Guided Practice: Configure Server Roles - IIS

Outcome

In this Guided Practice, you will write an Ansible playbook to configure IIS on Windows Server 2019. IIS (Internet Information Services) is a Web Server that runs on Windows and uses the Microsoft .NET platform.

Resources Needed

- VCASTLE Pod configured for the class. For this Guided Practice, we use the CentOS 8, Ubuntu 20.04 LTS and Windows 2019 Server.
- Your user needs to be able to elevate their privileges with sudo.

Level of Difficulty

Moderate

Deliverables

Deliverables are marked in red font or with a red picture border around the screenshot. Additionally, there are questions at the end. **Your username or studentid should be visible in all screenshots that you submit.**

General Considerations

Ansible should already be installed on your CentOS machine from a prior Guided Practice. Windows should have been restored to prior Network Adapter settings at the end of completing the previous Guided Practice. CentOS should already have pywinrm from a prior Guided Practice.

Adding the Windows Machine to Your Ansible Hosts File

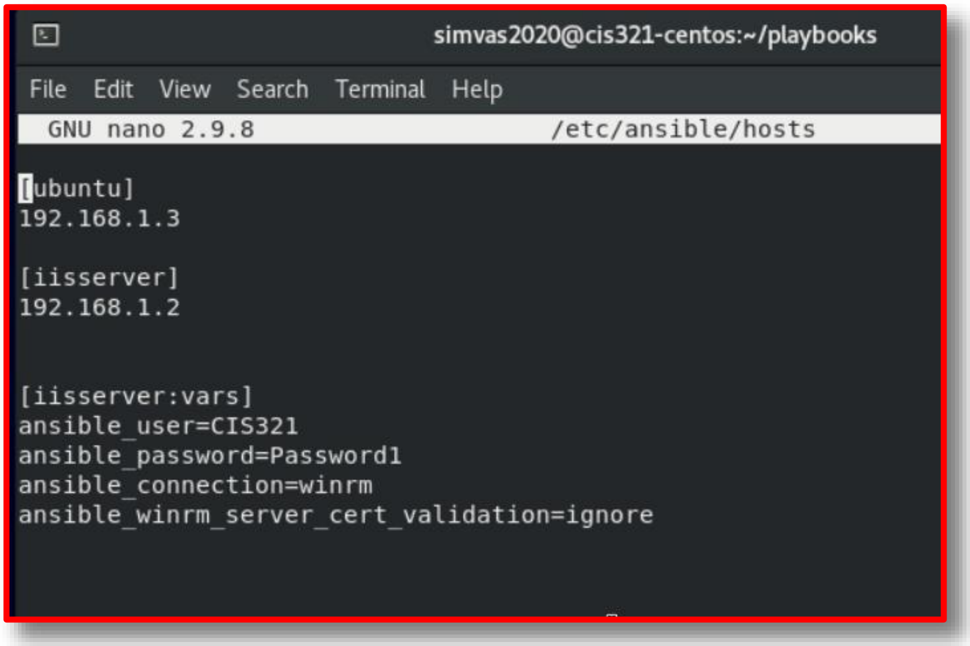
Let's check the Ansible hosts file, and make sure that your Windows 2019's computer's IP Address is assigned to one of the groups.

1. Log onto the CentOS computer using the user you created with your studentID.
2. Open the Ansible hosts file by typing:

```
sudo nano /etc/ansible/hosts
```

3. Add the Windows computer's IP Address is in one of the groupings. Make sure you use the IP address of your Windows server 2019 – in the example below, 192.168.1.2 is that address. Type:

```
[iisserver]

192.168.1.2



[iisserver:vars]

ansible_user=CIS321

ansible_password=Password1

ansible_connection=winrm

ansible_winrm_server_cert_validation=ignore
```



simvas2020@cis321-centos:~/playbooks
File Edit View Search Terminal Help

```
  GNU nano 2.9.8                    /etc/ansible/hosts

[ubuntu]
192.168.1.3

[iisserver]
192.168.1.2


[iisserver:vars]
ansible_user=CIS321
ansible_password=Password1
ansible_connection=winrm
ansible_winrm_server_cert_validation=ignore
```

Take a screenshot like the one above for your Lab Report.

Configuring an Ansible Playbook to Install IIS on Windows Server 2019

IIS (Internet Information Services) is a Web Server that runs on Windows and uses the Microsoft .NET platform. Before you begin, make sure you have completed prior Guided Practices, as this section assumes the following:

- pywinrm is installed on CentOS.
- WinRM is configured.
- The ExecutionPolicy on Windows is set to Unrestricted.

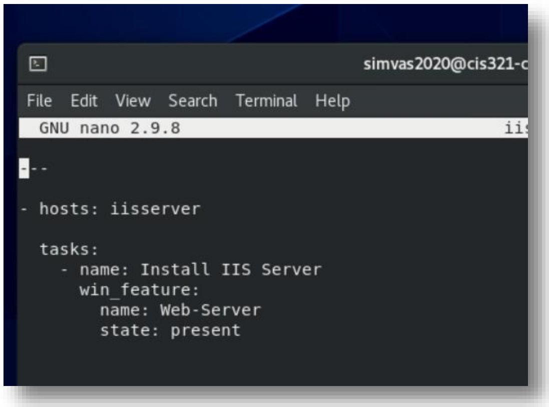
1. Now, we're going to write a playbook with just one task, which will target the Windows 2019 machine. The task will be to install IIS (Internet Information Services).

    a. Start by creating a new file called **iis.yml** under your playbooks folder. If you're using nano, type:

    ```
    cd playbooks
    sudo nano iis.yml
    ```

    b. Type:

    ```
    ---
    -  hosts: iisserver
        tasks:
          - name: Install IIS Server
            win_feature:
              name: Web-Server
              state: present
    ```



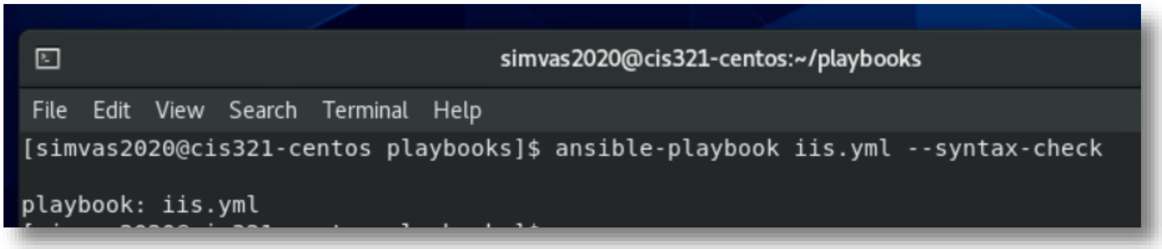
```
 GNU nano 2.9.8                         iis
---

- hosts: iisserver

  tasks:
    - name: Install IIS Server
      win_feature:
        name: Web-Server
        state: present
```

    c. Type **Ctrl+S** to save the playbook and **Ctrl+X** to exit nano

2. Check the syntax of your playbook, and correct any syntax errors before moving forward. Type:

```
ansible-playbook iis.yml –syntax-check
```

