How to set up SSH access on UNIX hosts

This document describes the process of setting up the SSH connection for running plug-ins from the Nagios component of GroundWork. Once in place, you can easily monitor many aspects of your UNIX hosts as long as they are running SSH.

Setting up SSH Monitoring on UNIX Hosts for GroundWork

Introduction

This document describes the process of setting up the SSH connection for running plug-ins from the Nagios component of GroundWork. Once this access is in place, you can easily monitor many aspects of your UNIX hosts as long as they are running SSH.

You need this setting of authentication keys so that the plug-ins run in a secure mode as the user from the GroundWork server may avoid embedded passwords.

The username on the Monitored systems, for simplicity, can be "nagios" and if it is possible, assign a uid of "502". Neither is essential but it can simplify the installation of the keys and the plugins.

The username on the Monitored systems can also be named "gwrk" or whatever. In this case it may be necessary to copy both the public key for user "gwrk" and for user "nagios" to the Monitored systems "gwrk" authorized_keys file.

Monitored System Requirements

User Login	An ordinary user login to the target host(s)
Home Directory	A home directory on the target host(s)
Sshd	Sshd daemon running on the target host(s)
Public Keys	Public key authentication enabled on the target host(s)

Red Hat Linux

- Log in as root to the target server
- Add the user, plus directory

```
useradd –m <username>
```

- Check that the user directory itself has permissions 755

(read-write-execute for owner, read-execute for group and for others)

```
ls –lad /home/<username>
```

- Set the user password

```
passwd <username>
```

- Give the user a password consistent with local policy.
- This should be the same password for all target systems to make it easy of the SE.

```
cd /etc/ssh
```

```
vi sshd_config
```

- locate entry for PubkeyAuthentication
- Make sure that it is set to "yes" and is not commented out.
- Save the file.
- Restart the daemon.

```
service sshd restart
```

Solaris

- Login in as root on the target server.
- Add the user and the user's home directory

```
useradd -m <username>
```

- check permissions of the user's home directory.

```
ls -iald /home/<username>
```

or

```
ls -iald /export/home/<username>
```

- If this client is using the autohome feature (one exported filesystem for all user home directories) there are some extra issues to deal with so that all hosts are included for login privilege and that the actual home directory that all target hosts share is the one accessed for this s

- Give the user a password

```
passwd <username>
```

- Give the user a password consistent with local policy.
- This should be the same password for all target systems to make it easy of the SE.
- Go to the ssh configuration directory (could be one of these:)
  - /opt/csw/etc
  - /usr/local/etc
- Another was to find it:

```
find / -name sshd_config -print
```

- Edit the file

```
vi sshd_config
```

- locate entry for PubKeyAuthentication
- Make sure that it is set to "yes" and is not commented out.
- Save the file.
- Restart the daemon.

```
/etc/init.d/sshd stop
/etc/init.d/sshd start
```

- Notice that the sshd daemon stops and restarts with a new process id

- If this is Solaris 8 then OpenSSL SSH is a Freeware addition (not a package). You will have to:
  - Get the source and install it on your system
  - Create an init script sshd for startup and shutdown and put it in /etc/init.d
  - Create links to that file in the appropriate state directories
  - Edit the sshd_config to disable role separation
  - Correctly set /etc/hosts.deny and /etc/hosts.allow with entries that permit the GroundWork server to access the ssh daemon. (The

At this point the client system is ready for GroundWork to create a public key for the user “gwrk” which will be connecting securely to execute agen steps are performed from the GroundWork server.

GroundWork

This example will use the username “nagios”. Any valid username may be used, as long as the checkcommands.cfg file refers to the same user name in the check_by_ssh definitions.

The example target host will be called "target".

- Log in to the GroundWork server as root, then become nagios

```
su - nagios
```

- Check that you are in the directory /home/nagios

- Create the public and private keys

```
ssh-keygen -t dsa -b 2048
```

- Hit enter on the passphrase prompts three times.
- Make sure the private key is locked down as owner, group “nagios” and permissions read write by owner only. If for some reason this is not so then make it so:

```
chmod 600 /home/nagios/.ssh/id_dsa
chown nagios:nagios  /home/nagios/.ssh/id_dsa
```

- Next, for each target, do the next 9 steps
  1. ssh to the target system. The First time you'll see a message about verifying the key, Enter yes and notice that the taget host has been added to the known hosts file.
  2. Check that you are now in the home directory on the target. If not, there is a problem you have to fix before continuing with this target. Fix it and continue.
  3. Create the libexec sub-directory under the users' home directory to hold the plugins.
  4. Create the .ssh sub-directory under the users' home directory to hold the public and private keys. chmod it to 700.
  5. Exit back to the GroundWork server.
  6. Copy the plugins to the targets /home/<username>/libexec directory.
     a. Be sure that they are the proper architecture for the target.
     b. Check that they have the right protections to allow them to be executed by the <username> on the target server.
  7. Copy the PUBLIC side of the DSA key to the target
     a. Copy using: scp .ssh/id_dsa.pub <username>@target:/home/<username>/.ssh/authorized_keys
     b. You may also copy the content of multiple id_dsa.pub files (e.g., for "gwrk" and for "nagios") using a cut and paste editor (these are text files).
     c. The point is to get them into the target machine’s "nagios" .ssh home directory in a file named authorized_keys.
     d. If this file already exists, or if there are multiple GW or other servers/users that will do ssh to this target, append the keys instead of overwriting the existing file.
  8. Test the connection from the GroundWork server with ssh. You should get in without a password prompt.
  9. Test the plugins as nagios, to simulate Monitoring.