

# کار با شبکه در خط فرمان لینوکس

تاریخ: ۱۳۸۸/۸/۲

نویسنده ها:

فرید احمدیان

مهدی هنرمند

برای وصل کردن کامپیوترمون به شبکه مورد نظر اول از هر چیزی باید چک کنیم ببینیم آیا کابل به کارت شبکه وصل هست یا نه ؟ و بعد از اون چک کنیم ببینیم که آیا کارت شبکه ما up هست یا به قول ویندوزی ها enable یا نه ؟  
برای این کار ما در لینوکسمون از دستور زیر استفاده میکنیم:

## **mii-tool**

خوب خروجی این دستور به ما وضعیت فعلی کارت شبکه و کابل شبکمونو میده.  
مثلا:

```
root@farid-notebook:~# mii-tool  
eth0: negotiated 100baseTx-FD flow-control, link ok
```

که نشون میده همه چیز مرتبه و کابل وصله! اگه کابل وصل نبود خروجی اینطوری میشد:

```
root@farid-notebook:~# mii-tool  
eth0: no link
```

و حتی اگه کارت شبکه مون به هر دلیلی غیر فعال باشه با این خروجی روبرو میشیم:

```
root@farid-notebook:~# mii-tool  
no MII interfaces found
```

همون طور که میبینید با این دستور خیلی چیزا میفهمیم ! اگه حالت اول باشه که خوش به حالمون ، اگه حالت دوم باشه که باید چک کنیم دو سر کابل درست وصل شده یا نه و یا کابلمون قطعی نداره که؟  
و اما برای حالت سوم باید کارت شبکه مورد نظرمونو فعال کنیم.

اما اول باید بدونیم کارت شبکه ها رو در لینوکس با چه اسمی نشون میدم؟  
معمولا اسم کارت شبکمون به صورته eth0 و یا eth1 و ... نشون داده میشه ،همینطور الا اخر هر چند تا که کارت شبکه داشته باشیم اینجا هم عدد داریم!  
خوب در حالت سوم به ما گفت که کارت شبکمون فعال نیس ! با فرض اینکه اسم کارت شبکمون eth0 باید دستور زیرو برای فعال سازیش بزنینم:

## **ifconfig eth0 up**

دستور ifconfig کارهای زیادی میکنه ولی در اینجا خیلی ساده تونستیم کارت شبکمونو فعال کنیم و البته یه آرگومان دیگش هم در این دستور down به جای up بود اگر قصد ما غیر فعال کردن کارت شبکمون بود!

به هر حال با زدن این دستور اگه اروری نگیریم ، با mii-tool چک کنیم ببینیم که همه چی ok شده!

تا اینجا کار تونستیم از نظر سخت افزاری مطمئن بشیم که همه چی ردیفه و حالا مونده تنظیم dns , default gateway , ip تا بتونیم هم در یک شبکه باشیم و هم احيانا از اینترنت استفاده کنیم!

خوب ما برای بعضی کارای تنظیماتیمون با دو گونه دستور طرفیم!  
دستورات موقت و دستورات دائمی که دستورات موقت با یک بار خاموش شدن کامپیوتر  
اثرشونو از دست میدن و بی تاثیر میشن.  
و تنظیمات دائمی که معمولا داخل یک سری فایل متنی ذخیره میشن.

تنظیمات **dns** معمولا در فایل زیر ذخیره می شود:

**/etc/resolv.conf**

برای وارد کردن یک dns به آدرس 1.2.3.4 خط زیر را به فایل بالا اضافه کنید:

**nameserver 1.2.3.4**

مرحله ی بعد گرفتن IP است. در صورت استفاده ی روتر از DHCP

**dhclient eth0**

و یا اگر روتر از DHCP استفاده نمی کند برای گرفتن IP به صورت دستی:

**ifconfig eth0 192.168.1.100 netmask 255.255.255.0 up**

و در نهایت برای اضافه کردن درگاه خروجی پیش فرض :

**route add default gw 192.168.0.1**

حال برای آزمایش درست بودن تنظیمات فوق میتونیم یک **ip** موجود تو اینترنت رو **ping** کنیم:

**ping 4.2.2.4**

و برای تست **dns** از دستور زیر استفاده میکنیم:

**dig yahoo.com**

## راه اندازی Apache Web Server در لینوکس slackware

خوب همین طور که دوستان طراح وب می دونن برای این که صفحات طراحی شده ی خودمون رو توی کامپیوتر شخصی خودمون ببینیم احتیاج به یه وب سرور داریم که یکی از معروف ترین هاش IIS نام داره که مربوط به ویندوزه و از اون بهتر و قدرتمند تر Apache هست که در خود لینوکس هم وجود داره و فقط کافیه فعال بشه . از دیگر استفاده های وب سرور ها اینه که تصور کنید در شبکه یه سایت رو می خواییم قرار بدیم که فقط کاربرانی که درون اون شبکه قرار دارن بتونن از اون استفاده کنن و یا حتی در سرور های هاست از این سرویس استفاده می شه . البته این ها که گفتیم فقط برای آشنایی اولیه بود و قطعا استفاده هایی خیلی وسیع تر داره . اما بریم سر اصل موضوع :

