nixCraft
Linux Tips, Hacks, Tutorials, And Ideas In Blog Format

SSH Public Key Based Authentication on a Linux/Unix server

last updated January 3, 2018 **in CentOS, Debian Linux, FreeBSD, Gentoo Linux, Howto, Linux, Linux desktop, Linux laptop, OpenBSD, RedHat/Fedora Linux, Solaris, Suse Linux, Sys admin, Tips, Ubuntu Linux, UNIX**

The SSH protocol recommended a method for remote login and remote file transfer which provides confidentiality and security for data exchanged between two server systems. The SSH depends upon the use of public key cryptography. The OpenSSH server offers this kind of setup under Linux or Unix-like system. This how-to covers generating and using ssh public keys for automated usage such as:

1. Automated Login using the shell scripts
2. Making backups
3. Run commands from the shell prompt and more
4. Login without password

How to configure SSH Public key-based authentication for a Linux/Unix

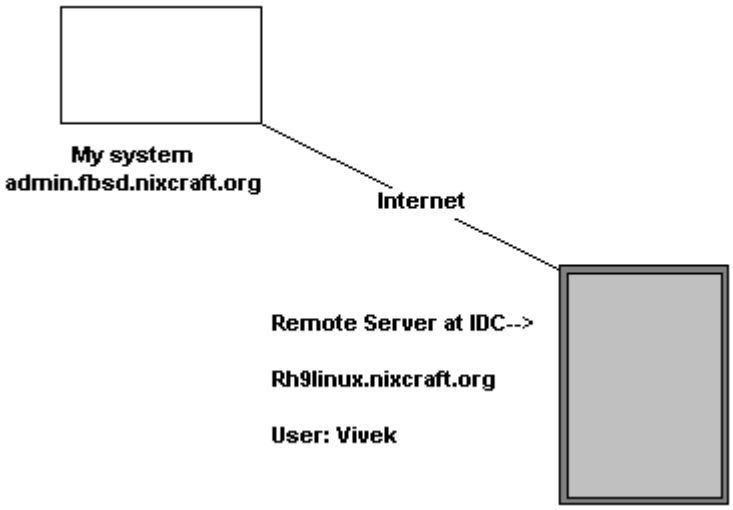
The steps and commands are as follows:

1. On your local system type: `ssh-keygen`
2. Install public key into remote server: `ssh-copy-id user@remote-server-ip-name`
3. Use ssh for password less login: `ssh user@remote-server-ip-name`

Let us see all commands in details.

Generating SSH Keys

First, log on to your workstation. For example, log on to workstation called admin.fbsd.nixcraft.org as vivek user. Please refer the following sample setup. You will be logged in, on your local system, AS THE USER you wish to make passwordless ssh connections.



ssh key based authentication

To create the cryptographic keys on **your local system** powered by FreeBSD/Linux/macOS/ UNIX workstation, enter:

```
ssh-keygen -t rsa
```

Assign the pass phrase (press [enter] key twice if you don't want a passphrase). It will create 2 files in ~/.ssh directory as follows:

- ~/.ssh/id_rsa : identification (private) key
- ~/.ssh/id_rsa.pub : public key

How to copy a public ley (~/.ssh/id_rsa.pub) to your server

Use the scp command to copy the id_rsa.pub (public key) from your local system to *rh9linux.nixcraft.org* remote server as authorized_keys file, this is know as, "installing the public key to server":

```
scp ~/.ssh/id_rsa.pub vivek@rh9linux.nixcraft.org:~/.ssh/authorized_keys
```

Another option is to use the ssh-copy-id command as follows from your local workstation:

```
ssh-copy-id user@remote-box
ssh-copy-id -i ~/.ssh/id_rsa.pub vivek@rh9linux.nixcraft.org
```

How to login to your remote server using SSH keys

From your local system (e.g. FreeBSD/macOS/Linux/Unix workstation) type the following command:

```
ssh user@remote-box
ssh vivek@rh9linux.nixcraft.org
```

Changing the pass-phrase on workstation

To change a passphrase for your ssh keys, use the ssh-keygen command as follows:

```
ssh-keygen -p
```

OR

```
cd ~/.ssh/
ssh-keygen -f id_rsa -p
```

How to use ssh-agen command

You can use the ssh-agent command to avoid continues passphrase typing at the CLI:

```
ssh-agent $SHELL
ssh-add
```

Now ssh server will not use prompt for the password. Above two commands can be added to your ~/.bash_profile file so that as soon as you login into workstation you can set the agent.

