TechTorials

Search

Navigation Home

- Home
- Tutorials
  - Batch Files
  - Cygwin
  - Linux
  - Windows
- Blog
  - Coding Projects
  - Site Updates
- Contact

Cygwin - SSHD Configuration

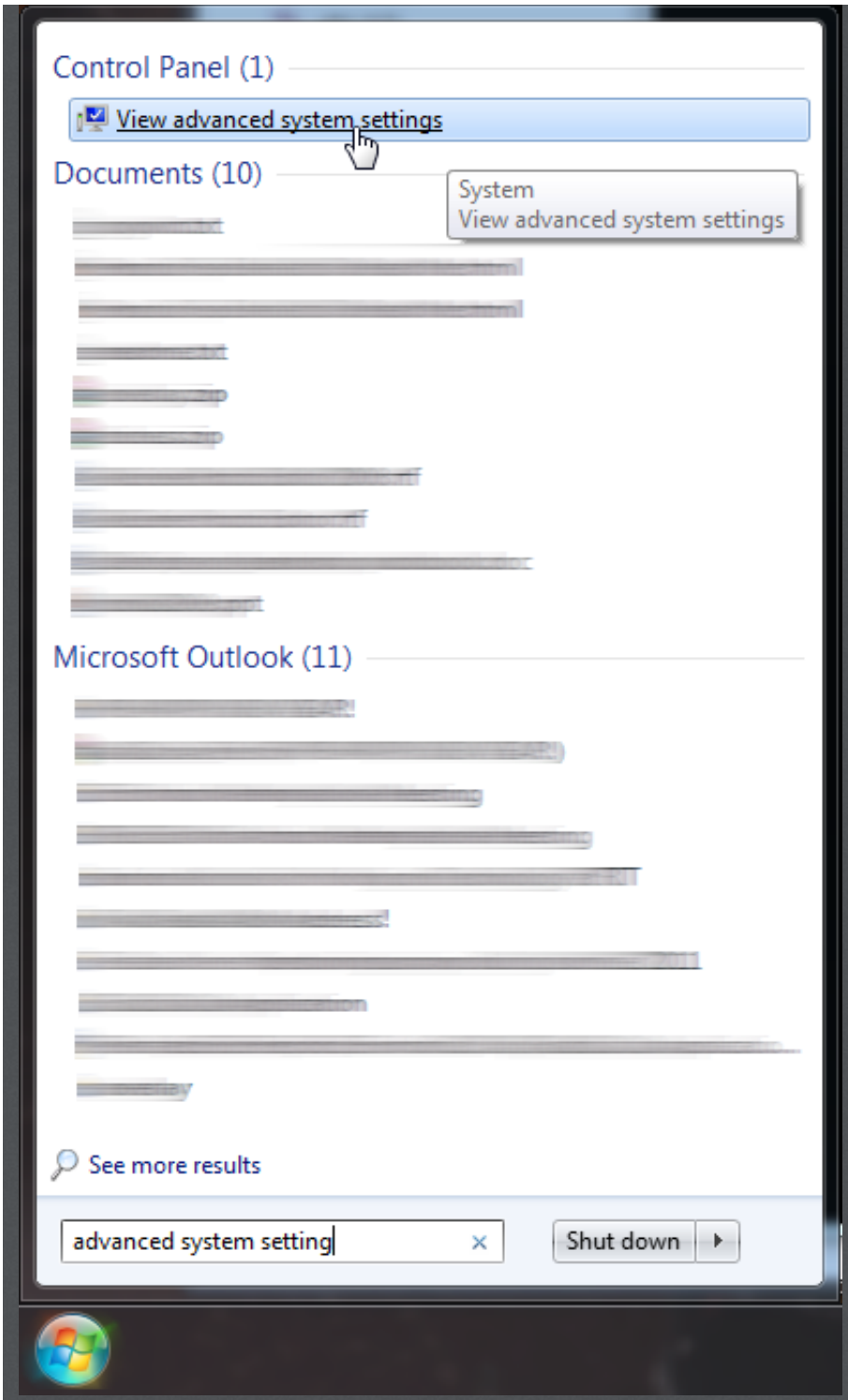
Written: 08/19/12

Last Updated: 06/15/13

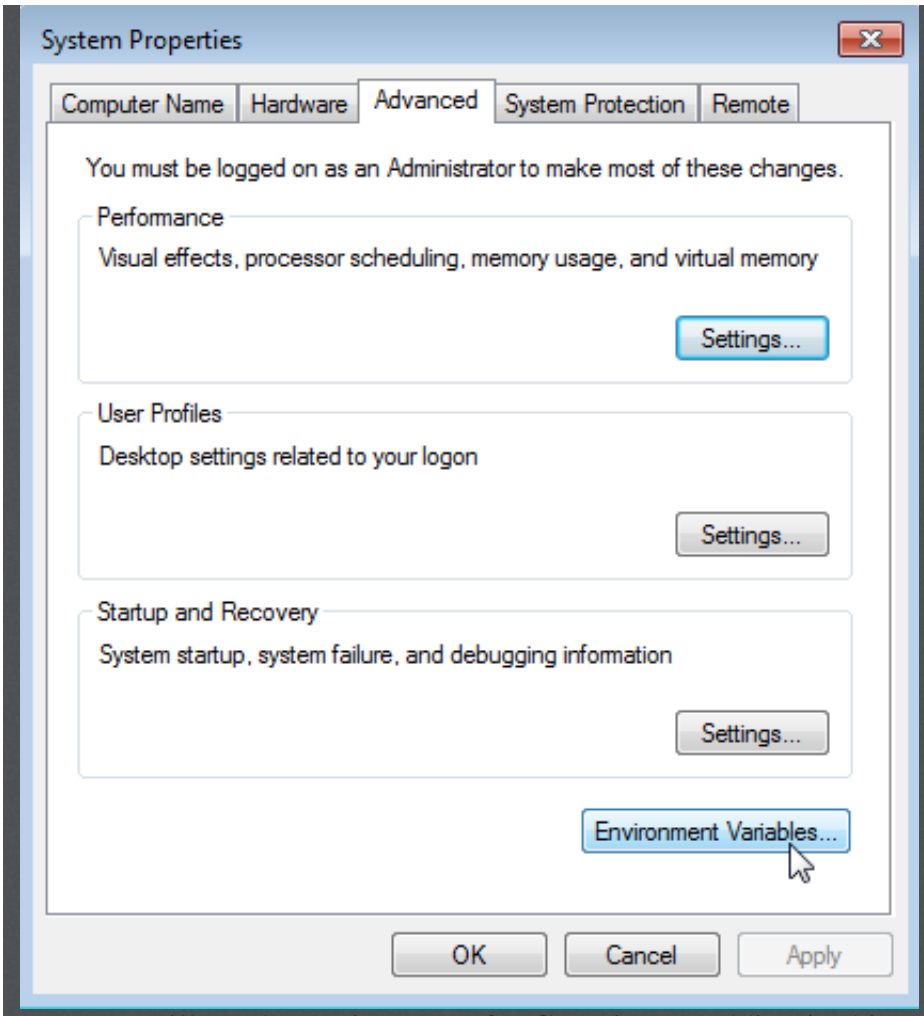
NOTE – With the environment variables “ntsec” and “tty” no longer being supported, this article has been completely updated to show how to create the server using the new settings. If you wish to see the legacy version, it can be found here.

This guide is the second part of my Cygwin SSH server series and assumes that Cygwin was installed according to the first guide. If you did not read the first part it can be found here. This article will focus on how to configure sshd, the ssh daemon, in Cygwin. My knowledge on this subject was obtained through painful experimentation and time spent searching online. Over two years have passed since that initial installation and I have still found Cygwin to be the best solution to my needs. I have also found some tweaks that I find make the setup easier and more secure. Those techniques will be shown in these guides as well as any known limitations.

Now that Cygwin has been installed, we need to make a Windows environment variable, create a new administrator account, install the daemon, and start the server. First, click on your start menu and type “advanced system settings”. Then click on the item in the Control Panel titled “View advanced system settings”.



The “System Properties” window should now be open. In the “Advanced” tab click on “Environment Variables…”.