باز هم تکرار می کنم که لینوکس ارادت خاصی به فایل داره همین جور که می دونین همه چیز به فایل . پس این سرویس هم یه فایل و جالب این که دیگه احتیاج به تنظیمات خاصی نداره و کار های عادی رو میشه خیله راحت باهاش انجام داد :

اول به این فایل مجوز اجرایی می دیم :

```
chmod +x /etc/rc.d/rc.httpd
```

حالا سرویس مجوز اجرایی داره و میشه اجراش کرد :

```
/etc/rc.d/rc.httpd start
```

برای این که چک کنیم که آیا پراسس این سرویس در حال اجرا هست یا نه ؟ :

```
ps -A|grep httpd
```

خوب دیگه همه چی تمومه و اگه از روی کامپیوتر های شبکه مرور گری رو باز کنیم و توی قسمت آدرس اون IP سرور رو وارد کنیم باید در عبارت زیر نمایش داده بشه :

### it is work

و اما اگه بخواییم فایل هایی که خودمون طراحی کردیم یا مربوط به یه سایت هست رو در سرور قرار بدیم باید اونها رو به شاخه ی زیر انتقال بدیم :

```
/var/www/htdocs
```

راستی فایل تنظیمات پیشرفته این سرویس در شاخه زیر که می تونین یه نگاه بهش بندازین :

```
vi /etc/httpd/httpd.conf
```

## راه اندازی سرور DHCP در لینوکس slackware

DHCP یا (dynamic host configuration protocol) یک پروتکل ارتباطی در شبکه محلی است که به مدیر شبکه امکان اختصاص ip به رایانه ها به صورت خودکار و از طریق یک سرور را فراهم می کند هر دستگاهی که بخواهد به اینترنت متصل شود، یک ip یکتا لازم دارد بنابراین در یک شبکه برای اتصال به اینترنت همه دستگاه ها باید ip داشته باشند.

اگر از dhcp استفاده نشود، باید برای تک تک کامپیوتر ها به صورت دستی ip وارد کرد ولی استفاده از dhcp باعث می شود که اختصاص ip به دستگاه ها به صورت خودکار انجام شود هر کامپیوتر هنگامی که روشن می شود از طریق سیستم عامل یک پیغام درخواست ip برای سرور dhcp می فرستد و dhcp هم یک ip از محدوده ip های مجاز شبکه، به آن کامپیوتر اختصاص می دهد.

شاید ساده ترین سرویس که در لینوکس فعال می کنیم همین سرویس باشد .  
باز هم لینوکس و فایل .

ابتدا با ویرایشگر به مسیر زیر میریم تا تنظیمات لازم رو اونجا انجام بدیم :

```
vi /etc/dhcp.conf
```

وقتی وارد شدیم باید متن زیر رو خط به خط وارد کنیم تا تنظیمات انجام بشه :

```
ddns-update-style none;  
subnet 192.168.2.0 netmask 255.255.255.0  
{  
option routers -. -. -. ;  
option subnet-mask 255.255.255.0 ;  
option domain-name-servers -. -. -. ;  
range 192.168.2.100 162.168.2.200 ;  
}
```

خوب حالا به توضیح برای خطوط بالا بدم .

خط اول باعث می شود که کامپیوتر برای خود DNS اتوماتیک تنظیم نکند .

خط دوم خط دوم IP و netmask سیستم رو مشخص می کنه که میتونه برای هر کس متغیر باشه و البته یاد آوری کنم که این IP مربوط به کارت شبکه ی دوم می شه که به شبکه داخلی ما متصله .

خط سوم برای کامپیوتر های شبکه روتر رو مشخص می کنه که به جای خطوط تیره می تونیم ip سرورمون رو وارد کنیم .

خط چهارم subnet mask رو مشخص میکنه که معمولا همون بالاییه .

خط پنجم DNS رو برای کامپیوتر های داخل شبکه مشخص می کنه که واضحه باید IP ما یا همون IP سرور لینوکس مون باشه .

در خط آخر هم برای کامپیوتر های درون شبکه به محدوده ی IP تعیین می کنیم که باید دقت

کرد با IP کارت شبکه لینوکس سرور از یه کلاس باشن .  
خوب بعد این همه پر گویی من کار تمومه و برای اجرا و فعال کردن از دستور زیر می تونین استفاده کنیند .

```
dhcpcd
```

## راه اندازی سرویس DNS در لینوکس slackware

سرویس DNS رو می خواییم روی سرور لینوکس شبکه ی خودمون فعال و تنظیم کنیم .همین طور که می دونین وقتی از یه کامپیوتر در شبکه یک آدرس رو توی مرور گر می نویسیم این آدرس ارسال میشه به سرور هایی در اینترنت و بعد از مشخص شدن IP مربوط به این آدرس توسط این سرور ها و ارسال اون به مرور گر ، مرور گر می تواند صفحه در خواستی ما مثل [www.google.com](http://www.google.com) رو به راحتی باز کند . چون آدرس کارکتری یا همون URL برای راحتی انسان ارائه شده و در خواست مرور گر ها باید بر اساس IP باشه .

