## آموزش جامع iptable (لینوکس فایروال)

iptables چیست ؟ iptables یک برنامه command line میباشد که برای config فیلترینگ packet و مجموعه قوانین برای linux 2.4.x و ورژن های بعدی میباشد و ابزاری بسیار مفید برای ادمین های شبکه است.   
پکیج iptables شامل ip6tables هم است که برای کانفیگ فیلترینگ پکت ipv6 مورد استفاده قرار میگیرد.

این مقاله به شما کمک میکند که بفهمید firewall در سیستم عامل linux چگونه کار میکند و اینکه iptables در linux چیست؟ فایروال سرنوشت پکت های ورودی و خروجی در سیستم را تعین میکند. iptables در حقیقت یک فایروال rule based میباشد و به صورت دیفالت روی اکثر سیستم عامل ها نصب است و به صورت پیش فرض بدون هیچ قانونی راه اندازی شده است. iptables در kernel 2.4 موجود بوده است و قبلا ipchains یا ipfwadm نامیده میشده است. iptables در حقیقت یکی از نخستین ابزاریست برای صحبت با kernel و تصمیم گیری در filter پکت ها. این مقاله به شما کمک میکند تا کامندهای اساسی iptables را یادبگیرید و همچنین قصد داریم قانون های عملی iptables را شرح دهیم که ممکن است شما هم به آنها نیاز داشته باشید.  
سرویس های متفاوت برای پروتکل های متفاوت استفاده میشوند. مانند :  
iptables مورد استفاده برای IPv4.  
ip6tables مورد استفاده برای IPv6.  
arptables مورد استفاده برای ARP.  
ebtables مورد استفاده برای Ethernet frames..

پیش نیازها :  
iptables نیازمند یک kernel هست که قابلیت filter کردن پکت (ip\_tables) را داشته باشد. تمامی release های کرنل 2.4 به بالا شامل آن هستند.

قابلیت های اصلی :  
لیست کردن محتوای مجموعه قوانین فیلتر کردن پکت ها  
اضافه حذف و تغییر رول ها در مجموعه قوانین فیلتر پکت ها  
  
فایل های اصلی iptables :  
  
/etc/init.d/iptables  
اسکریپت init برای start/stop/restart و ذخیره مجموعه قوانین.  
  
/etc/sysconfig/iptables  
مکانی که مجموعه قوانین ذخیره میشوند.  
  
/sbin/iptables  
باینری  
  
سه جدول زیر در iptables موجود هستند :

Filter  
NAT  
Mangle  
  
در حال حاضر 4 chain کلی وجود دارد :  
: INPUT  
chain پیش فرض برای ورود به سیستم  
  
: OUTPUT  
chain پیش فرض برای خروج از سیستم  
  
: FORWARD  
chain پیش فروض که پکت ها از طریق آن به اینترفیس دیگر ارسال میشوند  
  
: RH-Firewal-1-INPUT  
chain coustom که توسط کاربر تعریف میشود  
  
توجه :  
فایل های اصلی بالا ممکن است در لینوکس ubuntu کمی متفاوت باشند.  
  
طریقه start,stop,restart کردن فایروال iptables:

کد PHP:

# /etc/init.d/iptables start   
# /etc/init.d/iptables stop  
# /etc/init.d/iptables restart

برای استارت کردن iptables به هنگام بوت سیستم دستور زیر را اجرا کنید :

کد PHP:

#service iptables save

با دستور زیر وضعیت iptables را چک میکنیم :

: L-  
مجموعه قوانین را list میکند.  
  
: v-  
verbose میکند.  
  
: n-  
به فرمت numeric نمایش میدهد.

