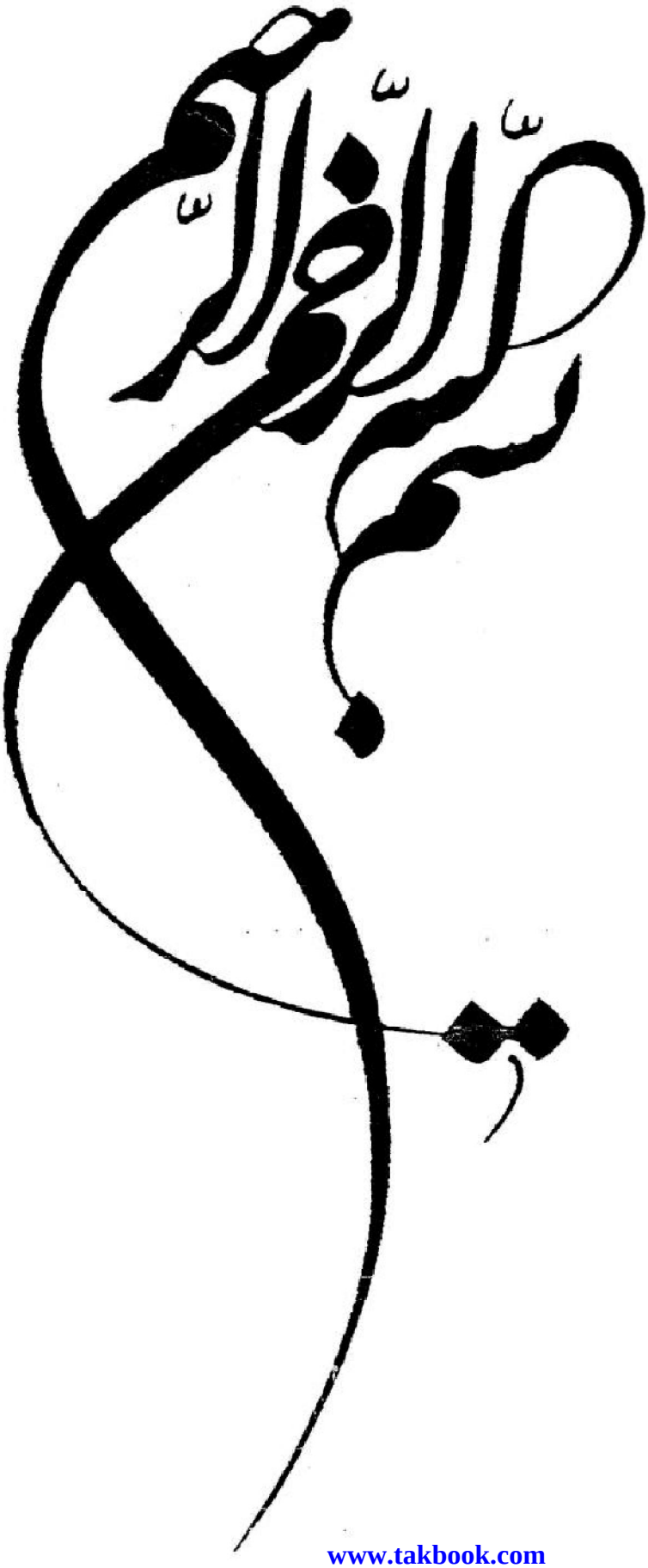
2014

راه اندازی SSH در لینوکس

نویسنده : حسام الدین توحید

SKYWAN13@YAHOO.COM



مقدمه مولف :

آنچه پیش رو دارید به صورت رایگان و تحت لیسانس GNU GPLv3 به علاقه مندان لینوکس هدیه می گردد . در تهیه این مقاله از سر فصل های درسی گفته شده در دوره های LPIc2 و RHCE استفاده شده ولازم می دانم از مهندس **مهدوی فر** به خاطر راهنمایی های مفیدشان و **مرکز آموزشهای پیشرفته دانشگاه شریف (لایتک)** تشکر کافی را داشته باشم. این مطالب با نگاهی کاربردی و بدون پرداختن به بحث های تئوریک گردآوری و عرضه شده است. امیدوارم مطالب ارائه شده بتواند باعث ارتقاء دانش فنی کاربران لینوکس و متخصصین IT شود. این آموزش بر اساس توزیع CentOSمی باشد، لذا خواهشمندم هرگونه نقص در محتوا را به ایمیل نگارنده ارسال فرمائید. انتشار این فایل با ذکر منبع بلامانع است و هر گونه استفاده نامناسب از محتوای ارائه شده بر عهده کاربر بوده و تمام حقوق معنوی این اثر به نویسنده آن تعلق دارد.

زکات علم نشر آن است.

موفق باشید

حسام الدین توحید

خرداد 1393

فهرست مطالب :

آشنایی با پروتکل SSH

مقدمه :

در سال 1995 یک دانشجوی دانشگاه هلسینکی به نام Tatu Ylönen پس از آنکه اطلاعات مهمی مثل رمز و نام های کاربری در شبکه دانشگاه مورد Sniff قرار گرفت به فکر ایجاد یک شبکه امن افتاد که این فکر در نهایت منجر به ایجاد یک Shell امن شد که جایگزینی برای telnet , rlogin , rcp , rsh , ftp شد. البته این دستورات هنوز در لینوکس وجود دارند اما با وجود SSH دیگر کسی از آنها استفاده نمی کند و تمامی اینها با دستورات scp و ssh جایگزین شدند.