Deleting the keys hold by ssh-agent

To list keys, enter:

```
ssh-add -l
```

To delete all keys, enter:

```
ssh-add -D
```

To remove specific key, enter:

```
ssh-add -d key
```

See also:

- keychain: Set Up Secure Passwordless SSH Access For Backup Scripts
- sshpass: Login To SSH Server / Provide SSH Password Using A Shell Script
- How To Setup SSH Keys on a Linux / Unix System
- How to upload ssh public key to as authorized_key using Ansible DevOPS tool
- Man pages: sshd(8),ssh(1),ssh-add(1),ssh-agent(1)

Posted by: Vivek Gite

The author is the creator of nixCraft and a seasoned sysadmin, DevOps engineer, and a trainer for the Linux operating system/Unix shell scripting. Get the **latest tutorials on SysAdmin, Linux/Unix and open source topics via RSS/XML feed** or weekly email newsletter.

---

40 comment

RWP June 13, 2002 at 9:59 am

thank u kind sir.

Ashish May 25, 2004 at 4:56 am

Hi Vivek Sir,

This is Ashish here. Yes... I caught you..
It very nice to see you once again. Where are you right now?? In india?? Wanted to meet you..
Now we are expecting some good technical documents from you as usual. Just now finished LLST written by you.

If possible please mail me at ashish_r_pathak@yahoo.com

Thanks and Regards,
/Ashish Pathak.
Pune, India.

Kevin July 12, 2004 at 12:57 am

Hi Vivek,

I am kevin here from mumbai. Thanks for this article on SSH. Also i liked your Shell programming tutorial. If possible can you give me some examples on Local and remote port forwarding techniques on SSH.

Best regards,
Kevin

Anonymous October 3, 2004 at 11:25 pm

Hi Vivek

this is Amit Shiknis here from Pune. Where are you now?How are you? i just gone through SSH docs its really very nice article.

Hope you will be fine. if possible mail me on amitshiknis@vsnl.net

regards
Amit

Vivek August 3, 2005 at 12:04 am

Kevin,
See url
for Local and remote port forwarding techniques on SSH:

GV May 2, 2007 at 3:15 pm

Hello,

I installed openSSH client on windows.

Create a public key using the command

ssh-keygen -t rsa

Copied the key to the unix box using the command

scp .ssh/id_rsa.pub user@hostname:.ssh/authorized_keys

changed the permissions on authorized_keys
chmod 600

The USERNAME on the windows and unix box are the same.

When I try to run the remote script using ssh

ssh user@hostname scriptname

It Prompts me for the PASSWORD. I am not sure what am I doing wrong here. Any help on this is much appreciated.

Thanks
GV

I am including the client side trace when I used ssh below.

C:Documents and Settingsgvarada.ssh>ssh -v stlap08d whoami
OpenSSH_3.8.1p1, OpenSSL 0.9.7d 17 Mar 2004
debug1: Reading configuration data /etc/ssh_config
debug1: Connecting to stlap08d [172.19.1.24] port 22.
debug1: Connection established.
debug1: identity file /home/gvarada/.ssh/identity type -1
debug1: identity file /home/gvarada/.ssh/id_rsa type 1
debug1: identity file /home/gvarada/.ssh/id_dsa type -1
debug1: Remote protocol version 2.0, remote software version OpenSSH_4.1
debug1: match: OpenSSH_4.1 pat OpenSSH*
debug1: Enabling compatibility mode for protocol 2.0
debug1: Local version string SSH-2.0-OpenSSH_3.8.1p1
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
debug1: kex: server->client aes128-cbc hmac-md5 none
debug1: kex: client->server aes128-cbc hmac-md5 none
debug1: SSH2_MSG_KEX_DH_GEX_REQUEST(1024

Jon May 24, 2007 at 10:00 am

Your method of not giving a pass phrase is *convenient* but not really *secure (IMHO). Much better to create a key with a pass phrase, and use*

```
ssh-add
```

to enter the pass phrase ONCE PER SESSION. That is, before you ssh to the remote machine, run

```
ssh-add
```

which will prompt you for the passphrase. For the rest of the session, ssh-add will authenticate for future ssh connections, which are in effect 'password-less'

Jon May 24, 2007 at 10:02 am

to the moderator: I withdraw my previous comment, the post does include ssh-add, but I had not read it thoroughly

Gabriel Menini June 26, 2007 at 7:59 pm

Nice tip. Thanks.

Now I did the same for an OpenSSH server which listen on port 22000. Actually, there's a firewall listening no that port, which redirects the incoming traffic to a LAN's OpenSSH server.

I've copied the key to the /home/user/.ssh/authorizedkeys but the client doesn't connect without prompting for the password...

Arul July 18, 2007 at 6:55 am

Hi, I am new to SSH. Can you tell me how can I automate connecting to remote unix boxes using SSH through a shell script and the SSH connections should take the password at runtime possibly using a config file

something like

cat server_repo.txt:
abc_server abc/def

where abc_server – unix box
abc – username
def – password

Note – I dont want to use "Passwordless Connectivity"

Thanks,
Arul.

