**RUNDECK DOCUMENTATION**

**http://rundeck.org/docs/index.html**

Rundeck Installation

## Yum RPM Install

Execute:

yum install java-1.8.0   
rpm -Uvh http://repo.rundeck.org/latest.rpm   
yum install rundeck 

Open rundeck-config file and update hostip and reboot the server:

vi /etc/rundeck/rundeck-config.properties

visit: <http://hostname:4440>

replace hostname by ip of rundeck server.

service rundeckd start

You can change admin password in realm file.

Rundeck Linux Nodes Configuration

Follow video link given bellow:

https://www.youtube.com/watch?v=RxNSqprG\_BU

Rundeck Windows Nodes Configuration

While trying to evaluate Rundeck I came across one challenge i.e integrating rundeck windows nodes. Though rundeck work perfectly fine with non windows nodes but for windows nodes management there are not much documentation or working examples. I had tried to use the google groups for getting some info on [rundeck windows nodes integration](https://groups.google.com/forum/" \l "!msg/rundeck-discuss/iKaUcfzmCYA/eb1B3nMGDWsJ" \t "_blank) and to my surprise I got few responses but all seems to be a work around way to integrate windows nodes.

Current [latest rundeck version](http://rundeck.org/downloads.html) is 2.5.1 with many different updates and new features. So in this article I will be using the latest available rundeck for all integration tasks. I had used [Rundeck WinRM plugin](http://rundeck-winrm-plugin/" \t "_blank) which uses the [overthere plugin](https://github.com/xebialabs/overthere/" \l "cifs_host_setup" \t "_blank)  Follow below steps to integrate rundeck windows nodes for inline commands. Overthere plugin does not currently support inline script execution etc. So to workaround the limitations I have done below steps.

Summary of steps needs to be done to integrate a windows node:

**1. Setup overthere plugin in rundeck server.**

**2. Setup/enable winrm in remote windows client.**

**Purpose:** To execute inline commands at remote windows node via winrm

**3.**[**Setup/enable OpenSSH server**](http://www.techpaste.com/2015/06/windows-ssh-server-setup-and-configuration/)**in remote windows node**

**Purpose:** For copy of files and password less login to remote server, etc.

**4. Disable the administrator mode so that it won’t ask for the UAC(User Access Control) screen each time.**

**Purpose:**To enable remote windows node commands to run in administrative mode by default.

**5. Add the windows node to resource.xml**

**Purpose:**To enable the windows server listed as a node ready to receive commands from rundeck.

**6. Execute test commands to check the integration.**

**Purpose:** To make sure windows node is properly integrated to Rundeck server

### **Setting Up overthere plugin in rundeck:**

1. Download the plugin from Github [Overthere Plugin](https://github.com/rundeck-plugins/rundeck-winrm-plugin/" \t "_blank)
2. Login to Rundeck machine and cd to RUNDECK\_BASE/libext or in this article /opt/rundeck/libext
3. Copy the rundeck-winrm-plugin-1.3.1.jar file to the libext folder.
4. Restart the RunDeck services to make the plugin active.

### **Setup/enable winrm in remote windows client:**

1. Login to the windows node machine as administrative user using remote desktop.
2. Open command prompt in administrative mode and execute below commands one by one to enable WinRM in the windows server

You can run the script to setup winrm available below

winrm qc  
winrm set winrm/config/client/auth @{Basic=”true”}  
winrm set winrm/config/service/auth @{Basic=”true”}  
winrm set winrm/config/service @{AllowUnencrypted=”true”}

C:\Windows\system32>winrm qc

WinRM already is set up to receive requests on this machine.

WinRM is not set up to allow remote access to this machine for management.

The following changes must be made:

Create a WinRM listener on HTTP://\* to accept WS-Man requests to any IP on this machine.

Enable the WinRM firewall exception.

Make these changes [y/n]? y

WinRM has been updated for remote management.

Created a WinRM listener on HTTP://\* to accept WS-Man requests to any IP on this machine.

WinRM firewall exception enabled.

C:\Windows\system32>winrm set winrm/config/client/auth @{Basic="true"}

Auth

Basic = true

Digest = true

Kerberos = true

Negotiate = true

Certificate = true

CredSSP = false

C:\Windows\system32>winrm set winrm/config/service/auth @{Basic="true"}

Auth

Basic = true

Kerberos = true

Negotiate = true

Certificate = false

CredSSP = false

CbtHardeningLevel = Relaxed

C:\Windows\system32>winrm set winrm/config/service @{AllowUnencrypted="true"}

Service

RootSDDL = O:NSG:BAD:P(A;;GA;;;BA)S:P(AU;FA;GA;;;WD)(AU;SA;GWGX;;;WD)