:php کد

[root@tecmint ~]# iptables -L -n -v  
  
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)  
 pkts bytes target     prot opt in     out     source               destination  
    6   396 ACCEPT     all  --  \*      \*       0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED  
    0     0 ACCEPT     icmp --  \*      \*       0.0.0.0/0            0.0.0.0/0  
    0     0 ACCEPT     all  --  lo     \*       0.0.0.0/0            0.0.0.0/0  
    0     0 ACCEPT     tcp  --  \*      \*       0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22  
    0     0 REJECT     all  --  \*      \*       0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-prohibited  
  
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)  
 pkts bytes target     prot opt in     out     source               destination  
    0     0 REJECT     all  --  \*      \*       0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-prohibited  
  
Chain OUTPUT (policy ACCEPT 5 packets, 588 bytes)  
 pkts bytes target     prot opt in     out     source               destination

--

دستور زیر قوانین iptables را به همراه شمارنده خط نمایش میدهد. با کمک گزینه line-numbers-- شما میتوانید قوانین را حذف یا اضافه کنید:

:PHP کد

[root@tecmint ~]# iptables -n -L -v --line-numbers  
  
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)  
num   pkts bytes target     prot opt in     out     source               destination  
1       51  4080 ACCEPT     all  --  \*      \*       0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED  
2        0     0 ACCEPT     icmp --  \*      \*       0.0.0.0/0            0.0.0.0/0  
3        0     0 ACCEPT     all  --  lo     \*       0.0.0.0/0            0.0.0.0/0  
4        0     0 ACCEPT     tcp  --  \*      \*       0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22  
5        0     0 REJECT     all  --  \*      \*       0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-prohibited  
  
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)  
num   pkts bytes target     prot opt in     out     source               destination  
1        0     0 REJECT     all  --  \*      \*       0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-prohibited  
  
Chain OUTPUT (policy ACCEPT 45 packets, 5384 bytes)  
num   pkts bytes target     prot opt in     out     source               destination

---

: با دستور زیر انجام میشود flushing یا iptables حذف قوانین

:php کد

[root@tecmint ~]# iptables -F

در حقیقت این دستور تمامی قوانین را از جداول پاک میکند. قبل از اجرا کردن این کامند از مجموعه قوانین تان backup بگیرید. برای حذف یا اضافه کردن قوانین ابتدا نگاهی به قوانین موجود در chain ها می اندازیم. دستور زیر باید مجموعه قوانین را در chain های input و output به همراه شمارنده خطوط نمایش دهد که کمک خواهد کرد بتوانیم قوانین را حذف یا اضافه کنیم:

[root@tecmint ~]# iptables -L INPUT -n --line-numbers  
  
Chain INPUT (policy ACCEPT)  
num  target     prot opt source               destination  
1    ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED  
2    ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0  
3    ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0  
4    ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22  
5    REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-prohibited

----

:php کد

[root@tecmint ~]# iptables -L OUTPUT -n --line-numbers  
Chain OUTPUT (policy ACCEPT)  
num  target     prot opt source               destination

-----

دستور زیر را باید اجرا کنیم: حذف کنیم chain input اگر بخواهیم قانون شماره 5 را از

:php کد

[root@tecmint ~]# iptables -D INPUT 5

--------

به شماره 4 و 5 دستور زیر را اجرا میکنیم: chain input بین rule برای وارد کردن یا اضافه کردن قانون

:php کد

[root@tecmint ~]# iptables -I INPUT 5 -s ipaddress -j DROP

---

در این مقاله هدف ما آموزش کاربرد و استفاده پایه از

iptable

برای مبتدیان بود.

--------------------------------------------------------------------------------------------------------------------------

**مهمترین دستورات فایروال ( linux iptables ) در لینوکس**

[](https://parspack.com/images/stories/news/iptables.jpg)Netfilter نام فایروال محلی سیستم عامل لینوکس می باشد که در لایه کرنل عمل می کند و از iptables برای کنترل آن جهت مدیریت و فیلتر کردن پکت ها بر روی IPv4 و IPv6 استفاده می گردد. با توجه به اهمیت مسئله امنیت سرور و بخصوص سرور های لینوکس ، در این مطلب تعدادی از رایج ترین دستورات فایروال لینوکس iptables تشریح می گردد که می تواند در خیلی از مواقع مشکلات مدیر سرور را مرتفع نماید. لازم به توضیح است که می بایست در استفاده از دستورات فایروال دقت لازم را مبذول فرمائید، چرا که یک اشتباه ساده، می تواند منجر به بسته شدن دسترسی شما به سرور گردد، البته در کلیه سرور های ارائه شده توسط پارس پک ، با مراجعه به پنل ابر سرور در بخش کاربری و امکان VNC می توانید اشکلات بوجود آمده در دسترسی به سرور را بررسی و مرتفع نمایید.