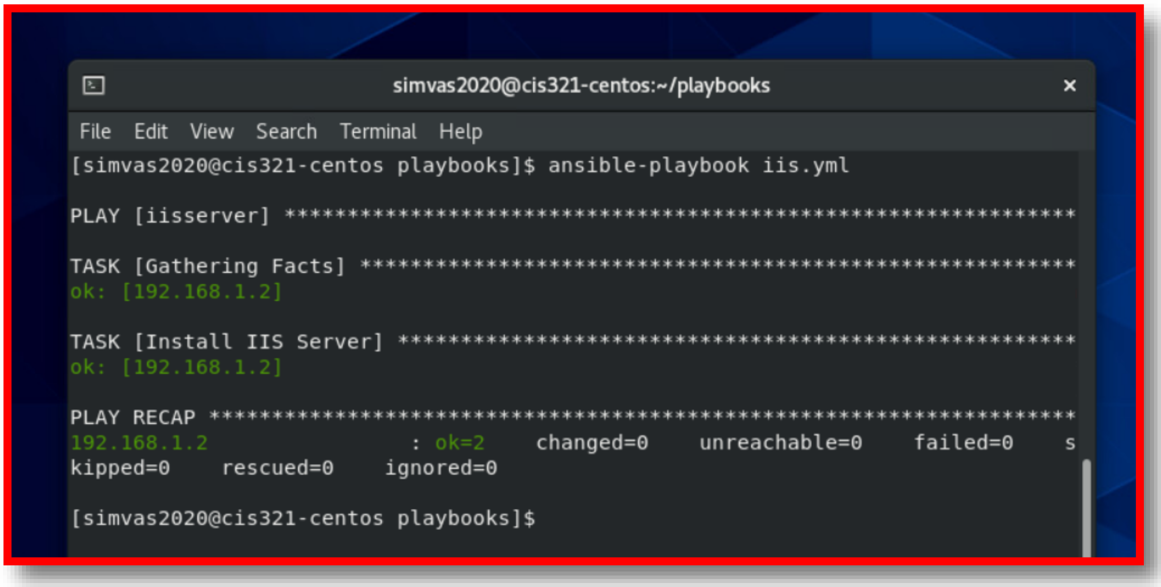
If your syntax is correct, your output should resemble the one below:


simvas2020@cis321-centos:~/playbooks
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos playbooks]$ ansible-playbook iis.yml --syntax-check

playbook: iis.yml
```

3. Run your playbook by typing:

```
ansible-playbook iis.yml
```


simvas2020@cis321-centos:~/playbooks
×
File Edit View Search Terminal Help

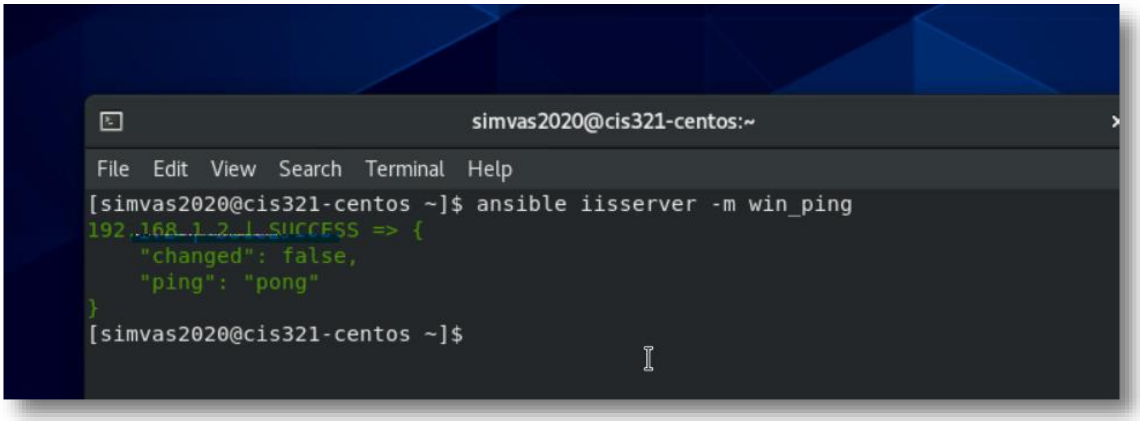
```
[simvas2020@cis321-centos playbooks]$ ansible-playbook iis.yml

PLAY [iisserver] ***************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.1.2]

TASK [Install IIS Server] ******************************************************
ok: [192.168.1.2]

PLAY RECAP *********************************************************************
192.168.1.2                : ok=2    changed=0    unreachable=0    failed=0    s
kipped=0    rescued=0    ignored=0

[simvas2020@cis321-centos playbooks]$
```

Take a screenshot like the one above for your Lab Report.

4. If you cannot run the playbook, type:

```
ansible iisserver -m win_ping
```

and verify you can connect:

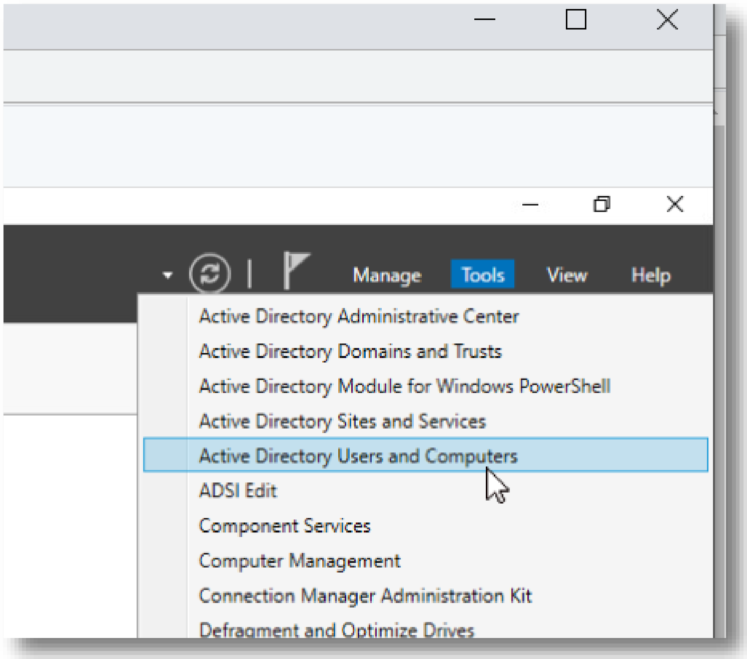

simvas2020@cis321-centos:~
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos ~]$ ansible iisserver -m win_ping
192.168.1.2 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
[simvas2020@cis321-centos ~]$
```

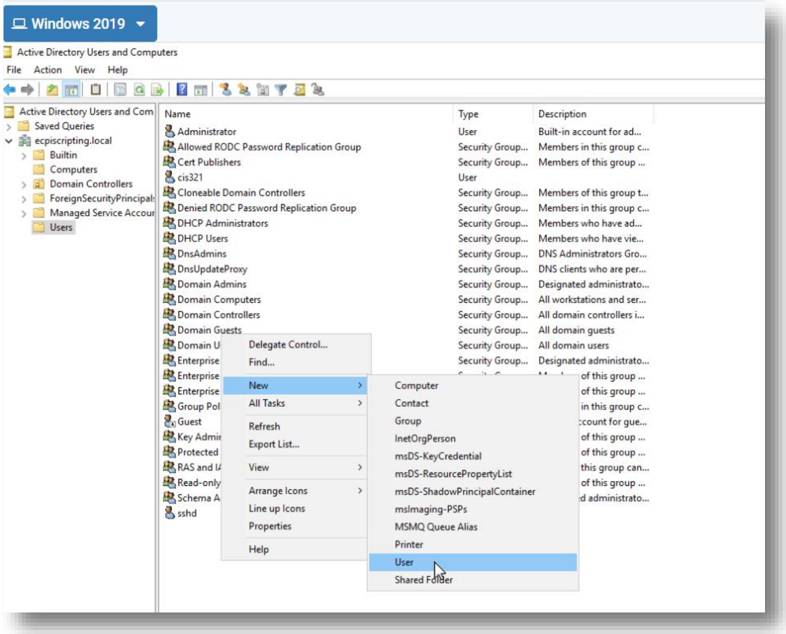
Copying a SSH Public Key on Windows to Allow Password Less Login

While we used WinRM to be able to run playbooks from the CentOS Machine to the Windows machine, we will also configure password less login for SSH to allow us to more easily connect to the Windows machine and copy files.

1. Log in to the Windows 2019 machine as the Administrator user using Password1.
2. In Server Manager click on **Tools → Active Directory Users and Computers**.