SSH مخفف Secure Shell است. SSH یک پروتکل ارتباطی امن بر پایه TCP/IP بین سرویس دهنده و سرویس گیرنده است که با رمز گذاری داده ها از افشای اطلاعات در طول مسیر جلوگیری کرده و یک کانال امن در سیستم عامل سرور برای دستیابی به خط فرمان برای کلایتنت ایجاد می کند.

کلمه Shell ممکن است این تصور را ایجاد کند که SSH یک مفسر فرمان است اما این کاملا اشتباه بوده بلکه یک پروتکل اتباط امن می باشد. الگوی اولیه رمزنگاری در سرویس ssh که در سال 1995 ارائه گردید در زمان خودش ابزاری مناسب محسوب می شد ولی با گذشت زمان محدودیت هایی در استفاده از آن پدیدار گشت که به جهت رفع این محدودیت ها نسخه دوم این سرویس ارائه شد. همواره سعی کنید تا با تنظیم عبارت Protocol در فایل های پیکربندی سیستم، خود را ملزم به استفاده از نسخه دوم این سرویس کنید. استفاده از SSHمحدودیتهایی نظیر لایسنس و پرداخت هزینه را در بردارد ، لذا برای رفع این محدودیت گروه OpenBSD شروع به ارائه موازی نسخه جدیدی به صورت رایگان نمود که نام این محصول را OpenSSH گذاشتند. با خرید لایسنس شرکت Tectia که ارائه کننده SSH تجاری می باشد علاوه بر پشتیبانی ازکاربر ، از صفحه مدیریت تحت وب SSH بهره مند می شوید ولی چنین مزایایی در OpenSSH وجود ندارد. راز محبوبیت پروتکل SSH کدگذاری شبکه ، ایجاد تونل امن و پشتیبانی از انواع متد های دیگر ارتباط امن می باشد.

مزایای استفاده از SSH

(نسبت به شیوه های قدیمی ارتباط از راه دور)

1. رمز گذاری دادها : (Encryption Data)

همانطور که گفته شد نیاز به یک اتصال امن بین سرور و کلاینتها و جلوگیری از Sniff ، مهمترین دلیل استفاده از SSH می باشد.

2. بررسی یکپارچگی دادها (Data Integrity)

این خاصیت برای جلوگیری از حمله های Insertion and Replay Attacks بسیار مفید می باشد. لازم به ذکر است رمزگذاری داده ها بدون استفاده از Session ID نمی تواند از حملات replay attacks جلوگیری کند. از نسخه دو این پروتکل این قابلیت اضافه شد تا پکتها در مسیر ارسال جایگزین و یا شبیه سازی نشوند. در این نوع حمله هکر دیتای تبادلی در نشست را مانیتور نمی کند بلکه بلکه مثل نرم افزارهای Keylogger خروجی صفحه کلید را مانیتور نموده و با مقایسع پکتهای تایپ شده با ترافیک جاری SSH متوجه کارکترهای خاص تایپ شده می شود.

3. قابلیت فشرده سازی : (Compression)

این پروتکل علاوه بر رمز نگاری ، اطلاعات ارسالی را فشرده می کند که این کار در ارتباطات کم سرعت بسیار مفید خواهد بود.

4. عدم اتصال به سرور جعلی : (Prevent Impersonation of host)

در یک انصال SSH هنگام اتصال به سرور، هویت سنجی صورت می گیرد و اگر یک ماشین با مشخصات سرور در مسیر کلاینت قرار گرفته باشد امکان میزبانی کلاینت و یا بالعکس را ندارد. در حالی که در پروتکل های قدیمی تر مثل Telnet این اتفاق اجتناب ناپذیر است. این نوع حمله به حمله مرد میانی موسوم است. (Man-In-The-Middle-Attack or MITM Attack) .

5. لاگ فایل : (Log Access)

SSH امکان فعال و یا غیر فعال شدن فرایند تهیه لاگ فایل ها را دارد با فعال شدن این امکان در مواقع بروز مشکل مدیر سیستم بعد از بروز خطا اولین موردی که برای رفع مشکل بررسی میکند لاگ فایل ها می باشد.

6. امکان استفاده از X11 Applications

SSH این قابلیت را دارد که برنامه های دیگر مثل نرم افزارهای گرلفیکی را کد گذاری کند. به قابلیت Port Forwarding هم می گویند. از این قابلیت برای Tunneling هم استفاده می شود.

7. موجود بودن کامندهای موسوم به r-Command

تمام امکانات کامندهای موسوم به r-Command در SSH وجود دارد. به عنوان مثال از سرور 1 به سرور 2 دستور date را اجرا می کنیم :

```
skywan13@localhost ~]$ ssh userx@x.x.x.x date
: ssh userx@x.x.x.x 's password
Tue Sep 21 18:11:28  IRDT  2014
```

علاوه بر کاربردهای رایج این پروتکل ، انعطاف پذیری بر حسب نیاز موجب محبوبیت این پروتکل در بین کاربران و متخصصان کامپیوتر شده است.