MaxConcurrentOperations = 4294967295

MaxConcurrentOperationsPerUser = 15

EnumerationTimeoutms = 60000

MaxConnections = 25

MaxPacketRetrievalTimeSeconds = 120

AllowUnencrypted = true

Auth

Basic = true

Kerberos = true

Negotiate = true

Certificate = false

CredSSP = false

CbtHardeningLevel = Relaxed

DefaultPorts

HTTP = 5985

HTTPS = 5986

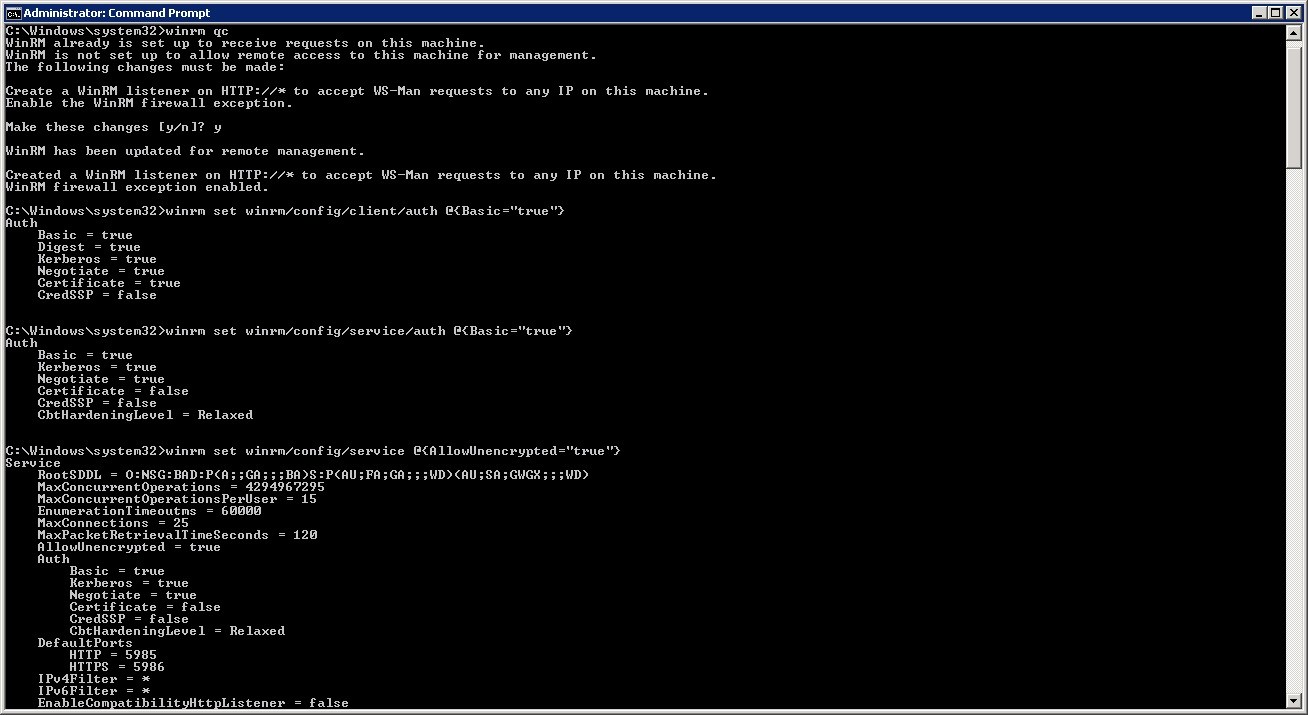
IPv4Filter = \*

IPv6Filter = \*

EnableCompatibilityHttpListener = false

EnableCompatibilityHttpsListener = false

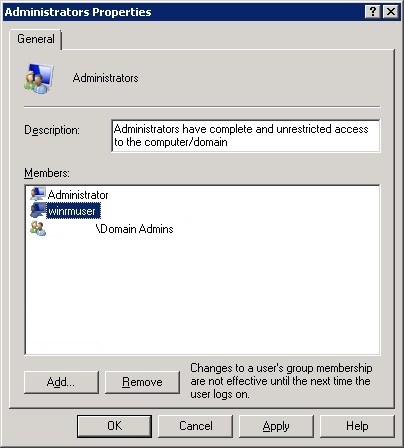
CertificateThumbprint

[](https://i1.wp.com/www.techpaste.com/wp-content/uploads/2015/08/winrm1.jpg)

Run below command to open the user and groups manager to add a user say winrmuser as the user to run the commands in remote user.

C:\Windows\system32> lusrmgr

Add a new local user winrmuser and add it to administrator group.

[](https://i2.wp.com/www.techpaste.com/wp-content/uploads/2015/08/winrm3.jpg)

### **Setup/enable OpenSSH server in remote windows node:**

Follow [Openssh Setup](#_Windows_ssh_server" \t "_blank) and [password less authentication](http://www.mls-software.com/opensshd-pki.html) link for setting up open ssh server for scp copy of files and artifacts to remote windows servers. Once this is done you will be able to send files with password less authentication from your rundeck server to the remote windows machine. If you have cygwin then you can also add that to the PATH to make your windows machine more compatible with running shell scripts also, so that you wont need to rewrite the shell script to batch script to run in windows machine.