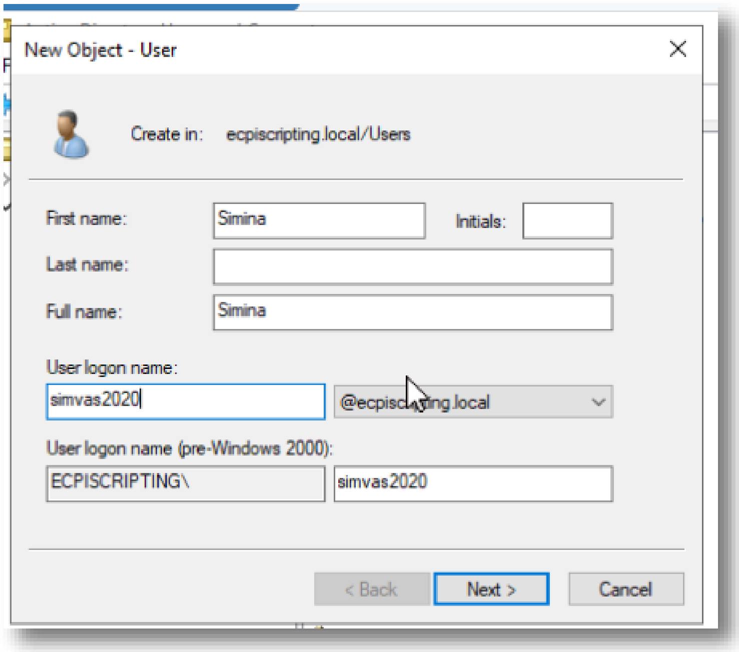

Manage
Tools
View
Help
Active Directory Administrative Center
Active Directory Domains and Trusts
Active Directory Module for Windows PowerShell
Active Directory Sites and Services
Active Directory Users and Computers
ADSI Edit
Component Services
Computer Management
Connection Manager Administration Kit
Defragment and Optimize Drives

3. Click on **Users**, right-click in the window, and select **New→ User**.



Name	Type	Description
Administrator	User	Built-in account for ad...
Allowed RODC Password Replication Group	Security Group...	Members in this group c...
Cert Publishers	Security Group...	Members of this group ...
cis321	User	
Cloneable Domain Controllers	Security Group...	Members of this group t...
Denied RODC Password Replication Group	Security Group...	Members in this group c...
DHCP Administrators	Security Group...	Members who have ad...
DHCP Users	Security Group...	Members who have vie...
DnsAdmins	Security Group...	DNS Administrators Gro...
DnsUpdateProxy	Security Group...	DNS clients who are per...
Domain Admins	Security Group...	Designated administrato...
Domain Computers	Security Group...	All workstations and ser...
Domain Controllers	Security Group...	All domain controllers i...
Domain Guests	Security Group...	All domain guests
Domain U	Security Group...	All domain users
Enterprise	Security Group...	Designated administrato...
Enterprise		of this group ...
Enterprise		of this group ...
Group Pol		in this group c...
Guest		:count for gue...
Key Admir		of this group ...
Protected		of this group ...
RAS and I/		this group can...
Read-only		of this group ...
Schema A		d administrato...
sshd		

4. Create a new user using your studentID.



Create in: ecpiscripting.local/Users

First name: Simina

Initials:

Last name:

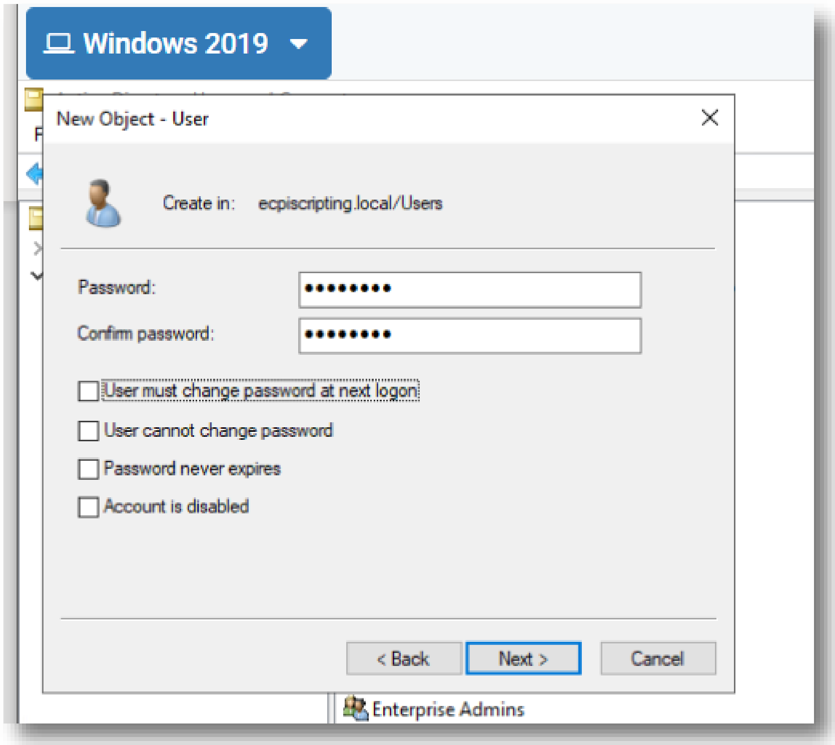
Full name: Simina

User logon name:

simvas2020 @ecpiscripting.local

User logon name (pre-Windows 2000):

ECPISCRIPTING\ simvas2020



Windows 2019

New Object - User

Create in: ecpiscripting.local/Users

Password: ••••••••

Confirm password: ••••••••

□ User must change password at next logon

□ User cannot change password

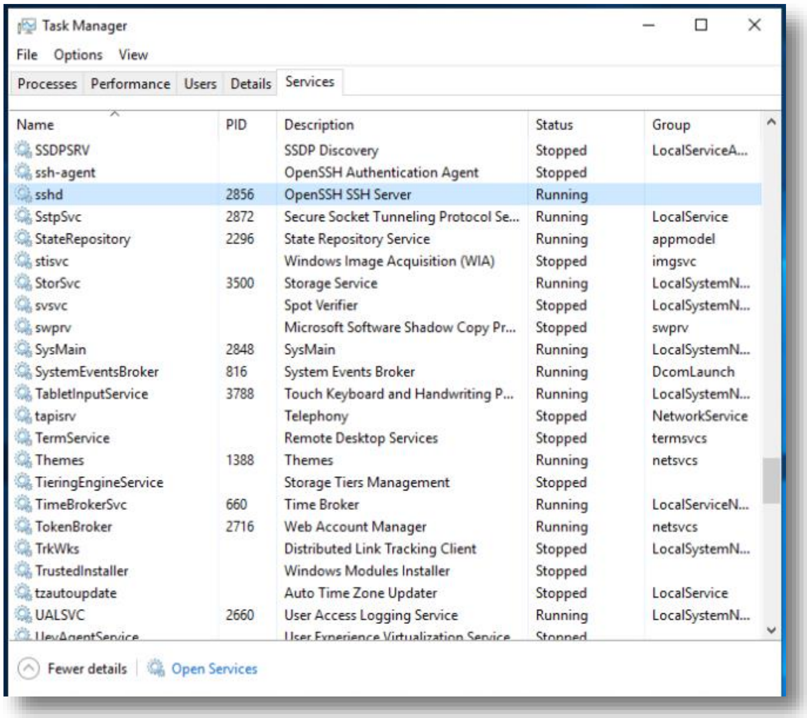
□ Password never expires

□ Account is disabled

< Back | Next > | Cancel

Enterprise Admins

5. Let us verify that OpenSSH is installed and running on Windows:
    a. Open Task Manager.
    b. Click on **More Details** at the bottom of the Task Manager Window.
    c. Click on the **Services** tab.
    d. Scroll down until you find an entry for sshd.