نصب و راه اندازی سرویس Openssh

عموما به شکل پیش فرض سرویس Openssh در زمان نصب سیستم عامل نصب می شود. همچنین از آنجا که ssh و scp جزیی ازیک برنامه هستند، هر دو از یک فایل پیکربندی استفاده کرده و توسط سرویس SSH مدیریت می شوند .بسته های rpm سرویس SSH را به راحتی می توان ازمنابع این نوع بسته ها در Internet تهیه کرد . معتبرترین مرجع جهت تهیه بسته های مربوط به این سرویس سایت ssh.com است که در آن شما قادر خواهید بود نسخه های تجاری و غیرتجاری سرویس SSH را به راحتی تهیه کنید. این سرویس از طریق کامپایل کد منبع آن و ابزار apt-get در سیستم های مبتنی بر debian نیز به راحتی قابل تهیه و استفاده است. همچنین کاربران فدورا با استفاده از yum می توانند این بسته را نصب کنند هرچند به صورت پیش فرض این بسته روی اکثر توزیعات لینوکسی نصب هست. SSH با حروف بزرگ به طور کلی به پروتکل SSH اطلاق میشود و ssh با حروف کوچک به نرم افزر سمت کلاینت گفته می شود که برای اتصال به سرور به کار می رود. پیش نیاز نصب SSH پکیج های zlib و OpenSSL است که در صورت استفاده از Yum این نیازمندیها به صورت اتوماتیک نصب خواهند شد. SSH تحت نظرinit اداره می شود.

ابتدا باید از نصب بودن پکیج Openssh اطمینان حاصل کنیم لذا با دستور زیر از سیستم query می گیریم :

#rpm -qa | grep openssh

در صورت نصب نبودن ، در سیستم های ردهت جهت نصب openssh از yum استفاده می کنیم :

#yum –y install openssh

بعد از نصب ، باید اطمینان حاصل کنیم که آیا پکیج Openssh بر روی سیستم نصب شده است یا خیر لذا با دستور زیر از سیستم query می گیریم :

#rpm -qa | grep openssh

سپس با دستور زیر شاخه ها و مسیرهایی که فایل های این سرویس در آن ایجاد شده است را چک می کنیم :

#rpm -ql openssh

و با این دستور هم اطلاعات لازم را در مورد پکیج این سرویس به دست می آوریم :

#rpm -qi openssh

سپس با دستور chkconfig مشخص می کنیم در چه runlevel هایی فعال باشد :

#chkconfig openssh on

و در انتها سرویس را reset می کنیم :

#service sshd restart

نکته : با نصب openssh چهار پکیج روی سیستم نصب می شوند که این دو پکیج مهمترین آنها هستند :

openssh-server openssh-client

همانطور که گفته شد SSH یک پروتکل کلاینت سروری است و در هر دو طرف یعنی سرور و کلاینت لینوکسی باید پیکر بندیهای لازم صورت پذیرد.

در سمت سرور در مسیر /etc/ssh/ و در سمت کلاینت لینوکسی در مسیر /home/user/.ssh/ باید تغییراتی اعمال شود . از فایلهای مسیر /etc/ssh/ برای پیکربندی سرور SSH استفاده می شود که این تنظیمات global بوده و به همه کلاینتها و یوزرها اعمال می شود. اما تغییراتی که در /home/user/.ssh/ انجام می دهیم برای تحت تاثیر قرار دادن عملکرد کامندهای ssh و scp است و فقط در سیستم کلاینت اعمال شده و local می باشد.

در این مقاله ابتدا تنظیمات سمت سرور را مرور کرده سپس به تنظیمات سمت کلاینت خواهیم پرداخت.

بررسی فایل های موجود در /etc/ssh

در زیر این دایرکتوری فایل های پیکربندی و الگوریتم های این پروتکل قرار دارد که مهمترین آنها در زیر توضیح داده شده است :

sshd _config
ssh _config
ssh _host_key
ssh _host_dsa_key
ssh _host_rsa_key
Moduli

sshd_config : این فایل مهمترین فایل این دایرکتوری است . با این فایل سرویس SSH را پیکربندی می کنیم. تنظیمات این فایل global بوده و همه یوزرها اعمال می شود.

ssh_config : با این فایل کلاینتهای ssh مورد پیکربندی قرار می گیرند. اگر در این فایل تغییراتی اعمال کنیم به همه یوزرها اعمال می شود.

ssh _host_key : این فایل کلید SSH ورژن یک می باشدکه از الگوریتم خاصی برای رمز نگاری استفاده نمی کند.

ssh _host_dsa_key
ssh _host_rsa_key

این فایل ها کلید SSH ورژن 2 می باشند که از الگوریتمهای rsa و dsa برای رمز نگاری استفاده می کند.

Moduli : اطلاعات dh که در معاوضه کلیدها بین طرفین اسفاده می شود در این فایل قرار می گیرد.