حالا می خواییم با این سرویس این کار بالا رو فقط یک بار انجام بدیم و از آن پس اطلاعات هر آدرس URL و IP مربوط به اون که قبلا استفاده شده در سرور ذخیره میشه و دیگه لازم نیست هر بار به اینترنت جهت این تبدیل آدرس به IP مراجعه بشه و در نتیجه سرعت بهتری رو خواهیم داشت (در واقع همون مفهوم cache).

و شروع کار :

همین طور که قبلا بار ها گفته شد در لینوکس همه چیز به صورت یک فایل قرار داده شده . پس این سرویس هم یک فایل هست که ابتدا باید به این فایل مجوز اجرایی بدیم تا بتونیم اجراش کنیم . پس به سادگی به صورت زیر عمل می کنیم :

```
chmod +x /etc/rc.d/rc.bind
```

حال این سرویس رو باید فعال کرد :

```
/etc/rc.d/rc.bind start
```

حالا برای این که اطمینان حاصل بشه که سرویس اجرا شده یه سر به قسمت پردازش ها در لینوکس می زنیم :

```
ps -A|grep named
```

در دستور بالا پروسس هایی که کلمه ی named داشتند را فیلتر و جدا سازی کردیم . چون این سرویس با این نام در لینوکس اجرا میشه .

اگه تونستیم پراسس رو ببینیم تا حالا همه چیز خوب پیش رفته و ما سرور رو تبدیل به یه DNS کردیم . و فقط باید تنظیمات رو کامل کنیم که برای این کار فایل را در مسیر زیر با ویرایش گر باز می کنیم :

```
vi /etc/resolve.conf
```

حالا در این فایل باید IP کامپیوتر خودمان رو قرار بدیم که حرفه ای تر و بهتر اینه که IP مربوط به local host که 127.0.0.1 هست رو قرار بدیم که اگه موقعی IP کامپیوتر رو تغییر دادیم مشکلی پیش نیاد .

برای اطمینان از اتمام کار فقط کافیسست مرورگر رو باز کنیم و یه آدرس بهش بدیم . اگر بدون مشکل صفحه باز شد ، همه چیز داره درست کار می کنه

## راه اندازی سرویس NAT در لینوکس slackware

اول ببینیم NAT چیه؟!

اگه یه شبکه محلی LAN داشته باشیم و قصد داشته باشیم به اینترنت متصل بشیم امکان داشتن آی پی ولید valid ip به تعداد سیستم ها تقریبا غیر ممکنه. مثلا یه آی پی ولید داریم و میخوایم همه سیستم هامون از اون آی پی استفاده کنن و بدون مشکل به اینترنت وصل بشن. در این سیستم که اینجا لینوکس هست سرویس NAT رو فعال میکنیم تا واسطه ای بین اینترنت و شبکه محلی مون باشه. برای این کار این سیستم باید 2 تا کارت شبکه داشته باشه تا یکی به اینترنت و اون یکی به بقیه سیستم ها وصل بشه. بقیه سیستم ها باید gateway خودشون رو آی پی سرویس NAT یعنی همون لینوکس ست کنن. خب مسلما باید بتونن همدیگرو ping کنن! یعنی ip کارت شبکه NAT که به بقیه سیستم ها وصله و بقیه باید هم رنج باشن. خب حالا فقط میمونه دستورات برای راه اندازی NAT در لینوکس ما. (برای این کار موقع نصب slackware سرویسی با این نام رو برای نصب شدن تیک زده ایم! rc.d\_ipforward ) اگر هم تیک نزده ایم دو راه موقت و دائم هست که دستوراتش به شرح زیر است.

روش موقت:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

روش دائم: اول باید مجوز اجرای به فایل بدیم:

```
chmod +x /etc/rc/rc.ip_forward
```

و بعد استارتش می کنیم:

```
/etc/rc.d/rc.ip_forward start
```

و حالا برای راه اندازی سرویس NAT هم دو راه موقت و دائم هست روش موقت: در اینجا اسم کارت شبکه ای است که اینترنت وصله

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

و روش دائم که به این صورته که دستور بالا رو جایی قرار میدیم که با هربار بالا اومد لینوکس

اجرا بشه و در عمل دائمی میشه. مهم اون جایی هست که دستورات موقع startup اجرا میشه.

```
Echo "iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE" >>  
/etc/rc.d/rc.local
```

مسیر بعد از << مسیر همون فایل هست که دستورات startup رو اجرا میکنه. فقط خدایی نکرده از < به جای << استفاده نکنید که کل فایل رو پاک میکنه و فقط دستور جدید شمارو داخلش قرار میده. یعنی << دستورات جدید رو به فایل اضافه می کنه. بدونه دست زدن به اطلاعات قبلی فایل.

خب حالا سرویس NAT ما آمادهست میتونید از هر یک از سیستم ها یه آدرس اینترنت رو (مثلا ping (4.2.2.4 کنید.

یاشا .. آز یاشا .. آزاد یاشا .. انسان یاشا