Task Manager

File Options View

Processes | Performance | Users | Details | Services

Name	PID	Description	Status	Group
SSDPSRV		SSDP Discovery	Stopped	LocalServiceA...
ssh-agent		OpenSSH Authentication Agent	Stopped	
sshd	2856	OpenSSH SSH Server	Running	
SstpSvc	2872	Secure Socket Tunneling Protocol Se...	Running	LocalService
StateRepository	2296	State Repository Service	Running	appmodel
stisvc		Windows Image Acquisition (WIA)	Stopped	imgsvc
StorSvc	3500	Storage Service	Running	LocalSystemN...
svsvc		Spot Verifier	Stopped	LocalSystemN...
swprv		Microsoft Software Shadow Copy Pr...	Stopped	swprv
SysMain	2848	SysMain	Running	LocalSystemN...
SystemEventsBroker	816	System Events Broker	Running	DcomLaunch
TabletInputService	3788	Touch Keyboard and Handwriting P...	Running	LocalSystemN...
tapisrv		Telephony	Stopped	NetworkService
TermService		Remote Desktop Services	Stopped	termsvcs
Themes	1388	Themes	Running	netsvcs
TieringEngineService		Storage Tiers Management	Stopped	
TimeBrokerSvc	660	Time Broker	Running	LocalServiceN...
TokenBroker	2716	Web Account Manager	Running	netsvcs
TrkWks		Distributed Link Tracking Client	Stopped	LocalSystemN...
TrustedInstaller		Windows Modules Installer	Stopped	
tzautoupdate		Auto Time Zone Updater	Stopped	LocalService
UALSVC	2660	User Access Logging Service	Running	LocalSystemN...
UevAgentService		User Experience Virtualization Service	Stopped	

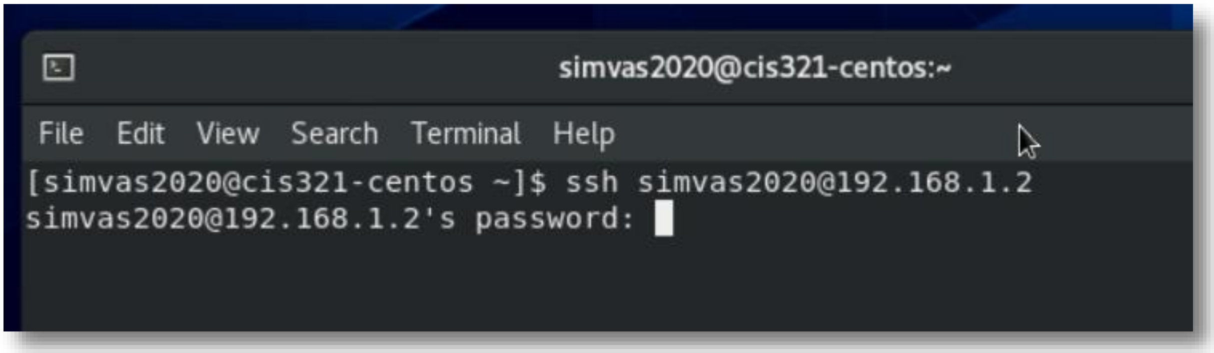
Fewer details | Open Services

6. Log in to CentOS with your user, and open a Terminal.
7. Check to see that you can log in via SSH to the Windows machine. Type:

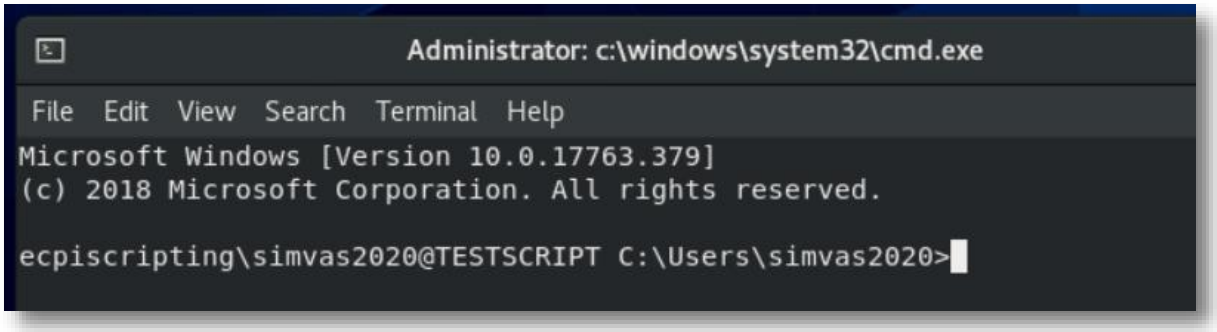
```
ssh yourUserName@WindowsIPAddress
```

   As an example, the user created here has the username simvas2020, and the IP address for Windows 2019 is 192.168.1.2. Therefore, the command was:

```
ssh simvas2020@192.168.1.2
```


simvas2020@cis321-centos:~
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos ~]$ ssh simvas2020@192.168.1.2
simvas2020@192.168.1.2's password:
```


Administrator: c:\windows\system32\cmd.exe
File Edit View Search Terminal Help

```
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

ecpiscripting\simvas2020@TESTSCRIPT C:\Users\simvas2020>
```

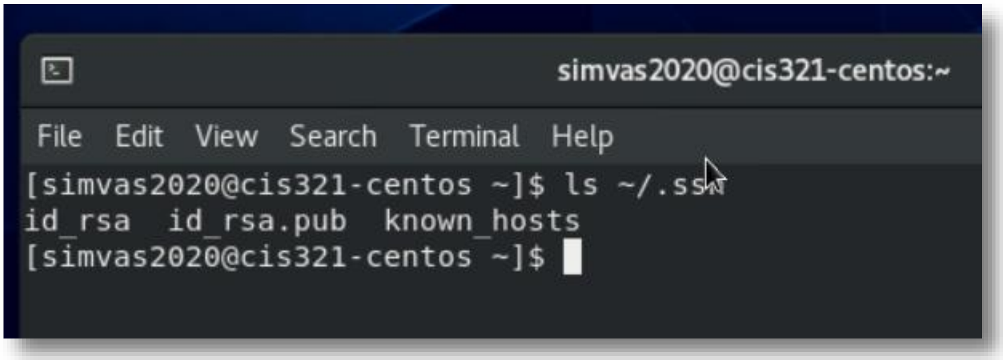
8. Type **exit** to close the connection.

```
exit
```

9. Now let us copy the ssh public key to Windows, so that we can ssh from CentOS to Windows without using a password.
   a. In the CentOS Terminal, type:

```
ls ~/.ssh
```

   You should see the contents of your .ssh folder similar to the one below. You have generated a ssh key-pair in a prior activity.


simvas2020@cis321-centos:~
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos ~]$ ls ~/.ssh
id_rsa  id_rsa.pub  known_hosts
[simvas2020@cis321-centos ~]$ 
```

b. We need to copy the id_rsa.pub key to the appropriate location on Windows. Previously, we used ssh-copy-id, but this will not work for Windows. Type:

```
sftp yourUserName@WindowsIPAddress
```