در این مطلب فرض بر این است که شما از دسترسی روت ( root ) برای وارد کردن دستورات استفاده می کنید.

**1.  نمایش وضعیت فایروال:**

برای مشخص شدن رول هایی که در حال حاضر در iptables وجود دارند، می توانید از دستور زیر استفاده فرمائید:

iptables -L -n -v

خروجی دستور می تواند به شکل زیر باشد:

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)

pkts bytes target prot opt in out source destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)

pkts bytes target prot opt in out source destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)

pkts bytes target prot opt in out source destination

 اگر در فایروال خود از قبل رول هایی اضافه کرده اید، احتمالا وضعیت به شکل زیر خواهد بود:

Chain INPUT (policy DROP 0 packets, 0 bytes)

pkts bytes target prot opt in out source destination

0 0 DROP all -- \* \* 0.0.0.0/0 0.0.0.0/0 state INVALID

394 43586 ACCEPT all -- \* \* 0.0.0.0/0 0.0.0.0/0 state RELATED,ESTABLISHED

93 17292 ACCEPT all -- br0 \* 0.0.0.0/0 0.0.0.0/0

1 142 ACCEPT all -- lo \* 0.0.0.0/0 0.0.0.0/0

Chain FORWARD (policy DROP 0 packets, 0 bytes)

pkts bytes target prot opt in out source destination

0 0 ACCEPT all -- br0 br0 0.0.0.0/0 0.0.0.0/0

0 0 DROP all -- \* \* 0.0.0.0/0 0.0.0.0/0 state INVALID

0 0 TCPMSS tcp -- \* \* 0.0.0.0/0 0.0.0.0/0 tcp flags:0x06/0x02 TCPMSS clamp to PMTU

0 0 ACCEPT all -- \* \* 0.0.0.0/0 0.0.0.0/0 state RELATED,ESTABLISHED

0 0 wanin all -- vlan2 \* 0.0.0.0/0 0.0.0.0/0

0 0 wanout all -- \* vlan2 0.0.0.0/0 0.0.0.0/0

0 0 ACCEPT all -- br0 \* 0.0.0.0/0 0.0.0.0/0

Chain OUTPUT (policy ACCEPT 425 packets, 113K bytes)

pkts bytes target prot opt in out source destination

Chain wanin (1 references)

pkts bytes target prot opt in out source destination

Chain wanout (1 references)

pkts bytes target prot opt in out source destination

برای نمایش رول ها به همراه شماره خط هر رول از دستور زیر استفاده فرمائید:

iptables -n -L -v --line-numbers

احتمالا خروجی دستور، مشابه زیر خواهد بود:

Chain INPUT (policy DROP)

num target prot opt source destination

1 DROP all -- 0.0.0.0/0 0.0.0.0/0 state INVALID

2 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 state RELATED,ESTABLISHED

3 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0

4 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0

Chain FORWARD (policy DROP)

num target prot opt source destination

1 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0

2 DROP all -- 0.0.0.0/0 0.0.0.0/0 state INVALID

3 TCPMSS tcp -- 0.0.0.0/0 0.0.0.0/0 tcp flags:0x06/0x02 TCPMSS clamp to PMTU

4 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 state RELATED,ESTABLISHED

5 wanin all -- 0.0.0.0/0 0.0.0.0/0

6 wanout all -- 0.0.0.0/0 0.0.0.0/0

7 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0

Chain OUTPUT (policy ACCEPT)

num target prot opt source destination

Chain wanin (1 references)

num target prot opt source destination

Chain wanout (1 references)

num target prot opt source destination

توضیح دستور :