پیکربندی سرویس SSH

در این قسمت به تشریح بعضی از قسمتهای فایل پیکربندی سرور SSH می پردازیم. برای اطلاع دقیق از تمامی آپشن های این سرویس به man sshd_conf رجوع کنید :

Port 445

پورت پیش فرض سرویس SSH پورت tcp 22 می باشد. SSH با این پورت روی تمام کارت های شبکه به حالت listen میرود. هر گاه زمانی احساس کردید افرادی قصد نفوذ به سیستم شما، از طریق پورت شناخته شده ای مثل 22 را دارند، می توانید با تغییر آن ، به پورتی که تداخلی با برنامه های کاربردی موجو د در سیستم ندارد، از این امر پیشگیری کنید. این کار را می توان تنها یک پیشگیری اولیه محسوب کرد، زیرا برنامه هایی در شبکه جهت تشخیص پورت هایی که هم اکنون در حال اجرای سرویس ssh هستند، نیز وجود دارند. با دستور netstat از غیرقابل استفاده بودن آن توسط سایر برنامه های سیستم اطمینان حاصل کنید . # netstat -an| grep 435

AddressFamily any

این خط می تواند سه مقدار داشته باشد. مقدار این خط مشخص کننده این است که از چه ورژن IP پشتیبانی کند.

مقادیر این خط :

inet : مشخص کننده IP ورژن 4 است.

int6 : مشخص کننده IP ورژن 6 است.

any : یعنی از هر دو ورژن IP پشتیبانی کند.

LitenAddress 192.168.1.1

Ssh به طور پیشفرض با تمام IPها ارتباط برقرار می کند و اجازه ارتباط می دهد در اینجا میتوانیم کارت شبکه خاصی را به عنوان کارت اختصاصی SSH آدرس دهی کنیم. این خط مربوط به IP ورژن 4 است.

ListenAddress : :

در IP ورژن 6 مقدار این خط معنی همه را می دهد.

HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key

با این آپشن ها مسیر کلید های rsa و dsa را معین می کنیم که البته نیازی به تعویض این مسیرها نیست.

KeyRegenerationInterval 1h

سرور SSH به طور پیش فرض هر یک ساعت کلیدهای تولید شده روی سرور را تولید مجدد می کند تا از حملات Capture data و یافتن الگوریتم جلوگیری کند.

ServerkeyBits 2048

طول کلیدهای رمزنگاری تولید شده به طور پیش فرض 1024 بیت است که برای امنیت بیشتر بهتر است بر روی 2048 تنظیم شود.

SyslogFacility AUTHPRIV

این خط مشخص میکند log هر کاربری که با ssh به سرور لاگین کرده و یوزر پسورد وارد کند ثبت شود.

LogLevel INFO

این خط می تواند دو مقدار INFO و DEBUG داشته باشد که سطوح log برداری را مشخص می کنند.

PermitRootLogin no

به صورت پش فرض یوزر root اجازه Remote access از طریق ssh را دارد که باید این اجازه از آن گرفته شود. با قراردادن no این اجازه را از آن می گیریم.

LoginGreaceTime 2m

این زمان مشخص می کندکلاینت که session برقرار کرده و صفحه لاگین را در اختیار دارد 120 ثانیه وقت دارد یوزر پسورد را تایپ و به سرور لاگین نماید اگر در این مدت اقدامی صورت ندهد کانکشن آن قطع شده و صفحه لاگین بسته می شود.

MAXAuthTrise 6

اگر به صورت پیش فرض یوزری 6 مرتبه پسورد را اشتباه وارد کند ، ssh صفحه لاگین را از او گرفته کانکشن را قطع می نماید.

Allow and Deny
Allow Users
Deny Users
Allow Groups
Deny Groups

در حالت پیش فرض تمامی یوزرها می توانند با استفاده از یوزرشان در سرور به آن ssh زده ولاگین کنند. اگر بخواهیم به یوزر یا گروهی اجازه دسترسی به ssh را داده یا دسترسی آنها را محدود کنیم باید در جلوی این آپشن ها یا نام یوزرها و گروه ها را جدا از هم وارد کرده و یا در لیستی جداگانه وارد و در اینجا آدرس فایل مربوطه را وارد کنیم .

AuthorizedkeysFiles .ssh/authorized_keys

این فایل جهت احراض هوی keybase مورد استفاده قرار می گیرد که بر روی دایرکتوری Home هر یوزر لینوکس قرار دارد. این خط مسیر این فایل را مشخص می کند.

PasswordAuthentication yes
PubkeyAuthentication yes

در SSH دو نوع احراض هویت وجود دارد ، یمی بر اساس Password و دیگر ی بر اساس Keybase می باشد. احراض هویت پیش فرض از نوع پسورد است. خط اول احراض هویت را بر اساس پسورد و خط دوم احراض هویت را بر اساس کلید فعال می کند.

PermitEmptyPassword no

بعضی از اکانتها دارای پسورد نیستند برای جلوگیری از لاگین شئن چنین اکانت هایی مقدار این خط را برابر no قرار می دهیم.

UsePam yes or no

سرویس SSH می تواند توسط دو مکانیزم امنیتی Pam و tcp_wrapper کنترل می شود. فعال کردن این گزینه باعث می شود کنترل امنیتی این سرویس تحت اختیار Pam قرار می گیرد که در این صورت باید فایل زیر را مورد پیکربندی قرار دهیم. /etc/security/access.conf و اگر بخواهیم با مکانیزم tcp_wrapper مورد کنترل قرار بگیرد باید فایل /etc/host.allow را کانفیگ کنیم.

جهت کار کردن با این دو فایل حتما باید راهنمای آنها را مورد مطالعه قرار دهیم. اگر بخواهیم بفهمیم کدام سرویس توسط tcp_wrapper کنترل می شود از این دستورات استفاده می کنیم.