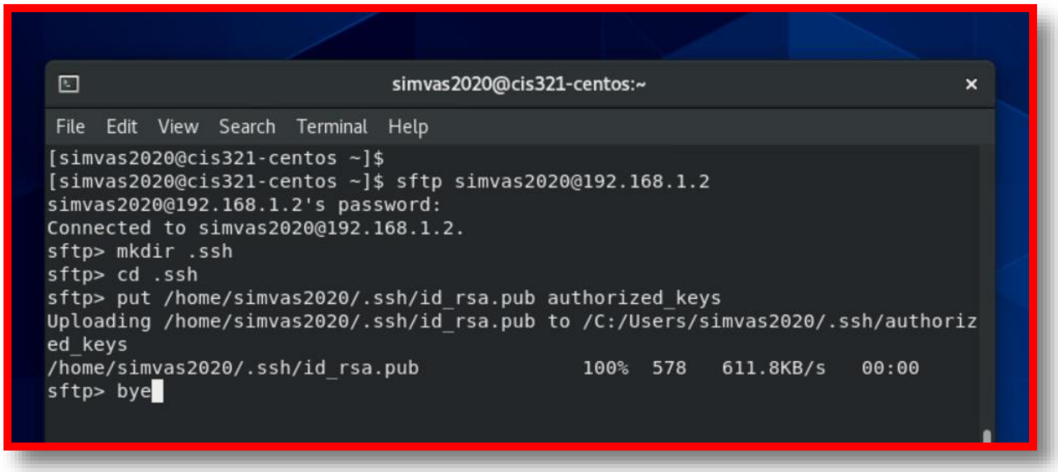
As an example, the user created here has the username simvas2020, and the IP address for Windows 2019 is 192.168.1.2. Therefore, the command was:

```
sftp simvas2020@192.168.1.2
```

c. Enter your password when prompted.

d. Type:

```
mkdir .ssh
cd .ssh
put /home/yourHomeFolder/.ssh/id_rsa.pub authorized_keys
```


simvas2020@cis321-centos:~
×
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos ~]$
[simvas2020@cis321-centos ~]$ sftp simvas2020@192.168.1.2
simvas2020@192.168.1.2's password:
Connected to simvas2020@192.168.1.2.
sftp> mkdir .ssh
sftp> cd .ssh
sftp> put /home/simvas2020/.ssh/id_rsa.pub authorized_keys
Uploading /home/simvas2020/.ssh/id_rsa.pub to /C:/Users/simvas2020/.ssh/authoriz
ed_keys
/home/simvas2020/.ssh/id_rsa.pub              100%  578   611.8KB/s   00:00
sftp> bye
```

Take a screenshot like the one above for your Lab Report.

e. Type the following to close the sftp connection:

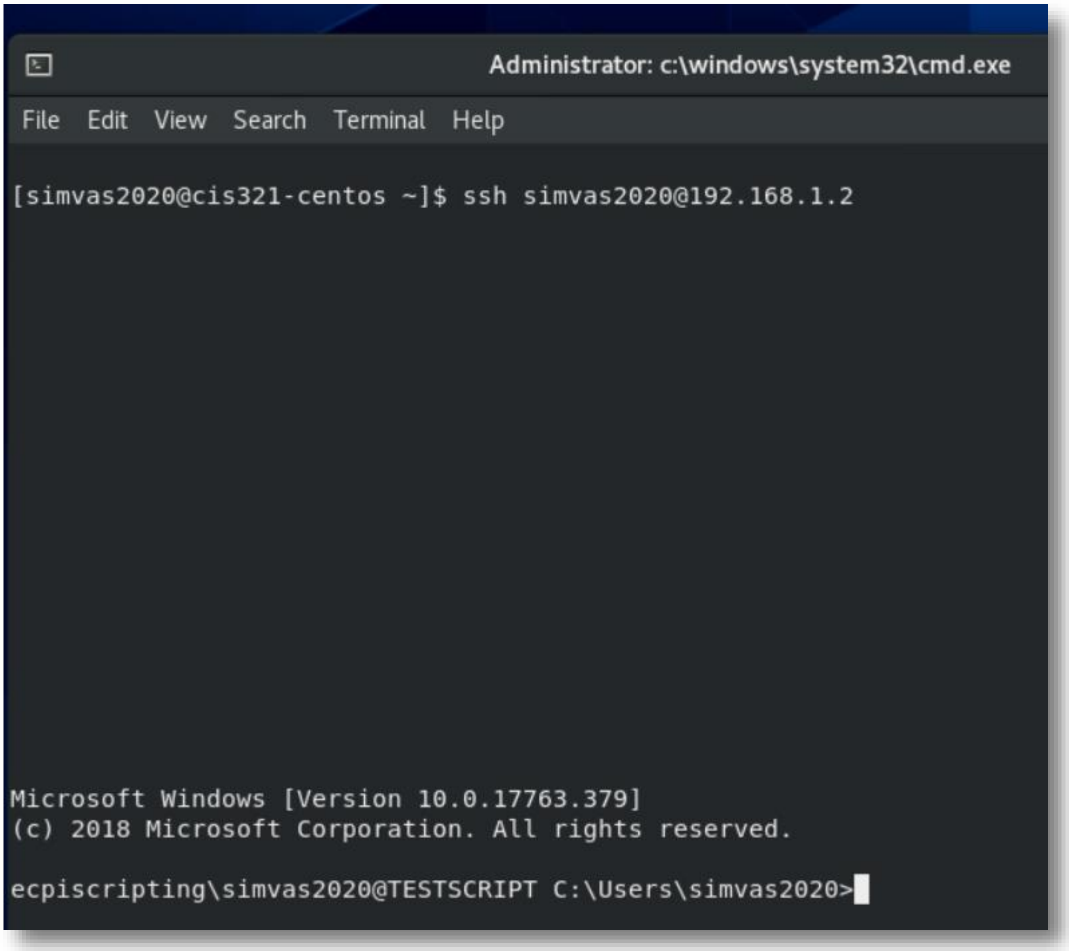
```
bye
```

f. Check that you can ssh into the Windows machine without the need of entering a password. Type:

```
ssh yourUserName@WindowsIPAddress
```

As an example, the user created here has the username simvas2020, and the IP address for Windows 2019 is 192.168.1.2. Therefore, the command was:

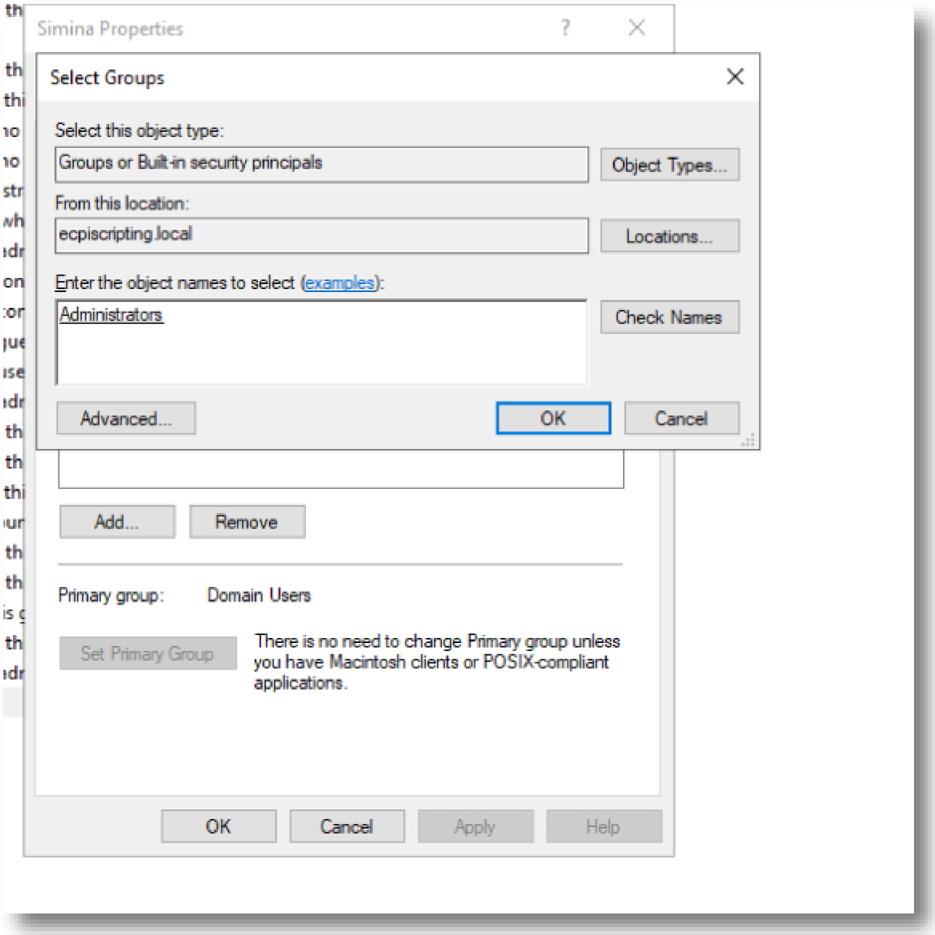
```
 ssh simvas2020@192.168.1.2
```


Administrator: c:\windows\system32\cmd.exe
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos ~]$ ssh simvas2020@192.168.1.2

Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

ecpiscripting\simvas2020@TESTSCRIPT C:\Users\simvas2020>
```