L- : لیست رول ها

v- : نمایش جزئیات

n- :  نمایش ip و پورت بصورت عددی

**2. تغییر وضعیت فایروال :**

از دستورات زیر برای روشن ، خاموش و ریست فایروال می توان استفاده کرد:

service iptables stop

service iptables start

service iptables restart

اگر می خواهید فایروال را خاموش و پس از ریستارت سرور نیز وضعیت به همین منوال باقی بمانید ، دستور زیر را وارد نمائید:

service iptables stop

chkconfig iptables off

**3. حذف قوانین و رول های فایروال :**

ابتدا به کمک دستورات زیر شماره خط رول را بدست آورید:

iptables -L INPUT -n --line-numbers

iptables -L OUTPUT -n --line-numbers

iptables -L OUTPUT -n --line-numbers | less

iptables -L OUTPUT -n --line-numbers | grep 202.54.1.1

حال به عنوان مثال برای حذف رول موجود در خط شماره 4 می توانید از دستور زیر استفاده فرمائید:

iptables -D INPUT 4

و یا از دستور زیر برای حذف قوانین مروبطه به ای پی مورد نظر خود استفاده فرمایید:

iptables -D INPUT -s 202.54.1.1 -j DROP

توضیح:

D- : حذف یک یا چند رول از زنجیره انتخاب شده

**3.1 حذف کلیه رول ها از فایروال ( Flush iptables ):**

برای حذف کلیه رول ها می توانید از دستور زیر استفاده فرمائید:

iptables -F

و اگر در تیبل nat و mangle نیز رولی اضافه کردید ، دو دستور زیر را نیز لازم است وارد نمائید:

iptables -t nat -F

iptables -t mangle -F

برای مشاهده وضعیت رول ها مجدد از دستور زیر استفاده فرمائید:

iptables -L -v -n

**4.  نحوه مسدود کردن ترافیک ورودی ، خروجی و یا فوروارد در فایروال :**

برای مسدود کردن کل ترافیک می توانید از دستورات زیر استفاده فرمائید ( دقت نمائید که این دستورات می تواند مانع از دسترسی شما به سرور گردد.) :

iptables -P INPUT DROP

iptables -P OUTPUT DROP

iptables -P FORWARD DROP

برای ذخیره نمودن تغییرات اضافه شده در فایروال می توانید از دستور زیر استفاده فرمائید:

service iptables save

**5. نحوه مسدود نمودن یک IP بر روی سرور :**

به کمک دستورات اول شما می توانید ترافیک ورودی از سوی ای پی 1.2.3.4 و به وارد کردن دستور دوم کل ترافیک ورودی برای رنج ای پی مورد مثال مسدود می گردد.

iptables -A INPUT -s 1.2.3.4 -j DROP

iptables -A INPUT -s 192.168.0.0/24 -j DROP

**6. نحوه مسدود کردن ترافیک ورودی بر روی یک پورت خاص :**

به کمک دستورات زیر می توان ترافیک ورودی را بر روی پورت 80 مسدود نمائید. لازم به توضیح است که معمولا وب سرور از پورت 80 برای نمایش وب سایت استفاده می کند.

iptables -A INPUT -p tcp --dport 80 -j DROP

iptables -A INPUT -i eth1 -p tcp --dport 80 -j DROP

برای مسدود کردن ترافیک ورودی بر روی پورت 80 تنها برای یک IP و یا یک رنج IP می توانید از دستورات زیر استفاده فرمائید:

iptables -A INPUT -p tcp -s 1.2.3.4 --dport 80 -j DROP

iptables -A INPUT -i eth1 -p tcp -s 192.168.1.0/24 --dport 80 -j DROP

**7. نحوه مسدود نمودن ترافیک خروجی:**

برای مسدود نمودن ترافیک خروجی برای یک IP خاص و یا یک رنج IP می توانید از دستورات زیر استفاده فرمائید:

iptables -A OUTPUT -d 1.2.3.4 -j DROP

iptables -A OUTPUT -d 192.168.1.0/24 -j DROP

iptables -A OUTPUT -o eth1 -d 192.168.1.0/24 -j DROP

**8. لاگ کردن فعالیت ها:**

خیلی از مواقع نیاز استفا تا از ریز فعالیت هایی که بر روی شبکه سرور در حال وقوع است مطلع شوید برای مثال برای جلوگیری از IP spoofing بر روی اینترفیس eth1  می توانید از دستور زیر استفاده فرمائید. در این دستور در حالیکه  ترافیک مربوطه مسدود می گردد ، تلاش های صورت گرفته نیز با پیشوند IP\_SPOOF A در لاگ ذخیره می شود:

iptables -A INPUT -i eth1 -s 10.0.0.0/8 -j LOG --log-prefix "IP\_SPOOF A: "

iptables -A INPUT -i eth1 -s 10.0.0.0/8 -j DROP

معمولا لاگ ها در مسیر /var/log/messages قرار دارند که البته با توجه به ویرایش لینوکس مورد استفاده می تواند متفاوت باشد. برای مشاهده لاگ های مربوطه می توانید از دستورات زیر استفاده فرمائید:

tail -f /var/log/messages

grep --color 'IP SPOOF' /var/log/messages

برای جلوگیری از حجیم شدن لاگ فایل می توان از قابلیت سوکت -m در دستور استفاده کرد که به کمک آن می توان تنظیم نمود که برای مثال در هر 5 دقیقه بیش از 7 مورد را در  لاگ ذخیره ننماید:

iptables -A INPUT -i eth1 -s 10.0.0.0/8 -m limit --limit 5/m --limit-burst 7 -j LOG --log-prefix "IP\_SPOOF A: "

iptables -A INPUT -i eth1 -s 10.0.0.0/8 -j DROP

**9. چگونه از باز یا بسته بودن یک پورت توسط فایروال بر روی شبکه آگاه شویم؟**

برای آگاهی از باز بودن و نبودن پورت می توانید از دستور زیر استفاده فرمایئد:

netstat -tulpn

برای اطلاع از اینکه آیا پورت پورت tcp 80 باز است یا خیر از دستور زیر استفاده فرمایید:

netstat -tulpn | grep :80

اگر پورت 80 باز نبود دستور زیر را وارد نمائید:

service httpd start

به کمک دستور زیر اطمینان یابید که پورت 80 بر روی فایروال بسته نشده است:

iptables -L INPUT -v -n | grep 80

در صورتیکه مسدود بود می توانید از دستور زیر استفاده فرمایئد:

iptables -A INPUT -m state --state NEW -p tcp --dport 80 -j ACCEPT

service iptables save

**10. غیر فعال کردن فایروال :**

[iptables-2](https://parspack.com/images/stories/news/iptables-2.jpg)اگر با مشکلی در دسترسی به فایروال روبرو هستید و یا فایروال مانع از کارکرد صحیح سرویس های اصلی سرور شما شده است می توانید به کمک دستورات زیر فایروال را خاموش نمائید. با استفاده از دستور دوم نیز پس از ریستارت سرور وضعیت به همین منوال باقی خواهد ماند و فایروال مجدد فعال نمی گردد، مگر آنکه شما از نرم افزار جانبی مانند csf استفاده کرده باشید که بدین ترتیب فایروال مجدد فعال می گردد و بدین شکل قابلیت غیرفعال شدن نخواهد داشت:

service iptables stop

chkconfig iptables off

برای آشنایی بیشتر با iptables و مطالعه manual های آن می توانید از دستور زیر استفاده فرمائید:

man iptables

برای مشاهده راهنمای آن دستور زیر قابل استفاده خواهد بود:

iptables -h

اگر بدنبال راهنمای فایروال تنها برای یک دستور خاص هستید نیز سینتکس زیر استفاده فرمائید:

iptales -j DROP -h

--------------------------------------------------------------------------------------------------------------------------