### **Disable the administrator mode in remote windows machine:**

1. Execute below commands to open the windows policies using below command and complete below steps.
2. C:\Windows\system32> GPEDIT.MSC

a. Go to Computer Configuration –> Windows Settings –> Security Settings –> Local Policies –> Security Options  
b. Search for below policies

User Account Control: Admin Approval Mode for the Built-in Administrator account

User Account Control: Behaviour of the elevation prompt for administrators in Admin Approval Mode

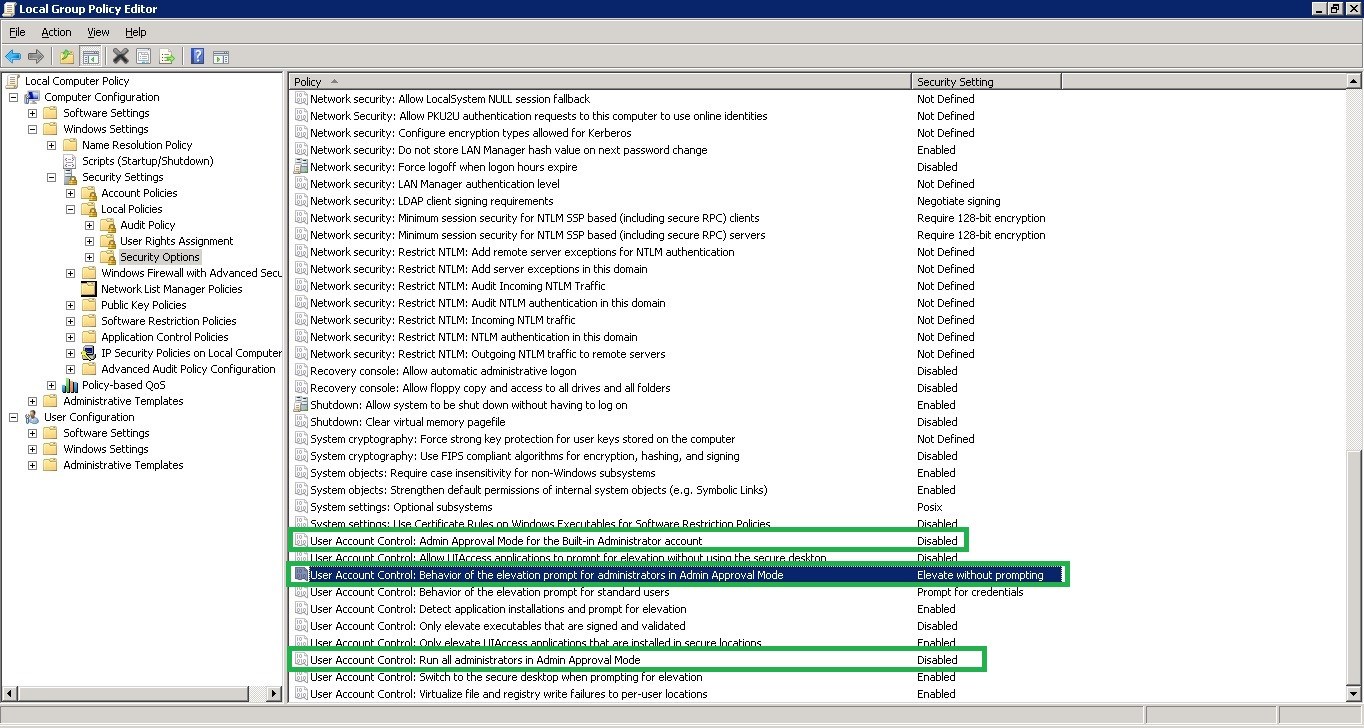
User Account Control: Run all administrators in Admin Approval Mode

c. Set the policies settings as follows in the following order as per the screenshot below:

Disabled

Elevate without prompting

Disabled

[](https://i1.wp.com/www.techpaste.com/wp-content/uploads/2015/08/disable_UAC.jpg)

### **Add the windows node to resource.xml :**

Open the resource.xml file present at /opt/rundeck/projects/MyProject/etc folder and add below entry to add the windows node to resource list.

<node name="remote\_windows\_host\_fqdn" connectionType="WINRM\_NATIVE" node-executor="overthere-winrm" winrm-password-option="winrmPassword" winrm-protocol="http" winrm-auth-type="basic"  username="winrmuser" winrmPassword="welcome\*123" description="Rundeck server node Windows" tags="" hostname="remote\_windows\_host\_fqdn:5985" osArch="x86\_64" osFamily="windows" osName="Microsoft Windows Server 2008 R2 Standard" osVersion="Microsoft Windows Server 2008 R2 Standard" />