10. This works well for a regular user, but we need to add the user to the Administrators group in order to be able to run ansible playbooks on Windows. Additional steps will be necessary in order to allow a user from the Administrators group to login without a password.
11. Let us add the user to the Administrators group.
    a. Exit from your existing connection, and then reconnect.

b. Open Active Directory Users and Groups.
c. Expand **Users.**
d. **Right click** on your user and select **Properties.**
e. Click on the **Member Of** tab.
f. Type **Administrators** in "Enter the object names to select" box of the window that opens.



Select this object type:
Groups or Built-in security principals

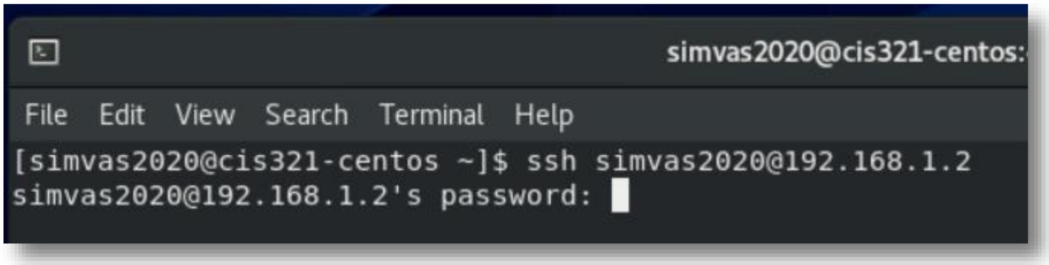
From this location:
ecpiscripting.local

Enter the object names to select (examples):
Administrators

Primary group: Domain Users

There is no need to change Primary group unless you have Macintosh clients or POSIX-compliant applications.

g. Click Check Names.
h. Click **OK.**
i. Once added to the Administrators group, you will no longer be able to SSH to Windows without being prompted for a password – go ahead and try to.

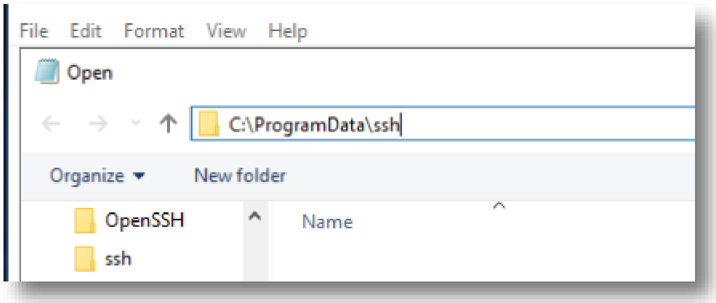


simvas2020@cis321-centos:
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos ~]$ ssh simvas2020@192.168.1.2
simvas2020@192.168.1.2's password: 
```

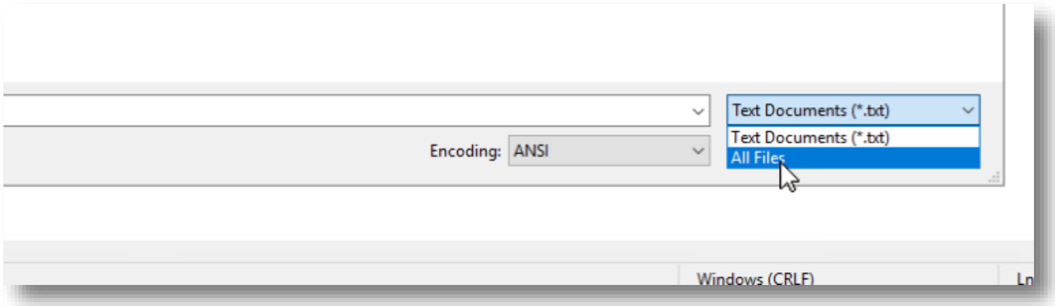
12. Let us examine sshd_config:

a. Open Notepad.
b. Click on **File → Open**.
c. Type **C:\ProgramData\ssh** is the navigation bar of the window that opens.



C:\ProgramData\ssh

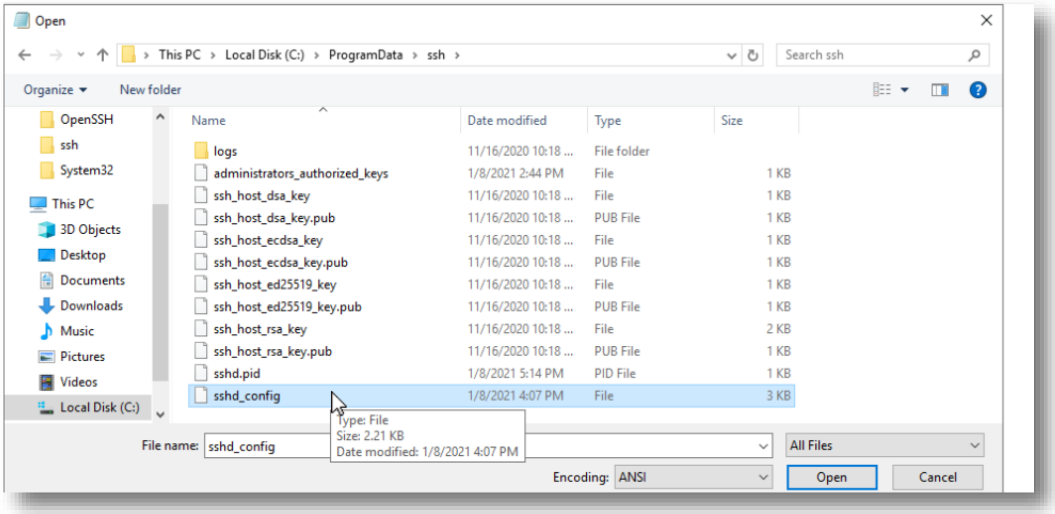
d. Select **All Files** in the filter box.



Encoding: ANSI

All Files

e. Select **sshd_config** and double click to open it.



Name	Date modified	Type	Size
logs	11/16/2020 10:18 ...	File folder	
administrators_authorized_keys	1/8/2021 2:44 PM	File	1 KB
ssh_host_dsa_key	11/16/2020 10:18 ...	File	1 KB
ssh_host_dsa_key.pub	11/16/2020 10:18 ...	PUB File	1 KB
ssh_host_ecdsa_key	11/16/2020 10:18 ...	File	1 KB
ssh_host_ecdsa_key.pub	11/16/2020 10:18 ...	PUB File	1 KB
ssh_host_ed25519_key	11/16/2020 10:18 ...	File	1 KB
ssh_host_ed25519_key.pub	11/16/2020 10:18 ...	PUB File	1 KB
ssh_host_rsa_key	11/16/2020 10:18 ...	File	2 KB
ssh_host_rsa_key.pub	11/16/2020 10:18 ...	PUB File	1 KB
sshd.pid	1/8/2021 5:14 PM	PID File	1 KB
sshd_config	1/8/2021 4:07 PM	File	3 KB

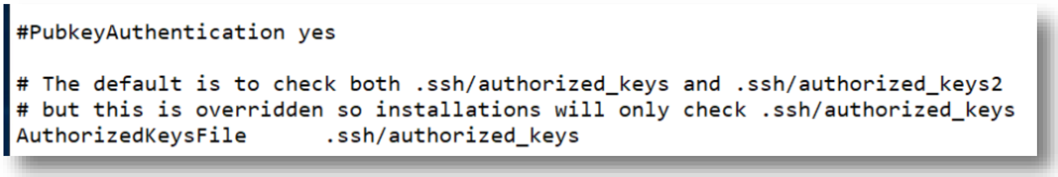
Type: File
Size: 2.21 KB
Date modified: 1/8/2021 4:07 PM

File name: sshd_config

All Files

Encoding: ANSI

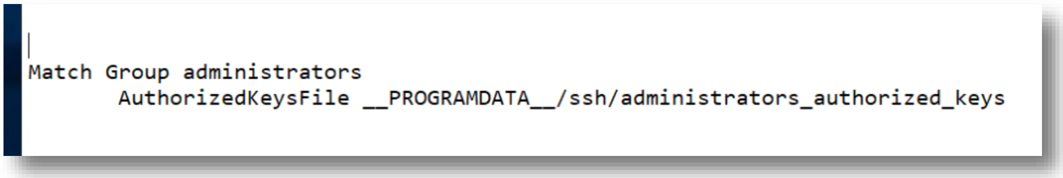
f. Scroll to the line that says **AuthorizedKeysFile.** This is the location where we placed the public ssh key for our user.