#whereis sshd
#ldd /usr/sbin/sshd

دستور ldd تمامی ماژول هائی که یک سرویس از آن استفاده می کند را نشان می دهد.خروجی این دستور لیست تمام ماژل های sshd را نشان می دهد. سومین ماژول نشان داده شده در خروجی ، ماژول libwarrp.so است. هر سرویسی که این ماژول را داشته باشد یعنی توسط tcp_wrapper کنترل میشود.

Accept ENV

این عبارت متغیرهایی را نشان می دهد که سرور ssh اجازه Export کردن آنها بر روی کلاینت را دارد.

X11Forwarding yes

این امکان را به ما می دهد تا بعد از اتصال به سرور SSH به صورت گرافیکی بتوانیم سرور را پیکربندی کنیم. بعد از اتصال ، ما با یک صفحه مشکی رنگ مواجه می شویم ، اما می توانیم از دستوراتی استفاده کنیم که دارای یک محیط گرافیکی هستند مثل دستورات setup , rconf , system-config-network . این دستورات کنسول گرافیکی خاصی را بر روی سرور اجرا می کنند. بهتر است این گزینه بر روی no تنظیم شود.

Banner /etc/ssh/banner

می توانیم پیامی طراحی کرده و در مسیر گفته شده قرار دهیم تا در موقع اتصال هر یوزر به SSH این بنر به نمایش در بیاید.

Printmotd yes

اگر این خط برابر yes باشد می توان پیامی طراحی و در مسیر /etc/motd قرار داد تا در موقع اتصال آن را نمایش دهد.

Protocol 2

Ssh دارای دو ورژن 1 و 2 می باشد ورژن 1 دارای آسیب پذیری های mitmاست و نباید به هیچ عنوان از آن استفاده شود.

ClientAliveInterval 600
ClientAliveCountMax 0

می توان یک حالت Idle TimeOut برای یوزرها ایجاد کرد تا اگر بین کار آنها فاصله ای بیافتد و عملیاتی صورت نگیرد باعث Logout در یوزر شودبا اضافه کردن این دو خط و مقادیر مورد نظر این امکان فعال میشود.

HostBasedAuthentication no

تا وقتی که این ویژگی فعال باشد یک یوزر از هاست خودش می اوند به هاست دیگری در شبکه هم لاگین کند .

IgnoreRhosts yes

فعال بودن این خط باعث می شود یوزرها نتوانند به فایل های shosts. و rhosts. دسترسی داشته باشند.

نکته : در حالت پیش فرض یوزرها امکان دسترسی به دایرکتوری هایی به جز دایرکتوری اصلی خود را دارندو می توانند به دایرکتوری هایی مانند bin,etc,..... دسترسی پیدا کنند. با استفاده از سیستم عامل هایی که بر پایه chroot هستند و یا ابزاری مانند rssh می توان ssh را در برابر دیگر یوزرها ایمن کرد.

استفاده از ssh جهت اتصال به یک کامپیوتر در شبکه

استفاده از ssh بسیار شبیه telnet است. جهت اتصال به یک ماشین دیگر در شبکه یا یک سرور SSH تحت یک کاربر دلخواه از آن ماشین از سوئیچ l- استفاده می کنیم. در اینجا چند مثال جهت اتصال به یک سرویس دهنده SSH ارائه شده که به بررسی آنها می پردازیم :

با استفاده از این فرمان تحت کاربر root به کامپیوتر 192.168.1.1 در شبکه متصل می شویم.

```
# ssh   192.168.1.1
```

حال اگر بخواهیم تحت یک کاربر دیگر این اتصال را انجام دهیم، از یکی از قالب های زیر می توانیم استفاده کنیم :

```
# ssh   -l   ali 192.168.1.1
# ssh   ali@192.168.1.1
```

و اگر بخواهیم برروی پورتی غیر از پورت 22 عملیات Login را انجام دهیم، دستور فوق بدین شکل تغییر پیدا می کند:

```
# ssh   -P 435 ali@192.168.1.1
```

و در صورتی که مایل باشیم ، دستوراتی مانند system-config-network که محیط گرافیکی دارند را اجرا کنیم از سوئیچ X– استفاده می کنیم :

```
# ssh   -X   ali@192.168.1.1
```

در اولین مرتبه ای که از طریق کلاینت لینوکسی با سروری ارتباط ssh برقرار می کنیم یک هشدار دریافت خواهیم کرد، دلیل آن هم این است که سرور مقابل می خواهد بر روی سیستم ما یک کلید آپلود کند.
یکی از امکانات جالب ssh قابلیت ورود و اجرای فرامین منفرد در یکی از سیستم های شبکه است. برای این کار کافی است فرمان مورد نظر را در یک جفت کوتیشن، در انتهای فرمان ssh قرار دهیم. در مثال زیر یک کاربر قصد دارد به نسخه کرنل موجود برروی سرویس دهنده 192.168.1.1 پی ببرد که برای این کار فرمان uname -a را بر روی سرویس دهنده اجرا می کند و بلافاصله خروجی آن به نمایش در می آید :

```
# ssh   ali@192.168.1.1 "uname -a"
Linux yadi 2.6.8-1.521 #1 Mon Aug 10:10:17 EDT 2004 i686 i686 i386
```