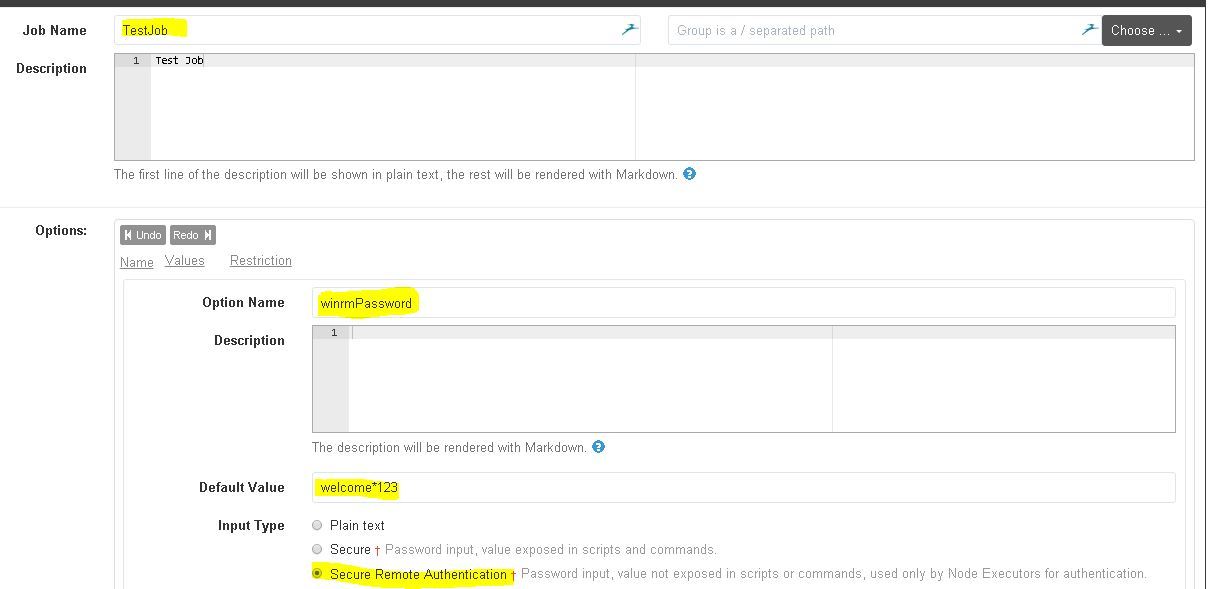
Example:

<node name="alfa1app1.techpaste.com" connectionType="WINRM\_NATIVE" node-executor="overthere-winrm" winrm-password-option="winrmPassword" winrm-protocol="http" winrm-auth-type="basic" username="winrmuser" winrmPassword="welcome\*123" description="Rundeck Exxon server node Windows" tags="" hostname="alfa1app1.techpaste.com:5985" osArch="x86\_64" osFamily="windows" osName="Microsoft Windows Server 2008 R2 Standard" osVersion="Microsoft Windows Server 2008 R2 Standard" />

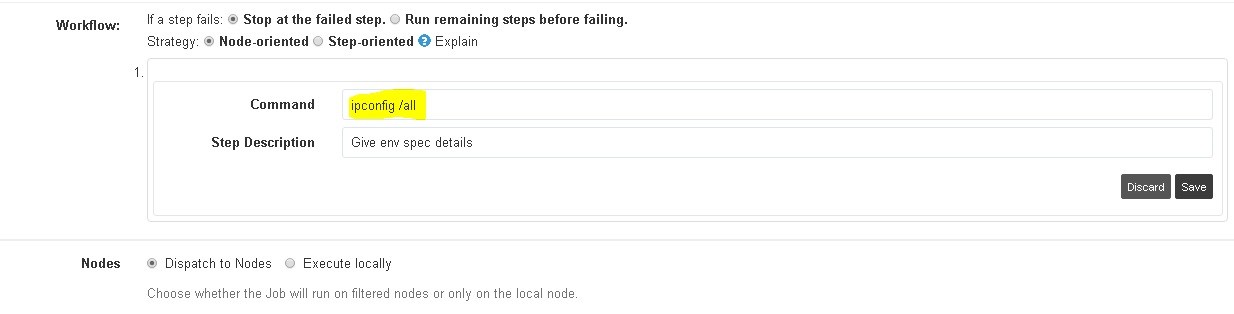
### **Execute test commands to check the integration:**

Create a test job with below options to authenticate and execute the remote command.

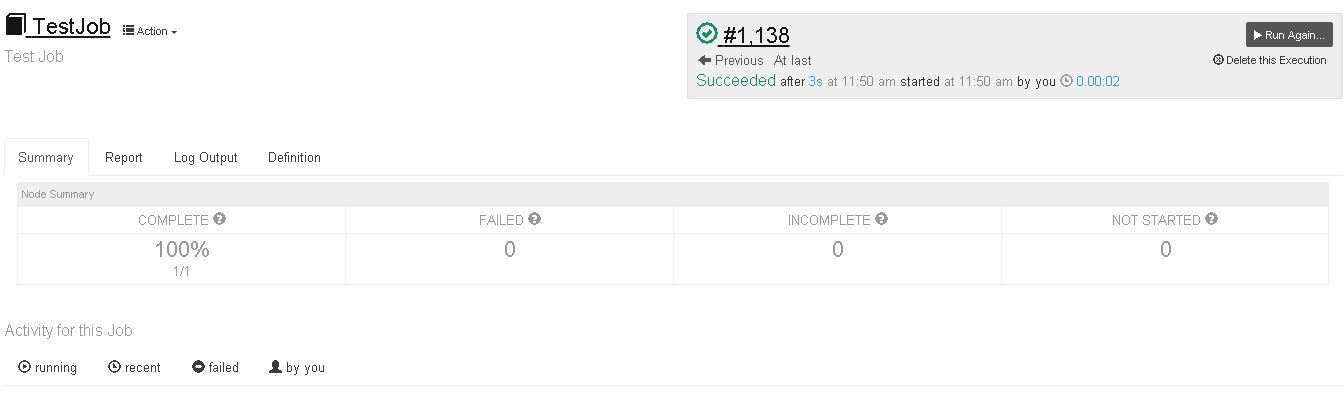
Make sure below options are checked.