Here we will set the environment for Cygwin. By adding in this environment variable, it will allow you to use any of the Cygwin commands without having to type in the full path to the executable. For example, you will be able to open up a terminal and type in “ls” to get a list of the directory contents, instead of having to type “/bin/ls.exe”. To set this variable, we need to edit the “Path” system variable. Click on “Path” under the “System variables” section (not under the “User variables” section) and click “Edit…”.

Traverse to the end of the string in the variable value field. Making sure that the string is not selected type “;c:\cygwin\bin” at the end of the string (or whatever the path to your /bin directory is). Make sure to include the semicolon and not to include any spaces. Click “OK”.

Click “OK”.

Click “OK”.

The sshd service that we will be creating must be run under a user. Do NOT use your current user. This is very important, because if you ever wish to delete the server you will need to delete the user. This username will also become the name of a service, so choose your name wisely. I like to use the name “sshd” as it explains exactly what the user is for. You can name it whatever you wish, just make sure to keep all of your names straight. Since I will be using sshd, you will need to exchange that name for whatever you pick throughout the rest of these guides.

To make a new user, click on the start menu and type “user accounts” in the search field. Click on “Add or remove user accounts”.

Click on “Create a new account” in the manage accounts screen.

Type in the name you wish to use for the user, select “Administrator”, and then click on “Create Account”.

Click on your newly created user’s account to edit it.

Click on “Create a password”.

Create a secure password. You should use mixed case (upper and lower case letters) and numbers at the very least. It is also a good idea to make it contain symbols and it should be longer than 7 characters. The security of this password is vital as the world will have access to try to break in and this could be your only defense. I will address security issues and show how to “lock down” the server in a later article. Windows requires that you create a password hint. I dislike this and do not wish to give anyone a hint. A trick to avoid giving a hint is to put a single space in the password hint field. Make sure you have this password handy as it will be used in the near future. Click “Create Password” once you are satisfied with the strength of your password.

We are now done with all of the Windows pre-setup and will proceed to the Cygwin configuration. Open up a Cygwin terminal. If you did not make an icon or a start menu entry you can find one by clicking on the start menu and typing “cygwin” in the search field.

Type the commands shown below in the terminal. These commands are used as permission workarounds to fix any permission errors that may occur.

```
chmod +r /etc/passwd
chmod u+w /etc/passwd
chmod +r /etc/group
chmod u+w /etc/group
chmod 755 /var
touch /var/log/sshd.log
chmod 664 /var/log/sshd.log
editrights -l -u sshd
editrights -a SeAssignPrimaryTokenPrivilege -u sshd
editrights -a SeCreateTokenPrivilege -u sshd
editrights -a SeTcbPrivilege -u sshd
editrights -a SeServiceLogonRight -u sshd
editrights -l -u sshd
```

The “chmod” command stands for change file mode bits. It is used to change the permissions on files and directories. The first four lines change the permissions on the files “passwd” and “group” to give us write access to them. The “passwd” file will contain all of the Windows users’ security information set by Windows; likewise, the “group” file will contain all of the Windows groups’ security information. The files do not contain any password information, as that portion is managed by Windows. These two files are critical as they will contain all of the user info needed for the ssh server.

The next three lines change the permissions for the log file so that logging may occur correctly. The “touch” command is used to change the timestamps on files. In this case we are updating the modification time to be the current time. To use the log file, the permissions on the “var” directory must be changed. The log file that we are changing permissions to is designed to log the actions of the ssh server. There are much better logging tools available and in a later tutorial I will go over how to use them, but for the meantime it’s important to set this up.

The last six commands are used to change Windows user rights and privileges. It is important that when issuing these commands you use the username you picked for the server in the place of “sshd”. The command in lines eight and 13 are used to display the permissions given to the user. There should not be any permissions set the first time the command is given, and afterwards the user should have all of the permissions assigned in lines nine through 12. These advanced privileges are necessary for the service to run correctly. More details can be found as to what each privilege is by doing a search at Microsoft’s TechNet Library.