```
#PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      .ssh/authorized_keys
```

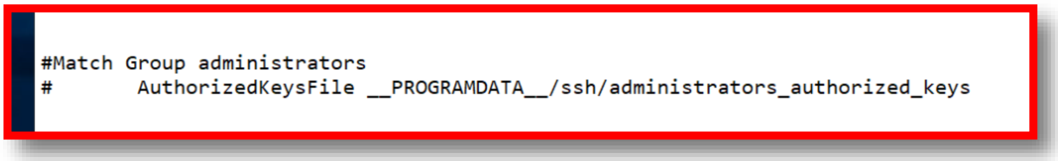
g. Scroll to the end of the file, and you will notice a few more lines.

```
 Match Group administrators

 AuthorizedKeysFile
__ProgramData__/sshadministrators_authorized_key
```



```
|
Match Group administrators
       AuthorizedKeysFile __PROGRAMDATA__/ssh/administrators_authorized_keys
```

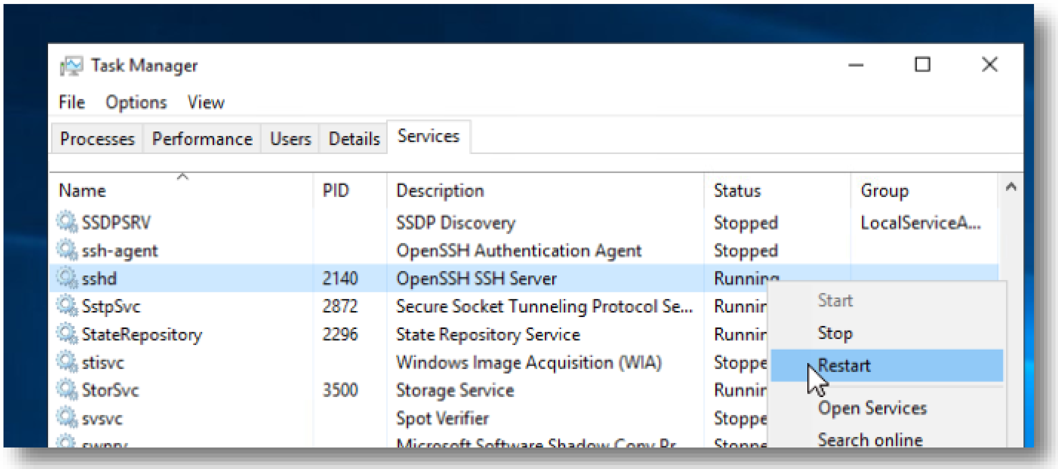
h. These two lines tell OpenSSH that the keys for users from the Administrators group are placed in a different location then we have placed them. There are multiple ways to deal with this, but we will go ahead and comment these two lines. Place a # at the beginning of each line. Then save and close the file.



```
#Match Group administrators
#       AuthorizedKeysFile __PROGRAMDATA__/ssh/administrators_authorized_keys
```

Take a screenshot like the one above for your Lab Report.

i. Restart the SSH service so that the configuration changes take effect. Find sshd under the **Services** tab in Task Manager, right-click in the **Status** column (current status should be Running), and select **Restart.**



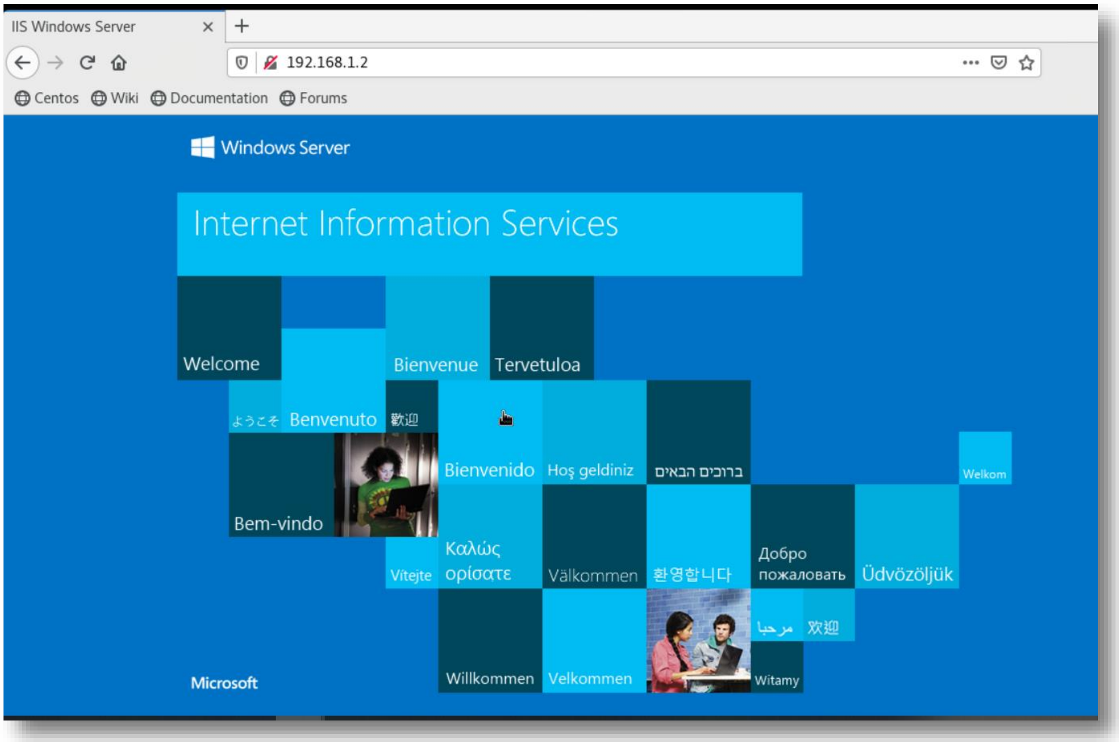
Name	PID	Description	Status	Group
SSDPSRV		SSDP Discovery	Stopped	LocalServiceA...
ssh-agent		OpenSSH Authentication Agent	Stopped	
sshd	2140	OpenSSH SSH Server	Running	
SstpSvc	2872	Secure Socket Tunneling Protocol Se...	Runnir	
StateRepository	2296	State Repository Service	Runnir	
stisvc		Windows Image Acquisition (WIA)	Stoppe	
StorSvc	3500	Storage Service	Runnir	
svsvc		Spot Verifier	Stoppe	
swprv		Microsoft Software Shadow Copy Pr...	Stoppe	

j. Go back to CentOs and try to SSH to the windows machine using your user. You will no longer be prompted for a password. Type exit to close your ssh connection.

Checking You Can Access the IIS Website

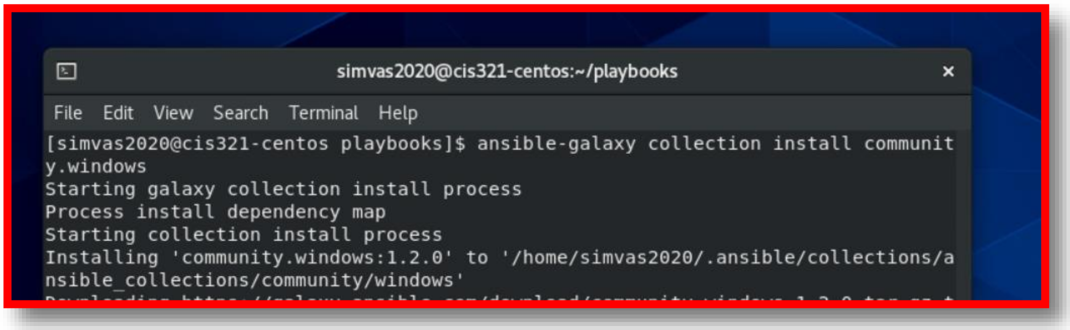
1. Go to CentOS, open a browser page, and type the following. The default website should be displayed.