[](https://i0.wp.com/www.techpaste.com/wp-content/uploads/2015/08/Rundeck_Job_WinRM.jpg)

Add a sample command (Ex: ipconfig /all) and choose execute on remote node and select the node which you had selected while adding to resources.xml file of rundeck.

[](https://i0.wp.com/www.techpaste.com/wp-content/uploads/2015/08/Rundeck_Job_WinRM1.jpg)

Now run the job and see if the job is successful and the output window is showing the commands output.

[](https://i0.wp.com/www.techpaste.com/wp-content/uploads/2015/08/Rundeck_Job_WinRM2.jpg)

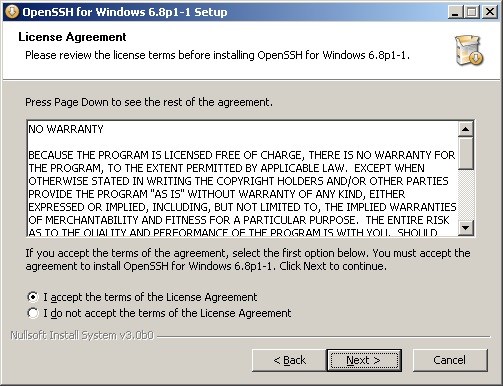
Note: Here we had a single project with multiple nodes out of which some are of windows and some were of unix nodes and rundeck server was hosted on a Linux machine. Here we had requirement of file copy, passwordless remote login to windows machine, unattended user access control and remote batch files execution, due to that reason we had to equip the remote windows host with so many pre-requisites. If you are not having any complex requirements and just want to play around creating a windows node then you can follow below url which has very simple way to integrate windows for sample command runs only. Do remember copy and other functionalities like inline script running etc, are not supported.

diegoluisi.eti.br/2015/06/linux/rundeck-how-to-add-windows-node/

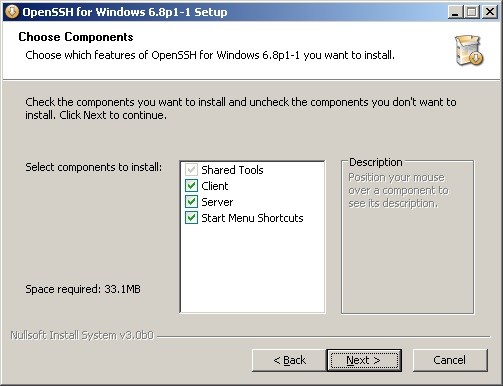
# Windows ssh server setup and configuration

For one of our projects we had requirement of accessing windows machines from linux environments to run few scripts for deployment. After little search we settled for openssh server for ssh connections and cygwin for script run in windows with public key authentication to have passwordless authentication. Below are the steps needs to be followed to achieve the same..

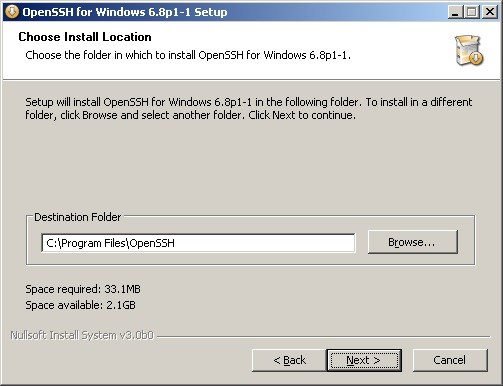
1. Download latest OpenSSH for your windows server from [Windows SSH server Download](http://www.mls-software.com/opensshd.html#botpage)  
2. We have used currently latest setupssh-6.8p1-1 version for this tutorial.  
3. Double click on the setupssh-6.8p1-1.exe file and click on next button.  
[](https://i0.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_1.jpg)  
4. Click on Next button and accept the license agreements.

[](https://i1.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_2.jpg)

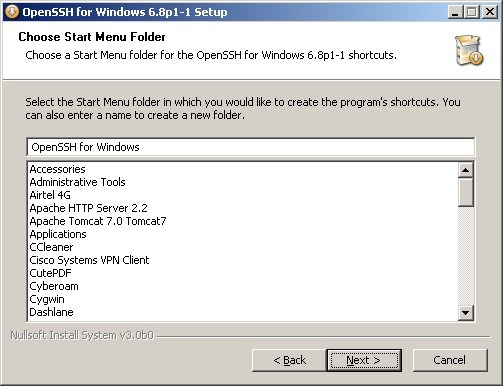
5.  Choose all the components and click on Next button.