**نکات IPTables**  
  
IPTable فایروال توکار لینوکس هستش که قدرت بی همتایی داره همین قدر ازش بگم که روتر میکروتیک یه جورایی تمامی قدرتش رو از IPTable می گیره...  
با IPTable هر چیزی که به فکرتون برسه در مورد امنیت شبکه ، مدیریت شبکه و مسیر یابی اطلاعات می تونید پیاده سازی کنید... فقط کافیه یک لینوکس نصب کنید به تعداد دلخواه کارت شبکه اضافه کنید و IPTable رو کانفیگ کنید. قدرت این فایروال هم طبیعتا بستگی به Platform ای داره که داره روش اجرا می شه یعنی می تونید به راحتی ترافیک کل یک کشور رو با یک سخت افزار صحیح و یک کانفیگ بهینه آنالیز کنید و در موردش تصمیم گیری کنید...  
با قدرت لینوکس و IPTable سیستم های فایروال زیادی ساخته شده مثلا آستارو ، میکروتیک ، IPFire و خیلی دیگه از سخت افزار ها و نرم افزار هایی که توی بازار می بینید که تا دلتون بخواد الان در بازار زیاد شدن و هر کسی هم ادعا می کنه مال ما قویتره... من شخصا گمان می کنم که JUNOS و IOS و همه شرکت هایی غول دیگه ای هم که سیستم عامل روتر و فایروال می سازن همشون از لینوکس های بهینه سازی شده و سفارشی سازی شده استفاده می کنن... حالا که همه دارن فایروال می سازن ما چرا برای خودمون یکی نسازیم؟  
با این مقدمه تو فکر این هستم که شروع به آموزش یک دوره مقدماتی تنظیمات IPTable و ساخت فایروال شخصی با لینوکس در این فروم با کمک همه بچه های خوب اینجا هستم...  
برای شروع بهتره که همه ما یک لینوکس روی سیستمون نصب کنیم و دو تا کارت شبکه روش بزاریم و اگه هم نخوایم منابع سیستمون رو اشغال کنیم می شه روی یک نرم افزار Emulator نظیر VirtualBox یا VMWare استفاده کنیم. برای ورژن لینوکس هم هر چی که خودتون راحترین ولی ئیشنهاد من نصب یک لینوکس سبک و لخت هستش ( نیازی به نصب محیط Desktop مثلا gnum یا نرم افزار های اداری نیست ) مثلا یک ابونتو سرور یا یک CentOS یا هر چی دیگه ای که خودتون می دونید. خوبی لینوکس لخت سرعت و نیاز کمش به سخت افزار قوی و هستش. مثلا بعدا می تونید فایروالتون رو روی یک Mainboard کوچیک و ارزون نصب کنید و با زدن چهار کارت شبکه و خرید یک باکس با برچسب شرکت خودتون بفروشید و بگید Cisco کیه داداش

---

برای شروع وارد مباحث تکنیکی IPTables نمیشیم

ورژن های مختلفی از IPTables وجود داره برای مثال:  
IPTables این ورژن مختص IPv4 هستش.  
IP6Tables همونطور که از اسمش پیداست برای IPv6 طراحی شده.  
arptables برای مدیریت بسته های ARP  
ebtables برای مدریت فریم های Ethernet بوجود آمده.  
دستورات IPTables بسته به نوع کاربردشون در جداولی (Table) ذخیره می شن که این جداول خود شامل زنجیره ای ( Chain ) از فرامین هستند.  
یکی از مهمترین این جداول جدول Filtering هستش که وظیفه اون سیاست گذاری و دادن مجوز برای ورود و خروج بسته های TCP/IP به سیستم است. این جدول شامل سه زنجیر INPUT برای ترافیک ورودی به سیستم ، OUTPUT برای ترافیک خروجی از سیستم و FORWARD برای ترافیک فوروارد شده از سیستم است.  
از دیگر جداول IPTABLES می شه به موارد زیر اشاره کرد:  
....... table  
nat table  
mangling table  
که بصورت مختصر جدول ....... رو معرفی کردیم و موارد دیگه رو در زمان شروع مباحث اون جدول معرفی می کنیم.  
به عنوان اولین گام مایل هستیم که اجازه ندیم به هیچ کامپیوتری که سرور ما رو PING کنه یعنی اصطلاحا می خوایم PING رو ببندیم...  
برای این کار باید به ترافیک پروتکل ICMP اجازه ورود ندهیم... برای این منظور باید به صورت زیر عمل کنیم:  
با مجوز کاربر ROOT وارد سیستم شوید.  
فرمان زیر رو اجرا کنید:  
iptables -A INPUT -p ICMP -j DROP  
فرمان زیر میگه که در انتهای ( بخاطر استفاده از سویچ A یعنی Append ) زنجیر INPUT ( یعنی ترافیک ورودی) پروتکل ICMP را DROP کن.  
نوع پروتکل رو با سویچ p- مشخص می کنیم که در جلوی این سویچ می تونیم موارد TCP ، UDP ، ICMP و all رو بنویسیم.  
برای مشخص کردن نحوه برخورد با ترافیک مورد نظر از سویچ j- استفاده می کنیم به معنی jump به موارد DROP ، LOG ، ACCEPT ، REJECT هستش. در اینجا تفاوتی که بین DROP و REJECT هست و اون هم این هستش که در DROP برای کاربر پیامی مبنی بر حذف بسته اش ارصال نمی شه ولی در REJECT به فرستنده پیام یک بسته عدم قبول ترافیک ارسال میشه.