مبحث ssh Client

در این قسمت از آموزش ، دیگر با سرور SSH کاری نداریم بلکه می خواهیم تنظیمات ssh client را بررسی کنیم. زمانی که برای اولین بار از طریق ssh به یک سرور یا ماشینی متصل می شویم، پیامی مبنی بر اینکه ماشین ما توسط سیستم مقصد شناخته شده نیست را دریافت خواهیم کرد. در همین زمان درخواستی جهت ذخیره سازی یک نسخه از کلیدهای شناسایی، ssh سرور مقصد برروی کامپیوتر خودمان دریافت می کنیم که با تائیدآن یک RSA key fingerprint که همان کلید Public سرور مقصد است بر روی سیستم Local آپلود می شود. این رویه را در زیر می توانید مشاهده کنید :

روال ذخیره سا زی کلید:

```
# ssh   192.168.1.1
The authenticaty of host 192.168.1.1 (192.168.1.1)' can't be established
RSA key fingerprint is 5d:d2:f5:21:fa:07:64:0d:63:1b:3b:ee:a6:58:58:bb
Are you sure you want to continue connecting (yes/no)? Yes
Warning : Permanently added '192.168.1.1' (RSA) to the list of known hosts.
root@192.168.1.1' password
Last login: The Nov 13 11:17:36 2014 from 192.168.1.1
No mail
```

این کلید در دایرکتوری home کاربر در پوشه ای به نام ssh. که یک پوشه مخفی است ذخیره می شود. در این پوشه فایلی به نام known_host وجود دارد که حاوی کلید های Public سرورهایی است که ما به آنها کانکشن زده ایم.

ssh سیستم local با public key دریافتی از سرور و private Key خودش اطلاعات را رمزنگاری کرده و به سمت سرور ارسال می کند. pubkey سرور مثل قفل میباشد که کلید آن privkey سرور است و با آن قفل دیتا رمز شده را باز میکند.

اگر سیستم عامل سرور یا سرویس ssh را بر روی سرور (server ssh) مجددا نصب کنیم و یا ip سرور را تغییر دهیم، کلیدهای pub تولید شده در سمت سرور با کلیدهای ذخیره شده در known_hosts کامپیوترهای سرویس گیرنده تطابق نخواهند داشت و از این رو ارتباط ssh بین کلاینتها و سرور برقرار نشده و کاربر پیغام خطایی مانند زیر دریافت می کند که در آن احتمال بروز حمله از طریق هکرها هشدار داده می شود. علت اصلی این پیام ، عدم همخوانی کلید pub سرور با کلید pub موجود بر روی کلاینت است.

پیغام های خطای سرویس دهندهssh

```
#ssh   ali@192.168.1.1
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@ WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY
Someone could be eavesdropping on you right now (man-in-the-middle-attack)
It is also possible that the RSA host key has just been changed
The fingerprint for the RSA key sent by the remote host is
d:d2:f5:21:fa:07:64:0d:63:1b:3b:ee:a6:58:bb5
Please contact your system administrator
Add correct host key in /root/.ssh/known_hosts:2
RSA host key for 192.168.1.10 has changed and you have requested strict checking
Host key verification failed
```

اگر اطمینان دارید که بروز پیام به دلیل نصب مجدد سرویس یا سیستم عامل سرور ویا تغییر ip سرور است، کافی است :known_host را ویرایش کرده و خطوط مربوط به کلید قبلی سرویس دهنده ssh را از آن حذف کنیم. پس از این کار با اتصال مجدد به سرویس دهنده ssh مجددا پیغامی مبنی بر ذخیره سازی کلید جدید در فایل ssh/known_hosts./~ دریافت می کنید و از این پس جلسات کاری مربوط به سرویس ssh بدون مشکل انجام خواهد شد چون عدم همخوانی کلیدها با دانلود کلید جدید و ذخیره آن در فایل known_host بر طرف گردیده است. به ازای ارتباط با هر سرور یک محتوای کلید pub در فایل known_host ذخیره می شود.

اگر بخواهیم دو کلید pub را با هم مقایسه کنیم یکی از آنها را از درو ن فایل known_host به یک فایل جدید منتقل کرده سپس توسط دستور diff آن دو را با هم مقایسه می کنیم.

```
# diff   file1   file2
```

نصب و راه اندازی ssh Client

همانطور که گفته شد ssh یک نرم افزار انحصاری است که بابت قابلیت های آن باید لایسنس خریداری کرد. اما بسته openssh رایگان بوده و محدودیت های ssh را ندارد. لذا جهت استفاده از ssh در کلاینتهای لینوکسی هم از openssh استفاده می شود.