[](https://i0.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_3.jpg)

6. Click on Next button keeping the location default.

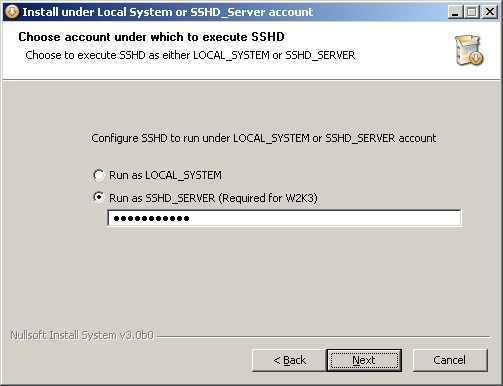
[](https://i0.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_4.jpg)

7. Click on Next button .

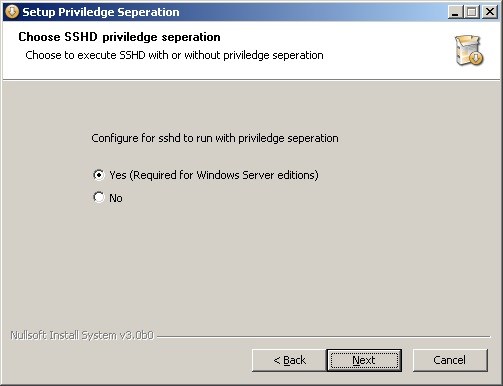
[](https://i1.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_5.jpg)

8. Choose Run as SSHD\_SERVER option and click on next after providing some password. (This option is required for password less authentication) .

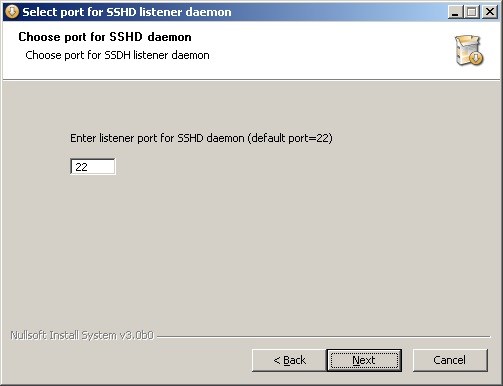
Note: Keep the password provided by you in some notepad as this needs to be used while authenticating for the first time before configuring the password less authentication mechanism.

[](https://i1.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_6.jpg)

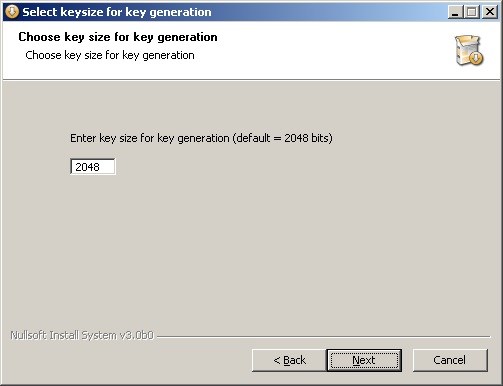
9. Choose SSHD privilege seperation – Yes

[](https://i0.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_7.jpg)

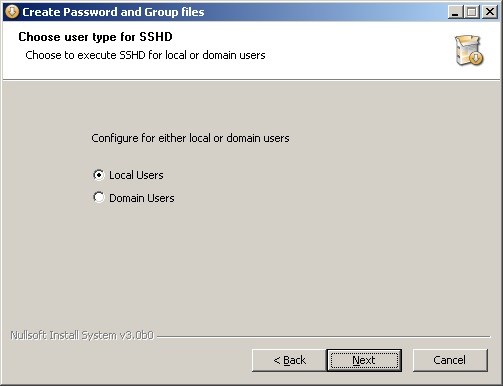
10. Keep the default port as 22 and click on Next button.

[](https://i2.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_8.jpg)

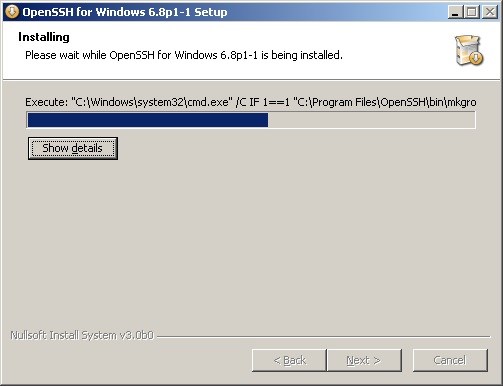
11. Keep the default key generation bits and click on Next button.

[](https://i0.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_9.jpg)

12. According to your requirement choose local or domain users option.

[](https://i1.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_10.jpg)

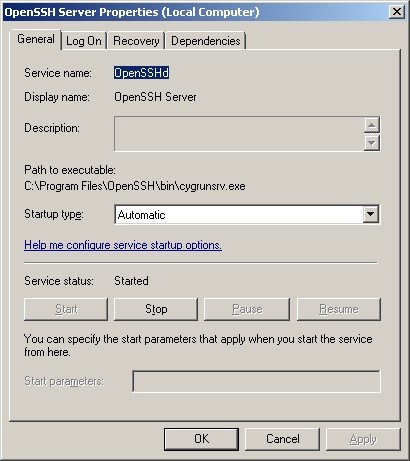
13.  Wait for the installation to complete. You might see some flash command prompts during the installation.

