# XigmaNAS - SSH Password-less / Key Authentication

#### Installed Version - NAS4Free-x64-LiveCD-9.0.0.1.43.ISO

## Introduction

SSH Key Authentication is used so you will not be prompted for a user-name and password when starting a secure, encrypted session. This is especially important if you want to automate your session. It is especially important if you will be transferring data over insecure networks such as the Internet.

This article provides links to information that has been known to help others. There is no best or easiest way to configure Key Authentication, everyone has their own favorite method. Everyone configures their system differently. You should follow the links and read the information, you may have to adapt procedures so they fit with your preferred system configuration. Eventually you will find a process that works great for you.

This article does not cover networking basics you will need to know and configure so you can communicate over the Internet.

## Prerequisites

Before deploying SSH Password-less / Key Authentication you should be familiar with:

- Linux file and directory permissions.
- Mounting and unmounting filesystems.
- Changing File Permissions and File Ownership.
- Executing commands in a terminal, CLI or via SSH session.
- Basic *nix commands and syntax.
- What Home directories are and how they are used.

Open a Terminal (Console, CLI) on your client PC and make sure you understand these commands:

- man mount
- man umount
- man chown
- man chmod

You can also look them up on FreeBSD.org's Man Pages.

### Install NAS4Free, if you already haven't

See SUG Section 2.3-Installing NAS4Free on disk.

Or you may run it from the CDROM as required for testing.

SUG Section 2.2-Using NAS4Free with the CDROM and a removable disk (LiveCD mode).

### Configure SSH Server and Client

For server details please read SUG Section 6.4-Service SSH. For client configuration and testing please read SUG Section 2.6.4-SSH Client Basic Configuration.

You should now be capable of communicating via SSH with your server. Use PING and TRACEROUTE to verify that each Server can reach the other. You can use WebGUI Tab> **Diagnostics|Ping** and **Diagnostics|Traceroute** for this purpose. Finally connect with your client software as explained in SUG Section 2.6.4-SSH Client Basic Configuration.

## Configure Password-less / Key Authentication

This provides the best level of security while using SSH. The following, high level procedure assumes you have basic knowledge of *nix and have already configured your SSH client and NAS4Free SSH server.

We are starting with a clean NAS4Free server and clean Linux client that have never connected with each other before and have never been configured for Password-less / Key Authentication. We will configure Password-less / Key Authentication for the server's root account.

This simple, easy, 7 step procedure will work assuming you faithfully follow the instructions, enter commands exactly as shown in the example session and have not screwed things up already by previously failing with another procedure. Why is this so? Because the folders and files you may already have created will not have proper permissions and the commands must use the existing folders rather than create new, correct ones for you.

1. Open a terminal session.
2. Connect to NAS4Free server via SSH as root using keyboard interactive authentication. This automatically creates ~/.ssh directory and ~/.ssh/known_hosts file on your client with correct permissions ( Windows users may have to create and secure ~/.ssh manually if their SSH client software is not properly configured ).
3. Execute ssh-keygen command accepting all defaults. This automatically creates ~/.ssh directory and 2048bit public/private keys.
4. Rename your public key to authorized_keys.
5. Exit your SSH session.
6. Copy your private key from NAS4Free server root ~/.ssh directory to your client user's ~/.ssh directory. Use whatever secure, encrypted method you like for this, I prefer SCP. If your *nix distribution does not include SCP by default, then you may have to install it.
7. Connect to your NAS4Free server via SSH, note that this time you are not asked for a password.

> ⚠ This walkthrough needs cleanup/repair because it destroys the client machine's ID (becoming the ID just generated on the NAS), which invalidates all the key authentication already in place from the client to other hosts.

In the sample session below each step is marked with an echo command.

```
youruser@ubuntu:~$
youruser@ubuntu:~# echo STEP#2
STEP#2
youruser@ubuntu:~$ ssh -l root 192.168.1.233
The authenticity of host '192.168.1.233 (192.168.1.233)' can't be
established.
DSA key fingerprint is b2:d0:99:cb:6e:b6:53:95:4d:f4:b3:02:1d:bc:32:db.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.233' (DSA) to the list of known hosts.
root@192.168.1.233's password:
Last login: Thu Apr  5 18:20:54 2012 from 192.168.1.233
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
    The Regents of the University of California.  All rights reserved.

Welcome to NAS4Free!

nas4free01:~#
nas4free01:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
f3:16:93:6a:65:73:b8:38:ab:0a:38:e7:92:8f:07:20 root@nas4free01.mzhome
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|                 |
|                 |
|E          o     |
|o       S O .    |
|..       B *     |
|ooo     = +      |
|o=..   . +       |
|.+o .....        |
+-----------------+

nas4free01:~# echo STEP#4
STEP#4
nas4free01:~# mv ~/.ssh/id_rsa.pub ~/.ssh/authorized_keys
nas4free01:~# echo STEP#5
STEP#5
nas4free01:~# exit
logout
Connection to 192.168.1.233 closed.
youruser@ubuntu:~# echo STEP#6
```

```
STEP#6
youruser@ubuntu:~$ scp -p root@192.168.1.233:~/.ssh/id_rsa ~/.ssh
root@192.168.1.233's password:
id_rsa                                        100% 1675     1.6KB/s   00:00
youruser@ubuntu:~# echo STEP#7
STEP#7
youruser@ubuntu:~$ ssh -l root 192.168.1.233
Last login: Thu Apr  5 18:22:36 2012 from 192.168.1.233
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
    The Regents of the University of California.  All rights reserved.

Welcome to NAS4Free!

nas4free01:~#
```

For more details and examples of how to generate keys and use them, please read the following:

- OpenSSH Key Management by Daniel Robbins. Good, basic explanation.
- *–dead link* ***{Setting up SSH with a private key}*** –.
- *–dead link* ***{Users passwordless ssh login}*** –.
- *–dead link* ***{SSH Security certificate}*** –.
- *–dead link* ***{server is refusing key}*** –.
- *–dead link* ***{SSH keygen.com does nothing?}*** –
- OpenSSH Client Documentation - Where you will find information about the SSH client used in NAS4Free. This is useful for learning how to start a session at the terminal or CLI.
- OpenSSH Manual pages - Where you will find all the documentation.
- http://svnweb.freebsd.org/ports/head/security/ssh-copy-id/files/ssh-copy-id?revision=300897&view=markup