ابتدا باید از نصب بودن پکیج openssh-clients اطمینان حاصل کنیم لذا با دستور زیر از سیستم query می گیریم :

```
#rpm  -qa | grep openssh-clients
```

در صورت نصب نبودن ، در سیستم های ردهت جهت نصب openssh-clients از yum استفاده می کنیم :

```
#yum –y   install openssh-clients
```

بعد از نصب ، باید اطمینان حاصل کنیم گه آیا پکیج Openssh بر روی سیستم نصب شده است یا خیر لذا با دستور زیر از سیستم query می گیریم :

```
#rpm   -qa | grep openssh
```

سپس با دستور زیر شاخه ها و مسیرهایی که فایل های این سرویس در آن ایجاد شده است را چک می کنیم :

```
#rpm   -ql   openssh-clients
```

و با این دستور هم اطلاعات لازم را در مورد پکیج این سرویس به دست می آوریم :

```
#rpm   -qi   openssh-clients
```

سپس با دستور chkconfig مشخص می کنیم در چه runlevel هایی فعال باشد :

```
#chkconfig   sshd   on
```

کانفیگ سرور SSH زمانی سودمند است که کلاینتها بخواهند به آن کانکشن بزنند اما کلاینت را برای این تنظیم میکنیم تا بتواند از سروری که تنظیمات آن پیش فرض نیست استفاده کند. فایل پیکربندی ssh Client در مسیر /etc/ssh/ssh_config قرار دارد. و در انتها با دستور زیر چک میکنیم ssh روی پورت 22 و یا هر پورت دیگری به حالت listen رفته باشد.

```
#netstat   -ntpl   |   grep   22
```

استفاده از ارتباط بدون پسورد در ssh Client

در مواردی برای اجرای مجموعه ای از دستورالعمل های متوالی در قالب یک اسکریپت لازم است که ، امکان کپی کردن فایل ها از طریق SCP را بدون وارد کردن کلمه عبور داشته باشیم. یا جهت مصون ماندن از خطر لو رفتن پسورد توسط نرم افزارهای مخرب نخواهیم پسوردی جهت احراض هویت وارد کنیم. به این نوع از احراض هویت Key Base Authentication گفته می شود.

در زمان استفاده از این امکان در سرویس ssh لازم نیست هیچ گونه نگرانی برای فاش شدن کلمه عبور و یا از کار افتادن اسکریپت مورد نظرمان با تغییر کلمه عبور داشته باشیم. به راحتی جهت انجام این کار می توان سرویس SSH را پیکربندی کرد تنها باید کلید این نوع از ارتباط را به سرور معرفی کنیم.

بدین وسیله سرویس دهنده ها قادر خواهند بود به کمک این کلیدهای از پیش نصب شده یکدیگر را تایید کرده، به تبادل دیتا بپردازند. ریسک امنیتی که در این روش وجود دارد این است که امکان دسترسی به یک حساب کاربری برروی سرورتنها از طریق وارد کردن نام کاربری صورت می گیرد که برای کم کردن خطر احتمالی این کار باید از حساب های کاربری غیر مدیریتی در سرور استفاده کنیم، تا در صورت فاش شدن حساب کاربری مربوط به SSH ، امکان اعمال نفوذ در کارهای مدیریتی سیستم میسر نباشد.

در روش اول کلاینت ، pubkey سرور را دریافت و اطلاعات را با pubkey سرور و privkey خودش رمز کرده و برای سرور ارسال می کند.

اما در این روش عکس این عمل را انجام می دهد. یعنی کلاینت یک pubkey و یک privkey تولید کرده و pubkey را بر روی سرور آپلود میکند تا تمام کارها به عهده خودش باشد. در این شیوه دیگر نیازی نیست برای احراض هویت پسورد وارد کنیم بلکه پسورد ما کلیدی است که بر روی سرور قرار داده دایم. در زیر شیوه این کار به طور کامل توضیح داده شده است .

پیکربندی ssh Client و تولید کلید

در اینجا به بررسی مراحلی که جهت تبدیل کردن یک کامپیوتر به سرویس گیرنده SSH، بدون درخواست کلمه عبور انجام شود می پردازیم دستورات و مراحلی که در ادامه شرح داده می شود بر روی سیستم کلاینت اعمال می گردد. در ssh برای تولید کلید از دو الگوریتم rsa و dsa استفاده می شود. از الگوریتم dsa برای امضاء دیجیتال استفاده می شود اما rsa هم برای امضاء دیجیتال و هم برای رمزنگاری کاربرد دارد. dsa سریع تر بوده ، ولی امنیت کم تری دارد ولی rsa کندتر بوده و نسبت به dsa از امنیت بیشتری برخوردار می باشد. الگوریتم پیش فرض برای تولید کلید ، الگوریتم rsa است .

1. ابتدا در کلاینت لینوکس یک جفت کلید رمزنگاری SSH که همان کلیدهایpub و priv هستند را برای حساب کاری که قرار است از آن جهت کپی کردن فایل ها استفاده شود، ایجاد می کنیم. این کار توسط فرمان ssh-keygen صورت می گیرد که نحوه انجام آن در زیر نشان داده شده است. دقت کنید زمانی که درخواست وارد کردن یک کلمه عبور از شما می شود تنها کلید Enter را فشار دهید، و هیچ کلمه ای را وارد نکنید ، البته با سوئیچ p– می توان پسورد آن را بعد از تولید کلید عوض کرد. دقت کنید در هر مسیری که باشید کلید تولید شده در همانجا ذخیره می شود، توصیه می شود کلیدها را در پوشه ssh. ذخیره کنید :

```
#cd   /home/skywan13/.ssh
#ssh-keygen   –b 2048
# ssh-keygen
Generating public/private dsa key pair
Enter file in which to save the key:(root/.ssh/id_dsa/)
Enter passphrase (emty for no passphrase):
Enter same passphrase again:
Your identification has been savd in.root/  .ssh.id_rsa
Your public key has been saved in.root/ .ssh/id_rsa.pub/
The key fingerprint is: e:73:59:83:96:93:4a:50:33:aa1
```

