه تان باید این نرم افزار را صدا بزنید . به عبارتی محاسبات در یک سیستم دیگر انجام می شود. sql server مايكروسافت با اين روش کار می کند یعنی client بعضی مواقع rpc می زند.

یکی از امکانات جالب در NFS ، Automount است. در شهرداری اسم سیستم Novell paeiz بود و با زدن دستور cd /paeiz می کرد که این شاخه روی یک سیستم دیگر قرار دارد؛ خودش اتوماتیک mount می کرد. به همین دلیل لیست دایرکتوری روی دیوار زده بودیم که مثلا فلان دایرکتوری در فلان محل قرار دارد.

در Automount اگر یک مدت استفاده نکنید خودش umount می کند.

برای اعمال تغییرات روی NFS باید فایل /etc/exports را تغییر دهید که هر خط دستور آن مشخص می نماید چه عنصری و با چه کامپیوترهایی به اشتراک گذاشته شده اند.

فرمت دستورات به این صورت است که اول مسیر دایرکتوری به اشتراک گذاشته شده و سپس host_list درج می شود:

```
[n.pardis@lpi ~]$ cat /etc/exports  
/tmp 192.168.100.1(rw)
```

طبق محتویات این فایل به ماشین گفته ایم که اگر کسی از 192.168.100.1 وصل شود اجازه دارد زیر /tmp و write کند.

با دستور زیر /tmp را بالا می آوریم که به اسم dummy در سیستم ما وجود دارد:

```
[root@lpi n.pardis]# mount 192.168.100.3:/tmp /dummy
```

بعدا می توانیم Automount را فعال کنیم.

این سیستم بسیار شبیه mapped drive مايكروسافت است که فقط در ویندوز کار می کند در حالی که NFS در همه سیستم عامل ها کار می کند.

یکی دیگر از استفاده های NFS این است که مثلاً اگر دایرکتوری ریشه (/) پاک شد ، اگر سیستم با سیستم دیگری sync باشد می توانید با user & pass خود می توانید به سیستم دوم وصل شوید و مثلاً اگر /home در سیستم اول خراب شد در exports این آدرس را به سیستم دوم map می کنیم و وقتی login می کنید /home سیستم دوم را می بینیم نه اول . در شهرداری کامپیوترها را با NFS ، samba دوستی لینوکس و ویندوز ولی NFS روی اکثر سیستم عامل ها وجود دارد.

می تواند به یکی از صورت های زیر باشد:

- نام کامپیوتر اگر در مجموعه قلمرو محلی باشد
- نام قلمرو به صورت کامل
- نام کامپیوتر شامل علامت * و به عبارتی wild card
- به صورت IP Address
- به صورت آدرس شبکه

اگر الان با mount دایرکتوری را به فایل سیستم وصل کنید پس از reboot آن را نمی بینیم برای رفع این مشکل می توان تحت دایرکتوری /etc/fstab نام کامپیوتر و دایرکتوری که قرار است به اشتراک گذاشته شود را قرار داد تا دسترسی به دایرکتوری اشتراکی اتوماتیک گردد. مثال:

نام کامپیوتر	دایرکتوری	server: /export/data	/mnt/database	nfs	default	0	2

وقتی که سیستم crash می کند fsck وارد می شود و عدد آخر ترمینال بالا اولویت های file system checking را بیان می کند که سیستم به تعداد و به صورت موازی fsck می فرستد که همه را چک کند.

```
[root@lpi n.pardis]# less /etc/fstab
```

```
#  
# /etc/fstab  
# Created by anaconda on Sat Sep 22 23:31:41 2012  
#  
# Accessible filesystems, by reference, are maintained under '/dev/disk'  
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info  
#  
/dev/mapper/vg_lpi-lv_root / ext4 defaults 1 1  
UUID=1dcb9e79-d1bd-4bf1-a46b-0ec1672676bd /boot ext4  
defaults 1 2  
/dev/mapper/vg_lpi-lv_swap swap swap defaults 0 0  
tmpfs /dev/shm tmpfs defaults 0 0  
devpts /dev/pts devpts gid=5,mode=620 defaults 0 0  
sysfs /sys sysfs defaults 0 0  
proc /proc proc defaults 0 0
```