BusyBecky September 5, 2007 at 2:49 pm

Hi,

Why is it mandatory to have the same username on both source and target servers?
Is there any workaround to this limitation?

Thnx.

nixCraft September 5, 2007 at 7:41 pm

No it is not required. You can use user name tom on client and username jerry on server.

surendra kumar May 21, 2008 at 4:36 am

hi vivek,
i think this method will not work for different users
ie what i want to say is user1 can not login to user2 account with out password in to server can u conform it and revert back again?
thanks in advance
surendra

Shankar September 17, 2008 at 4:39 pm

Hi Vivek,

In your step 3 as below. It will prompt for the password of user vivek on rh9linux.nixcraft.org to complete the copying of the public key.

$ scp .ssh/id_rsa.pub vivek@rh9linux.nixcraft.org:.ssh/authorized_keys2

Is there any method by which I can pass this value non-interactively.

Thanks
Shankar

Brendan October 18, 2008 at 12:51 pm

Regarding using scp to copy is_rsa.pub into authorized_keys2, I don't believe this to be a good idea if there is any chance that you need more than one user or public key to have access to the server.

In this case, rather do the following:
ssh vivek@rh9linux.nixcraft.org “cat >> .ssh/authorized_keys2” < .ssh/id_rsa.pub

This will pipe the public key through the ssh session and append it to the existing file if it exists. Otherwise it will create the file with the contents of your id_rsa.pub

Tricky April 15, 2009 at 6:13 pm

Lol. Came back here to figure out how I did that thing ^^ before. 😀

... and realised I hadn’t explained properly:
the authorized_keys2 file can contain multiple keys. By using scp, you might overwrite any previously-placed keys with a single key. By appending (using the >>) you specifically add your key to the end of the authorized_keys2 file and you won’t lose any previous keys.

sandip April 23, 2009 at 7:04 am

hi
i hav did as u mentioned abow but it wont work it is asking for the passwd

hari May 29, 2009 at 7:49 am

Hi,

Please run # passwd -d login_name for each user and
then check.

regards
hari

Rajesh June 12, 2009 at 6:28 am

HI Vivek,

Your article on SSH is very nice. It very helpfull for us.

Keep doing the great work

Regards,
Rajesh

sreekar September 8, 2009 at 7:05 pm

sir,
your article is very educational. i also referred your tutorial on shell scripting. The way you write in simple language makes a difficult concept also understandable. I think this is a trait of all Indian writers.

thank you for the good work

sincerely,
sreekar

> **nixCraft** September 9, 2009 at 4:30 am
>
> @Sreekar,
>
> Thanks for feedback!
>
> I'm glad to know this site helped you to understand Linux and shell scripting.

Wanga October 22, 2009 at 9:55 am

Am not able to login into another computer even after installing ssh on both computers.
It tells me the permission denied ,please try again and when i try again it doesnt log in.
And yet other people are able to use ssh comfortably. My computer is also uptodate

Tricky October 22, 2009 at 12:46 pm

Hi Wanga

Likely you have not got the ssh daemon running on the computer you want to connect to, though there could be many other reasons it is not working. Could you paste any error messages you might be getting when you try to connect?

crazyswap January 9, 2010 at 8:26 am

I can't log into my server,it shows network error:connection time out.kindly help.

Tricky January 9, 2010 at 3:04 pm

Hi crazyswap

Try running a tcptraceroute (http://en.wikipedia.org/wiki/Tcptraceroute) to your server to confirm that the problem is not the network:
tcptraceroute server.name.or.ip 22

You may need to install tcptraceroute.

If tcptraceroute fails only on the last step then it is likely that the ssh service is not running on the server. If your server is under paid hosting, contact your hosting provider to find out what the cause is.

Barun May 1, 2010 at 7:14 pm

Hi Vivek,

Is there any way to skip typing in the passphrase while login through ssh? For example, some cron jobs run daily, which open ssh sessions to remote machines to do something. Even to have 'ssh-add' executed, we need to provide the passphrase.

~ Barun.

> **nixCraft** May 3, 2010 at 9:19 am
>
> Try keychain
>
> > **Tricky** May 3, 2010 at 8:24 pm
> >
> > I'm not sure if keychain would work for ssh sessions created by cronjob while you're not logged in. A passphraseless key would work in that case except that passphraseless keys are not so good. What you could do is limit a separate passphraseless key to only be able to execute a single command:

Add a separate key to the authorized_keys file but start the line of the key with the command that will be run remotely. For example if you want to remotely execute a script called /usr/local/bin/cronjob1, put the key in as:
command="/usr/local/bin/cronjob1" ssh-rsa AF899EDC23.....rest-of-key.....A3C== cronjob_description@my-desktop

