What is the password for ``ssh root@localhost``?

7

I need to ssh to localhost using root account, by `ssh root@localhost` . When it prompts for passwords, I can not login with all possible passwords. On setting of localhost machine, regular user `xxx` and root user share the same password (the password that works for `sudo -s`), but it does not works for `ssh root@localhost` . So where to look at the password for `ssh root@localhost`

PS: with the password, I can login to regular account on `ssh xxx@localhost` .

PPS: to answer further questions on motivation of doing so, `localhost` is just a computer in a private network and setting up `ssh root@localhost` is just to relieve some manual management in a prototype system.

2

ssh root

1 Are you doing this in a script, or is this for interactive use? If interactive, why not just log in as some other user and then use `sudo` as needed? If for a script, then you really shouldn't be using password authentication anyway, since it means you have to put the password in plaintext in the script. – Daniel Pryden Aug 3 '12 at 15:46

I am doing this in a script, but a passphraseless ssh based on public/private key is intended to setup, so password is not really stored in plaintext. Good comments, though – Richard Aug 4 '12 at 14:50

6 Answers

17

`ssh root@localhost` uses the same password for root. It looks like you have not set root password. To do that log in as root using `sudo -s` then use `passwd` command to set root password.

After that you must be able to ssh as root

- How to find out root password for installing software

edited Apr 13 '17 at 12:23
Community ♦
1

answered Aug 3 '12 at 12:10
Manula Waidyanatha
6,637 1 17 19

3 this works. just out of curiosity, is there any difference between the root account when doing `sudo -s` and when doing `ssh root@localhost` ? – Richard Aug 3 '12 at 12:20

2 you should never do that, if you set a password for the user root you will enable the user root and this is a really bad move for the system's security. – user827992 Aug 3 '12 at 13:53

1 @user827992 it is necessary to enable the user root to take advantage of ehealth-aussie.blogspot.com/2012/01/…. However, you can mitigate the security damage by disabling root login for openssh. dropbear will allow root login but only with a key and only for the brief time between power on and decryption. – emory Aug 3 '12 at 14:56

Login as a normal user and use `sudo -i` to drop to a root shell.

You should not use the root account or change it's password for the matter.


Bruno Pereira
61.7k 27 182 209

7

If you decide to use the root account.

Disable the ability to log in over ssh with root if ssh is exposed to internet. Instead login as a user, and `su` to root.

This will make sure to prevent brute force attacks against your computer. Since its harder to guess both the user and password. Knowing the username one would only have to brute force the password

changing the line:

```
PermitRootLogin yes
```

under `/etc/ssh/sshd_config`

to

```
PermitRootLogin no
```

and run:

```
service sshd restart
```

to reload the configuration.

edited Apr 9 at 11:09

answered Aug 6 '12 at 21:12


tomodachi
10.2k 4 23 44

The user root is not enabled and is not capable of doing a log in, you can see that doing so:

the field that usually contains an encrypted password is filled with a `!`.

If you read the shadow(5) manpage you will get this

> If the password field contains some string that is not a valid result of crypt(3), for instance ! or *, the user will not be able to use a unix password to log in (but the user may log in the system by other means).

You should not enable the user root for security reasons.

edited Jan 27 '15 at 17:17
muru
1

answered Aug 3 '12 at 12:11
user827992
2,024 2 14 16

0

set root password using following command

```
sudo passwd
```

answered Jun 17 '16 at 7:30
D Nilesh
150 6

Your answer while possible may be a bit late as the OP already accepted the other answer. Your help is of course appreciated, however you might be better served by focusing your efforts on the unanswered questions – Wayne_Yux Jun 17 '16 at 7:59

-1

They answered your question about login to ssh with non super user (say `goblin@192.168.0.3 -p 22`).

First create another account without `su` privleges. After you log into ssh as a non super user you can enter this command to switch to root (escalate your privileges):

```
su
```


muru
1

Sean KaliLinux Calcagni
1

1 they scripted it that way for security reasons. dont enter root as your username enter a non super user and the enter password for non superuser and the su to superuser (su stands for substitute user) its less safe letting you ssh directly into root than it is ssh-ing into average user and Substituting user or escalating privledges sudo stands for super user Do so while in a normal user account you can always type sudo before your actual command and it will grant superuser privleges for That command. <3 happy Linux-ing <3 – Sean KaliLinux Calcagni Nov 22 '16 at 18:32