وقتی لینوکس بالا می آید این فایل را می خواند و می فهمد که چه پارتیشنی را باید به چه دایرکتوری بدوزد، جنس پارتیشن چیست و ... شما می توانید یک پارتیشن را دراینجا `read only` کنید یا برای آن `owner` یا `group` تعريف کنید(مثل `devpts` در ترمینال بالا که `mod=620` دارد و یا `gid` برای آن تعريف شده است)

یک نکته امنیتی: ممکن است شخصی از شما بخواهد فلاش مموریش را در سیستم `mount` کنید و یک دایرکتوری به اسم خودش درست کنید . شما هم شاید فکر کنید که یک userID معمولی کاری نمی تواند بکند ولی دو دقیقه بعد سیستم `down` می شود!

شخص ممکن است از قبل در سیستم خودش (با root) فایلی درست کرده باشد و به آن `s` permision داده باشد و در بین فایل هایش در فلاش قرار داده باشد که با آن می تواند در سیستم شما قدرت root را به دست آورد.

اگر cd-rom ، فلاپی یا هارد را `mount` کردید با `option nosuid` این کار را انجام دهید:

```
[n.pardis@lpi ~]$ man mount
group      Allow an ordinary (i.e., non-root) user to mount the
filesystem
           if one of his groups matches the group of the device.
This
       option implies the options nosuid and nodev (unless
overridden
           bv subsequent options, as in the option line group,dev,suid).
```

یا ممکن است نرم افزاری اجرا کند که سیستم را کند می کند:

```
[n.pardis@lpi ~]$ man mount
noexec  Do not allow direct execution of any binaries on the mounted
        filesystem. (Until recently it was possible to run
binaries
        anyway using a command like /lib/ld*.so /mnt/binary. This
trick
        fails since Linux 2.4.25 / 2.6.0.)
```

یا فایل با پیشوند .. گذاشته باشد که نبینیم ولی برنامه ای مخرب داخل آن باشد.

در بعضی سیستم ها پارتیشن ها در این فایل با LABEL مشخص شده اند :

<code>LABEL=/shell-project</code>	<code>/shell-project</code>	<code>ext3</code>	<code>default</code>
<code>ts</code>	<code>1 2</code>		
<code>LABEL=/opt</code>	<code>/opt</code>	<code>ext3</code>	<code>default</code>
<code>ts</code>	<code>1 2</code>		
<code>LABEL=/home</code>	<code>/home</code>	<code>ext3</code>	<code>default</code>
<code>ts</code>	<code>1 2</code>		

فرض کنید در اداره لینوکس نصب کرده اید و هارد شما با /dev/hdb نام گذاری شده است و روزی سیستم شما بالا نمی آید؛ متوجه می شوید که رابط روی مادربرد خراب است و دو تا connector هم دارید که یکی hdb می شود و دیگری hdc و hdd . شما رابط را جدا می کنید و دومی را به آن متصل می کنید؛ تا لینوکس بالا می آید panic میدهد چون شما در fstab گفته اید hdb در حالی که الان hdc است ولی قتی LABEL می زنید دیسک ها را می گردد ببینند کدام به سیستم فعال و آماده است کدام یک LABEL بگذاریم.

روی پارتیشن label می گذارد و اگر دیسک را جای دیگری وصل کنید یا connector را عوض کنید سیستم panic نمی دهد:

```
[n.pardis@lpi ~]$ man e2label
```

E2LABEL(8)

E2LABEL(8)

NAME

e2label - Change the label on an ext2/ext3/ext4 filesystem

SYNOPSIS

```
e2label device [ new-label ]
```

DESCRIPTION

e2label will display or change the filesystem label on the ext2, ext3, or ext4 filesystem located on device.

If the optional argument new-label is not present, e2label will simply display the current filesystem label.

If the optional argument new-label is present, then e2label will set the filesystem label to be new-label. Ext2 filesystem labels can be at most 16 characters long; if new-label is longer than 16 characters, e2label will truncate it and print a warning message.

It is also possible to set the filesystem label using the -L option of tune2fs(8).

```
[root@lpi n.pardis]# nmap 127.0.0.1

Starting Nmap 5.21 ( http://nmap.org ) at 2012-11-14 23:11 IRST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000038s latency).
Not shown: 990 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
631/tcp   open  ipp
2049/tcp  open  nfs

Nmap done: 1 IP address (1 host up) scanned in 1.18 seconds
```