Then in the cronjob, ensure that the ssh session specifies that you want to use a non-default ssh key with "-i":
0 22 * * * /usr/bin/ssh -i /home/user/.ssh/cronjob1id_rsa user@server "/usr/local/bin/cronjob1"

When the new key is used, the server will always execute the cronjob1 script even if you specify a different command. This can be useful in other ways however I think this is getting towards tutorial territory. 😉

nixCraft May 3, 2010 at 8:48 pm

> Iâ€™m not sure if keychain would work for ssh sessions created by cronjob while youâ€™re not logged in.
Why not? We have live backup server that pulls data from 20 Linux servers using rsnapshots. rsnapshots is called from cronjobs, all you've to do is in your backup script:

```
#!/bin/bash
# get keys for ssh, rsync, rsnapshot
/usr/bin/keychain /root/.ssh/id_dsa

# start backup
rsync source dest...
```

All my backup server ssh keys are protected and server generally don't go offline. I've the following in /root/.bash_profile

```
/usr/bin/keychain --clear $HOME/.ssh/id_dsa
```

The –clear option is very handy as it allows cron job to do password less login but all users including an intruder must provide a passphrase-key for interactive login.

HTH

> **Tricky** May 3, 2010 at 10:39 pm
>
> > > I'm not sure if keychain would work for ssh sessions created by cronjob while you're not logged in.
> > Why not?
> Maybe should have been more specific – I'm referring to keys which have a passphrase as these keys cannot be used non-interactively.
>
> I do like the –clear now that you've made me aware of it. ☺

sakthi February 3, 2011 at 6:36 am

WE have a script which tries to scp to the same machine
machine1>> scp -r user@machine1:fromdir todir
As the keys are not inplace it is prompting for password. Is there any way we could automate this part by generating keys?. I would appreciate if you could give me the steps to perform the ssh.

Allen Cohen July 12, 2011 at 1:27 am

I've used your method to ssh without a password for a non-root user, say "user". This works as long as I'm logged in as "user".
But if I run as root, the following still asks for "user"'s password.
i.e.: the following works w/o a password:
su – user
ssh host date
But the following asks to the password of "user":
su – root
ssh user@host date

Daniel January 10, 2012 at 1:58 pm

Hi all,

Can perhaps anybody give me a hint for the following ssh issue?

I have machine A and machine B. (AIX machines). I'm logged in as root and wants to check/create ssh keys for some users. For example user STAFF1 has ssh keys on machine A but not on machine B I would like to create ssh keys (ssh-keygen -t rsa ...)
To check if keys are already there I just would check if id_rsa and id_rsa.pub files are existing in machineb:/home/STAFF1.
The main problem is how to generate keys for / as user STAFF1 on the remote machine? My understanding is that I need to be the user when I create the keys, otherwise I would need to use su in a way that it works on a remote machine like
> ssh machineb 'su STAFF1; ssh-keygen -t rsa...' which doesn't work.
Is there a command where I (as root) can create keys for another user????

I'm looking extreeeeeeeeeemly forward to here something from you ☺

Best regards from Germany,
Daniel

Patch March 30, 2012 at 3:53 pm

@Daniel

When creating a key it is only user specific because of where it is located, create your key and then move to the appropriate users home directory usually under a sub directory .ssh.

ssh someuser@machineb ls -al /home/STAFF1/
ssh someuser@machineb ssh-keygen -t rsa -C "User STAFF1 key" -f /home/STAFF1/.ssh/id_rsa

-C is a comment you can associate with the key to know who its for and -f places the key in the correct location.

Nasimuddin Ansari September 5, 2012 at 7:47 am

ssh-copy-id command is better way to copy identity (keys) on remote system. It is basically a shell script – /usr/bin/ssh-copy-id

$ man ssh-copy-id

> **Pavel** December 11, 2012 at 8:45 pm
>
> But this tool is included to only few distros. Sadly not all...(working daily on Solaris :-()

Ritesh September 11, 2012 at 2:20 pm

Hi,

I am trying to connect to b@M1 from a@M1.
Upto step to is fine but after that the key fingerprint is generated.
and when i try to copy *.pub file onto b@M1, it prompts for password.

Kindly help.

Thanks,
Ritesh

Fred February 6, 2013 at 6:58 pm

authorized_keys2 has been deprecated since 2001. You should just always use authrized_keys

Vijay Kanta July 20, 2013 at 6:56 am

This website and the author never cease to amaze me. You have taught me a lot in my Linux journey. Kudos for the very helpful article. 😉

mohamed October 18, 2014 at 8:42 am

while installing oracle grind infra structure ssh cat work through the forms but it work manually fine without password ...ssh node2 date & ssh node1 date works whats the problem plz ??

Have a question? Post it on our forum!

PRIVACY

TERM OF SERVICE

CONTACT/EMAIL

DONATIONS

SEARCH