[](https://i2.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_11.jpg)

14. Click on Finish button to complete the installation.

[](https://i2.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_12.jpg)

15. This completes the installation of OpenSSH server. Make sure you have the new openSSH service created in services.msc panel. Make sure you are able to start the service successfully.

[](https://i2.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_13.jpg)

### **For password less authentication or public key authentication configuration follow below steps:**

1. Stop the SSHD service from services.msc console.  
2. cd to C:\Program Files\OpenSSH\etc and open sshd\_conf file in any text editor. Modify/Add below details.

PermitRootLogin yes

StrictModes no

RSAAuthentication yes

PubkeyAuthentication yes

AuthorizedKeysFile .ssh/authorized\_keys

IgnoreUserKnownHosts yes

3. cd to C:\Program Files\OpenSSH\etc and open ssh\_config file in any text editor. modify/Add below details.

RhostsRSAAuthentication yes

RSAAuthentication yes

PasswordAuthentication yes

4. Restart the SSHD server either using the command prompt in administrative mode or using the services.msc console to make the changes take effect.

5. Once Restart completes successfully. Use [putty](http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html) or similar software to login to the Linux machine from where you want password less authentication or public key authentication setup.

– Try logging in to the windows host using the password to make sure connection is getting established.

ssh windows\_user@windows\_host\_name

6. Generate the RSA keys like below:

[root@MyLinuxMachine ~]# ssh-keygen -t rsa

Generating public/private rsa key pair.

Enter file in which to save the key (/root/.ssh/id\_rsa):

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:

74:4a:71:b9:ab:cb:96:cc:68:77:c7:0e:19:bd:3b:ef root@MyLinuxMachine

7. Now execute below command to copy the keys to the windows machine for password less authentication.

ssh-copy-id windowsusername@mywindowsmachine

if you get below error and ssh-copy-id fails to copy to your windows machine then follow below steps to manually copy the keys.

246 [main] cat (61540) D:\cygwin64\bin\cat.exe: \*\*\* fatal error - cygheap base mismatch detected - 0x1802F1408/0x1802FA400.

This problem is probably due to using incompatible versions of the cygwin DLL.

Search for cygwin1.dll using the Windows Start->Find/Search facility

and delete all but the most recent version. The most recent version \*should\*

reside in x:\cygwin\bin, where 'x' is the drive on which you have

installed the cygwin distribution. Rebooting is also suggested if you

are unable to find another cygwin DLL.

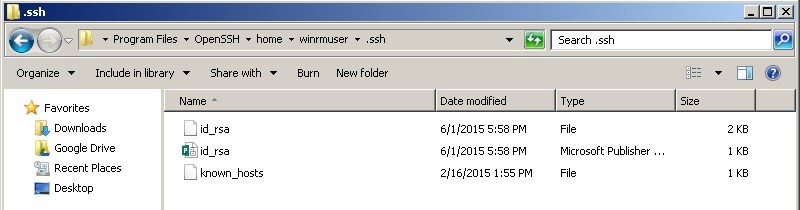
/bin/sh: line 1: 60464 Segmentation fault cat >> .ssh/authorized\_keys

Now try logging into the machine, with "ssh 'windowsusername@windows\_machine\_name'", and check in:

.ssh/authorized\_keys

to make sure we haven't added extra keys that you weren't expecting.

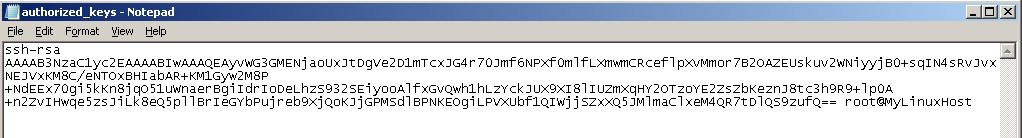
– Create a .ssh folder under C:\Program Files\OpenSSH\home\WINDOWS\_USERNAME folder. Example: winrmuser under we have created a folder .ssh like below screenshot.

[](https://i2.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_14.jpg)

– Create a file called authorized\_keys in notepad and add below output from the public key generated at the Linux Host.

[root@My\_Linux\_Machine ~]# cat /root/.ssh/id\_rsa.pub

ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAyvWG3GMENjaoUxJtDgVe2D1mTcxJG4r70Jmf6NPXf0mlfLXmwmCRceflpXvMmor7B2OAZEUskuv2WNiyyjB0+sqIN4sRvJvxNEJVxKM8C/eNTOxBHIabAR+KM1Gyw2M8P+NdEEx70gi5kKn8jqO51uWnaerBgiIdrIoDeLhzS932SEiyooAlfxGvQwh1hLzYckJUX9XI8lIUZmXqHY2OTzoYE2ZsZbKeznJ8tc3h9R9+lp0A+n2ZvIHwqe5zsJiLk8eQ5pllBrIeGYbPujreb9XjQoKJjGPMSdlBPNKEOgiLPVXUbf1QIWjjSZxXQ5JMlmaClxeM4QR7tDlQS9zufQ== root@My\_Linux\_Machine

[](https://i2.wp.com/www.techpaste.com/wp-content/uploads/2015/06/OpenSSH_server_Installation_15.jpg)

– Save the file.