-------

در این قسمت با دو سویچ مهم از IPTABLES آشنا می شویم:  
سویچ s- و d- برای مشخص کردن source و destination در یک بسته TCP/IP به ترتیب از سویچ های معرفی شده استفاده می کنیم.  
فرض کنید تصمیم داریم که به کاربران شبکه 192.168.1.0/24 اجازه دهیم شبکه 192.168.2.0/24 رو PING کنن برای این کار باید یک rule در chain مربوط به خروج ( OUTPUT ) بسته ها در جدول ....... اضافه کنیم:  
iptables -A INPUT -p ICMP -s 192.168.1.0/24 -d 192.168.2.0/24 -j ACCEPT  
به همین سادگی می توانید منبع و مقصد یک بسته TCP/IP رو در فرامین خود مشخص کنید. بدیهی است که برای مشخص کردن یک آدرس خاص نیز می توان به همین روش استفاده کرد. مثلا می خواهیم به ماشین admin شبکه اجازه دهیم که کل شبکه رو PING کنه و بقیه ماشین ها این اجازه رو در شبکه 192.168.1.0/24 نداشته باشن ٍ آدرس ماشین Admin برابر 192.168.1.2 می باشد:  
برای این منظور از دو عدد rule به صورت زیر استفاده می کنیم:  
iptables -A INPUT -p ICMP -s 192.168.1.0 -d 192.168.1.0/24 -j ACCEPT  
iptables -A INPUT -p ICMP -s 192.168.1.0/24 -j REJECT  
  
برای پاک کردن محتویات جداول IPTables می توانید از فرمان iptables -F یعنی FLUSH استفاده کنید.  
  
برای دیدن محتویات جداول IPTables می توانید از فرمان iptables -L استفاده کنید:  
  
خروجی:  
<LEFT>  
root@yunix-LT:~# iptables -L  
Chain INPUT (policy ACCEPT)  
target prot opt source destination  
ACCEPT icmp -- 192.168.1.0/24 192.168.2.0/24  
ACCEPT icmp -- 192.168.1.0 192.168.1.0/24  
REJECT icmp -- 192.168.1.0/24 anywhere reject-with icmp-port-unreachable  
  
Chain FORWARD (policy ACCEPT)  
target prot opt source destination  
  
Chain OUTPUT (policy ACCEPT)  
target prot opt source destination  
<LEFT/>  
در پست بعدی به تحلیل خروجی فرمان iptables -L می پردازیم.