Now that we have a user and that the permissions are all configured correctly, it is time to install the service. Start the script by typing the command “ssh-host-config”. The answer to give to all of the questions is shown below. Once again where I use “sshd” make sure you use the correct name given earlier. The last two lines will be the password that you assigned the user earlier.

```
1  ssh-host-config
2  yes
3  yes
4
5  yes
6  sshd
7  sshd
8  PASSWORD
9  PASSWORD
```

To setup Local Security Authority (LSA) authentication, run the script by issuing the command “cyglsa-config”. LSA authentication will allow for the sshd service to run under the SYSTEM account, among other things. Here is a good explanation as to everything that it entails. Answer “yes” to all of the questions. This will automatically reboot your system so make sure you are ready to reboot before running the script. You have the option to delay the reboot, but it is a good idea to go ahead and reboot right after this command is issued.

Cygwin should now be correctly configured with sshd. When you reboot your computer you should see the service sshd (or whatever you called it) running. At this point, only the most basic setup has been completed. Security and firewall settings, among other things, still need to be configured. Click here to go to the next topic in the series.

Share this:

Discussion (5)

There are 5 responses to “Cygwin – SSHD Configuration”.

1.

SshDemon responded:

10 months ago · Reply

Excellent instructions, but a few comments:
(1) You may need to revise these instructions to reflect installing latest Cygwin/SSH on Windows 8.1. Things seem quite different now. Also, the default ssh-host-config is generating TWO new user accounts. One called “cyg_server” and the other “sshd” (defaults). This is very confusing since it seem that you should not use either one for SSH login purposes. In your instructions it seem that you replace the former with the latter.
(2) I’m not sure you need the “cyglsa-config” anymore. In fact it is very unclear why you need it. Please explain better or check up on it.
Thanks for your effort.
(3) You instructions for creating Windows User Groups doesn’t work for non-professional editions of Windows 8/8.1 and others. This is very disappointing and confusing. It may still be possible by command line.

- ### James responded:

  10 months ago · Reply

  Thank you for your input. To address your points:
  (1) I assumed things would be different on Windows 8.1; however, I haven’t had the time to look into it. It sounds like they have updated the script; back when I wrote this tutorial it did not create a cyg_server user. I used the sshd account to keep things separate. The account technically has SSH permissions; however, I view it as more of a “root” account and simply keep it locked down. It should be possible to link sshd to any other account, but I have not personally tested this.
  (2) Please refer to the link for more information about that particular script. The script was needed back when I wrote this tutorial (things may have changed…). It gave the SSH server LSA authentication, which is needed in Windows to allow users to log onto the system. You can read more about LSA authentication, here.

(3) That is quite upsetting, thanks for pointing that out. It appears that the “Local Users and Groups Manager” (lusrmgr.msc) is not included in the home editions. A workaround would be to use the “net localgroup” command, which should work fine for Windows 7/8 non-pro editions. You can use it via the following syntax “net localgroup [groupname] [username] /add”

2.

SPalato responded:

6 months ago · Reply

I had problems getting “ssh-host-config” to work without warning using cygwin64 on Window 8. /etc/passwd/ wasn’t getting updated by windows, and chown couldn’t find the user. That can be fixed using “mkpasswd -cl > /etc/passwd”.

Thanks for the good tutorial!

-

James responded:

6 months ago · Reply

Thanks for the feedback! I’m sure that information will help others.

3.

anil responded:

5 months ago · Reply

Thank you very much for details instructions. I have tried this on Windows-2012 R2 x64 with latest Cygwin. Worked perfectly.

Leave a Reply

Enter your comment here...

Site Navigation

- Batch Files – Overview
- Blog
- Contact

Top Posts and Pages

Recent Pages

Recent Posts

- **Python Status Bar**

  September 27, 2014

- **DDoS Attack**

  March 8, 2014

- **Gentoo Kernel Articles Completed!**

  November 16, 2013

- **Gentoo Series Update**

  November 3, 2013

- **Parts 2 & 3 of Gentoo VMware series completed!**

  August 25, 2013

Subscribe via Email

Enter your email address to subscribe to this site and receive notifications of new posts by email.

Join 7 other subscribers

Email Address

Subscribe