8. Now try to login from the linux host using the same username and windows hostname. It will not ask for username and password and will directly login to your windows host.

ssh windows\_user@windows\_host\_name

#### **Troubleshooting:**

Incase the service fails to install or fails to start after install then follow below steps to fix it.

Note: You might need cygwin utilities installed incase below commands are not available

Make sure below config is there in PATH in sysdm.cpl if not add it.

%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\;C:\Program Files\Microsoft\Web Platform Installer\;

Go to the OPENSSH Home folder and run below command

chmod 700 /var/empty

run below command to check the local users available:

mkpasswd -l

Search for the sshd\_server username and execute the command like below to change the ownership.

chown WINDOWS\_HOSTNAME+sshd\_server /var/empty

Now try to start the service and check the logs if does not start: cat /var/log/sshd.log

you can run below command to check if the service has started listening on port number 22.

netstat -an | find “22”

# If the above password authentication doesn’t work the force the users to provide password manually but drawback is that if password is different for different node then the above type of windows node authentication will not work:

# In the options field add a field called exactly like “winrmPassword” and make rundeck use it for secure password authentication.

# 

# Update resources.xml file for the job:

# <node name="win\_node" connectionType="WINRM\_NATIVE" node-executor="overthere-winrm" winrm-password-option="winrmPassword"

# winrm-protocol="http" winrm-auth-type="basic" username="winrmuser"

# hostname="ec2-54-213-198-191.us-west-2.compute.amazonaws.com"/>

# 

Key based authentication windows:

1. At first make sure password less authentication is happening though openSSH.
2. Run the bellow script to properly configure winrm in node.

#WinRM Settings

winrm invoke EnableRemoting http://schemas.microsoft.com/wbem/wsman/1/config/service

winrm s winrm/config/service '@{IPv4Filter="\*"}'

winrm s winrm/config/service '@{IPv6Filter="\*"}'

winrm s winrm/config/service/Auth '@{Basic="true"}'

winrm s winrm/config/service '@{EnableCompatibilityHttpListener="true"}'

winrm s winrm/config/service '@{EnableCompatibilityHttpsListener="true"}'

#RemoteShell settings#

winrm s winrm/config/winrs '@{AllowRemoteShellAccess="true"}'

winrm s winrm/config/winrs '@{MaxProcessesPerShell="2147483647"}'

winrm s winrm/config/winrs '@{MaxShellsPerUser="2147483647"}'

winrm s winrm/config/winrs '@{IdleTimeout="7200000"}'

winrm s winrm/config/service '@{AllowUnencrypted="true"}'

#Firewall Rule Creations(Inbound/Outbound

$status =netsh advfirewall show all State| Out-String

if($status.Contains("ON"))

{

New-NetFirewallRule -Name "Zenoss-WINRM-Inbound2" `

-DisplayName "Zenoss-WINRM-Inbound" `

-Description "Inbound rule for Windows Remote Management" `

-Group "Windows Remote Management" `

-Protocol TCP `

-LocalPort 88, 135, 445, 5985-5986 `

-Direction Inbound `

-Action Allow `

-Profile Domain,Private,Public

New-NetFirewallRule -Name "Zenoss-WINRM-Outbound2" `

-DisplayName "Zenoss-WINRM-Outbound" `

-Description "Outbound rule for Windows Remote Management" `

-Group "Windows Remote Management" `

-Protocol TCP `

-LocalPort 88, 135, 445, 5985-5986 `

-Direction Outbound `

-Action Allow `

-Profile Domain,Private,Public

}

1. put the private key ,which you have generated using ssh-keygen –l id\_rsa , in to a file which can be accessed from resources.xml file for authentication as windows node already have public key in .ssh folder under C:\Program Files\OpenSSH\home\WINDOWS\_USERNAME . Make sure proper permission is set to private key id\_rsa for accessing the private key from resources.xml file.
2. Update resources.xml file:

<?xml version="1.0" encoding="UTF-8"?>

<?xml version="1.0" encoding="UTF-8"?>

<project>

<node name="localhost" description="Rundeck server node" tags="" hostname="localhost" osArch="amd64" osFamily="unix" osName="Linux" osVersion="3.10.0-514.el7.x86\_64" username="rundeck"/>

<node name="Linux\_node1" username="ec2-user" hostname="54.201.92.163" ssh-key-storage-path="keys/ss/chef-key-sourav.pem"/>

<node name="win\_node1" username="winrmuser" hostname="ec2-54-203-5-102.us-west-2.compute.amazonaws.com" ssh-keypath="/tmp/id\_rsa" ssh-authentication="privateKey" ssh-key-passphrase-option="option.sshKeyPassphrase"/>

</project>