----

در این جا به بررسی خروجی دستور iptables -L می پردازیم:  
در خروجی فرمان فوق ستون هایی به شرح زیر نمایش داده شده است:  
target prot opt source destination  
در target ما می تواین هدف پرش (jump) قانون مورد نظر را ببینیم که از بین گزینه های ACCEPT, REJECT,DROP,LOG می تواند باشد.  
در ستون prot می توانیم پروتکل مورد استفاده در این قانون را مشاهده کنیم یعنی udp,tcp و ...  
در ستون opt تنظیمات و جزئیات مربوط به هر rule نمایش داده می شود.  
ستون های source و destination نیز که منبع و مقصد بسته های را مشخص می کند.  
در iptables ما می تواینم به جای استفاده از سویچ های کوتاه شده نظیر j- از کلمات کامل شده استفاده کنیم مانند jump-- برای j- یا source-- بجای s- بدین ترتیب خوانایی فرمانهای iptables بیشتر شده و بخاطر آوری آنها نیز راحتر می گردد.  
یک سناریو کوچک :  
فرض کنید از یک ip آدرس به شماره 11.12.13.14 یکی از سرویس دهنده های شما مورد حمله DOS قرار گرفته و شما مایل به Block کردن ترافیک ارسالی از این آدرس IP هستید برای این کار به صورت زیر عمل کنید:  
iptables --Append INPUT --protocol all --source 11.12.13.14 --jump DROP  
تا بدین جا با تنظیمات ساده از iptables آشنا شدیم حال می خواهیم بدانیم که چطور می توان بر روی ترافیک لایه 4 یعنی لایه transport یا به زبان ساده تر کنترل ترافیک پورت ها سیاست امنیتی اعمال کرد:  
برای این منظور از سویچ dport-- استفاده می شود.  
مثال1: فرض کنید می خواهیم ترافیک ورودی به شبکه از پورت 5050 را Block کنیم:  
iptables -A INPUT -p TCP --dport 5050 -j DROP  
مثال2: در این مثال می خواهیم ترافیک پورت telnet یعنی 23 را برای شبکه 192.168.1.0/24 بسته و به جای آن ترافیک ssh یعنی پورت 22 را باز کنیم:  
iptables -A OUTPUT -p tcp --dport 23 -s 192.168.1.0/24 -j DROP  
iptables -A OUTPUT -p tcp --dport 22 -s 192.168.1.0/24 -j ACCEPT  
لازم به ذکر است که بجای سویچ dport-- می تواین از سویچ destination-port--نیز استفاده کرد.

---------

در ادامه مباحث مطرح شده :  
اولویت بازبینی قوانین مربوط به IPTables بر طبق مکان درج آنها در هر Chain مشخص می شود. به عنوان مثال در دو قانون زیر:  
iptables -A OUTPUT -p all -j DROP  
iptables -A OUTPUT -p all -j ACCEPT\*  
دسترسی کل شبکه قطع خواهد شد به دلیل اینکه قانون iptables -A OUTPUT -p all -j DROP بالاتر از قانون دسترسی به کل پروتکل های شبکه است ( قانون دوم).  
در سویچ A- هر گاه ما قانون جدیدی بنویسیم در پایین زنجیر مربوطه درج خواهد شده برای اینکه یک قانون را در موقعیت خاصی در زنجیر ها درج کنیم باید از سویچ I-\* به معنی INSERT استفاده کنیم. به مثال زیر دقت کنید در این مثال ما می خواهیم قانون خود را در مکان دوم زنجیر مربوط به OUTPUT درج کنیم:  
iptables -I OUTPUT 2 -p tcp --dport 21 -s 192.168.2.1 -j ACCEPT  
همانطور که ملاحظه می فرمایید برای مشخص کردن مکان درج Rule\* محل درج را به صورت عددی با شروع از یک بعد از اسم زنجیر (Chain) می نویسیم در اینجا بعد از زنجیر OUTPUT\*محل درج را نوشته ایم.  
به همین ترتیب برای حذف یک قانون خاص از مجموعه قوانین یک زنجیر از سویچ D- به معنی DELETE استفاده می کنیم. به مثال زیر دقت کنید.  
در این مثال می خواهیم قانون شماره سوم را از زنجیر INPUT حذف کنیم.  
iptables -D INPUT 3  
دستور فوق باعث حذف قانون شماره سوم زنجیر INPUT می شود.  
حال به معرفی دو ویژگی مدیریت بسته های TCP/IP\*می پردازیم:  
سویچ sport-- خواهر سویچ dport-- محسوب می شود و برای مشخص کردن شماره \*پورت منبع بسته های TCP/IP\*است. با یک مثال کاربرد سویچ sport--\*را نشان می دهیم:  
فرض کنید مایل هستیم که ترافیک ورودی به سمت پورت ۴۴۴۴ را مسدود کنیم برای این کار از قانون زیر استفاده می کنیم:  
iptables -A INPUT -p tcp --sport 4444 -j DROP  
با استفاده از سویچ های sport-- و dport-- می توانید قوانین خود را بر روی \*پروتکل های ارتباطی مورد نظر خود اعمال کنید.

--------