2. بعد از اتمام مراحل کار فایل های مربوط به کلیدهای رمزنگاری ساخته شده در پوشه ssh. از شاخه خانگی کاربر جاری ذخیره می شوند. فایل id_dsa.pub به عنوان کلید عمومی است که با سرویس دهنده مقصد به اشتراک گذاشته می شوند. و فایل id_rsa به عنوان کلید priv مورد استفاده قرار می گیرد. بعد از اتمام مراحل تولید کلید باید به سرور

مورد نظر ، کلید pub ساخته شده را معرفی کنیم. این انتقال باید به صورت امن صورت پذیرد لذا با دستور scp آن را به سرور مورد نظر منتقل می کنیم :

```
#cd /home/skywan13/.ssh
#scp    id_rsa.pub   h.tohid@192.168.1.1:/home/h.tohid/.ssh
```

3. حال باید به سرور مقصد بفهمانیم چه طور از این کلید کپی شده استفاده کند. چون سرور هنوز متوجه نمی شود که فایل کپی شده ، pubkey ما می باشد. لذا برای مطلع کردن سرور باید محتوای فایل id_rsa.pub را داخل فایل جدیدی به نام authorized_keys قرار دهیم. سرور کلیدهای pub یوزرها را فقط از این فایل می خواند.

```
#cat    id_rsa.pub   >   authorized_keys
```

سپس باید در سرور ، وارد فایل پیکربندی SSH شده و دوخط زیر را تغییر دهیم. اگر هر دو خطوط yes باشند یعنی یوزر هم از طریق کلید و هم از طریق پسورد اجازه احراض هویت دارد. روش ارتباط با کلید امن تر می باشد و بهتر است خط پسورد مقدار no داشته باشد.

```
#vi   /etc/ssh/sshd_config
PubkeyAuthentication   yes
Password Authentication   no
RSAAuthentication   yes
#service sshd restart
```

نکته مهم : می توانیم برای کپی به جای استفاده از scp از دستور زیر برای این کار استفاده کنیم که بسیار بهتر است ، زیرا زمانی که عمل کپی را انجام می دهد، در سرور فایل کپی شده را با نام authorized_keys ذخیره کرده و فایل را در دایرکتوری یوزر مورد نظر و در پوشه /.ssh قرار می دهد ، لذا دیگر نیازی نیست که با ssh به سرور مورد نظر وصل شده و محتویات فایل id_rsa.pub را درون فایل authorized_keys قرار دهیم.

```
#ssh-copy-id   -i   ~/.ssh/id_rsa.pub   h.tohid@192.168.1.1
```

می توانیم برای ایجاد امنیت بیشتر در زمانی که کلید را می سازیم یک پسورد اختصاصی به آن بدهیم تا احراض هویت کاربر با دو لایه امنیت صورت پذیرد.. اگر هر دو آپشن گفته شده در فایل کانفیگ yes باشند ، 3 مرتبه اول پسورد یوزر را می پرسد و اگر اشتباه تایپ شود برای احراض هویت پسورد کلید را می پرسد.

با اتمام این کار، سرور از بابت هر کانکشنی که بر مبنای ssh کار کند از یوزر مربوطه پسورد نمی خواهد زیرا احراض هویت بر اساس کلید انجام میشود.

در زیر مروری دوباره ای خواهیم داشت بر دستورات تایپ شده در ssh Client :

Local System:

```
#cd  /home/skywan13/.ssh
#ssh-keygen  -t  rsa  -b  2048
#scp  id_rsa.pub  h.tohid@192.168.1.1:/home/h.tohid/.ssh
#ssh  h.tohid@192.168.1.1
or
#ssh-copy-id  -i  ~/.ssh/id_rsa.pub  h.tohid@192.168.1.1
```

Remote System :

```
#cd  /home/h.tohid/.ssh
#cat  id_rsa.pub  >  authorized_keys
#vi  /etc/ssh/sshd_config
PubkeyAuthentication  yes
Password Authentication  no
RSAAuthentication  yes
#service sshd restart
```

نکات کاربردی :

1. با دستور زیر می توانید پسورد کلید را بر روی سیستم کلاینت Cash کرد :

```
#ssh-add
```

2. این دستور نشان می دهد بر روی سیستم کلاینت چه کلیدهایی موجود است :

```
#ssh-add  -l
```

3. و این دستور Cash را پاک میکند :

```
#ssh-add  -d
```

4. در صورت برقرار نشدن ارتباط ssh با سرور از دستور زیر برای رفع مشکل استفاده می کنیم :

```
#ssh  -vv  h.tohid@192.168.1.1
```

5. کلاینتها می توانند بدون OpenSSH server هم کار کنند. اگر نیازی ندارند به اینکه کلاینتی به آنها remote login بزند و فقط می خواهید یک طرف به منبعی وصل شوید می توانید openssh server را از روی سیستم پاک کنید :

```
#chkconfig sshd off
#yum erase openssh-server
#netstat -ntlp | grep 22
```

منابع :

مطالب متفرقه منتشر شده در اینترنت
سر فصل دوره های RHCE و LPIc2
راهنما Openssh
سایت centos .org