```
192.168.1.2
```



2. Create a playbook to stop the default website.
   a. First, install the community.windows collection. Type:

```
ansible-galaxy collection install community.windows
```

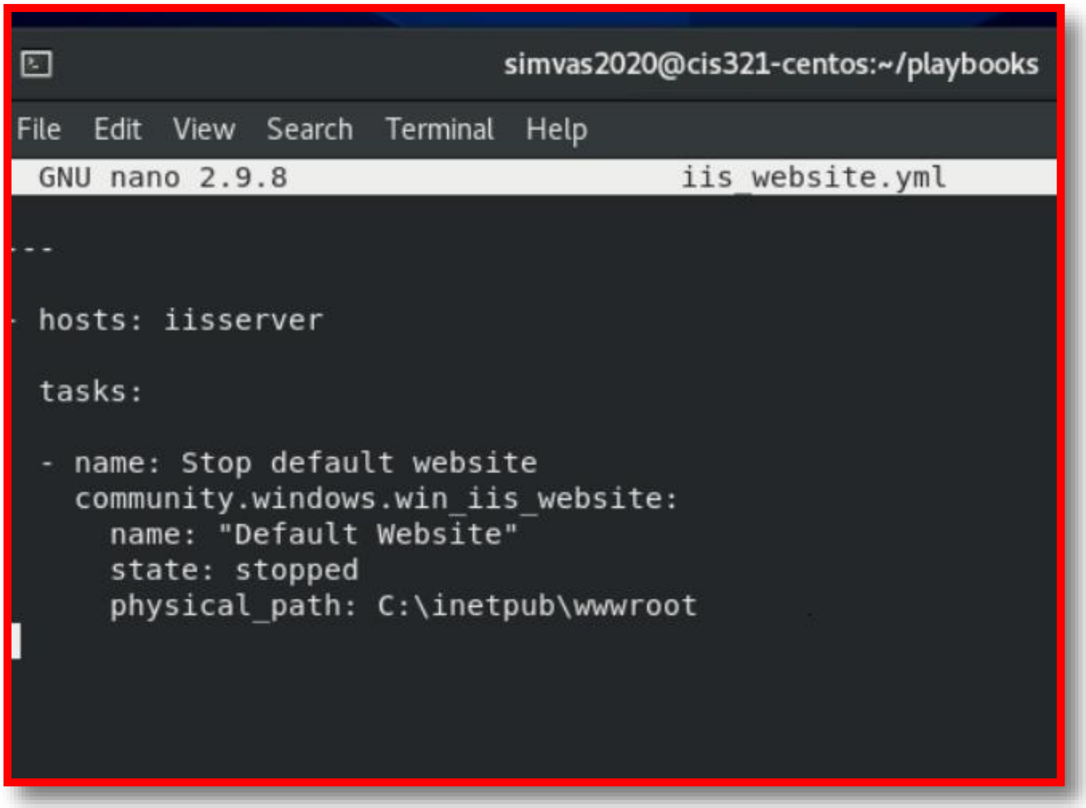

simvas2020@cis321-centos:~/playbooks
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos playbooks]$ ansible-galaxy collection install communit
y.windows
Starting galaxy collection install process
Process install dependency map
Starting collection install process
Installing 'community.windows:1.2.0' to '/home/simvas2020/.ansible/collections/a
nsible_collections/community/windows'
```

Take a screenshot like the one above for your Lab Report.

3. Inside your playbooks directory, create a file called **iis_website.yml**, and add to it the following lines:

```
---

- hosts: iisserver

  tasks:

  - name: Stop default website
    community.windows.win_iis_website:
      name: "Default Website"
      state: stopped
      physical_path: C:\inetpub\wwwroot
```

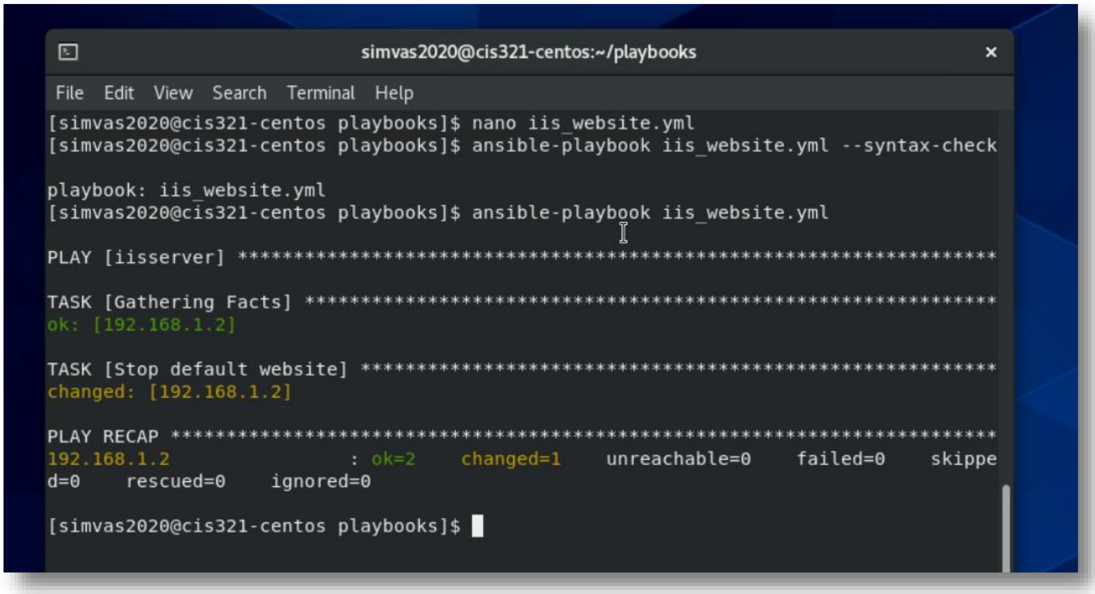


simvas2020@cis321-centos:~/playbooks
File Edit View Search Terminal Help
GNU nano 2.9.8 iis_website.yml

```
---

- hosts: iisserver

  tasks:

  - name: Stop default website
    community.windows.win_iis_website:
      name: "Default Website"
      state: stopped
      physical_path: C:\inetpub\wwwroot
```

Take a screenshot like the one above for your Lab Report.

4. Save the playbook, check its syntax, correct any errors, and run it.

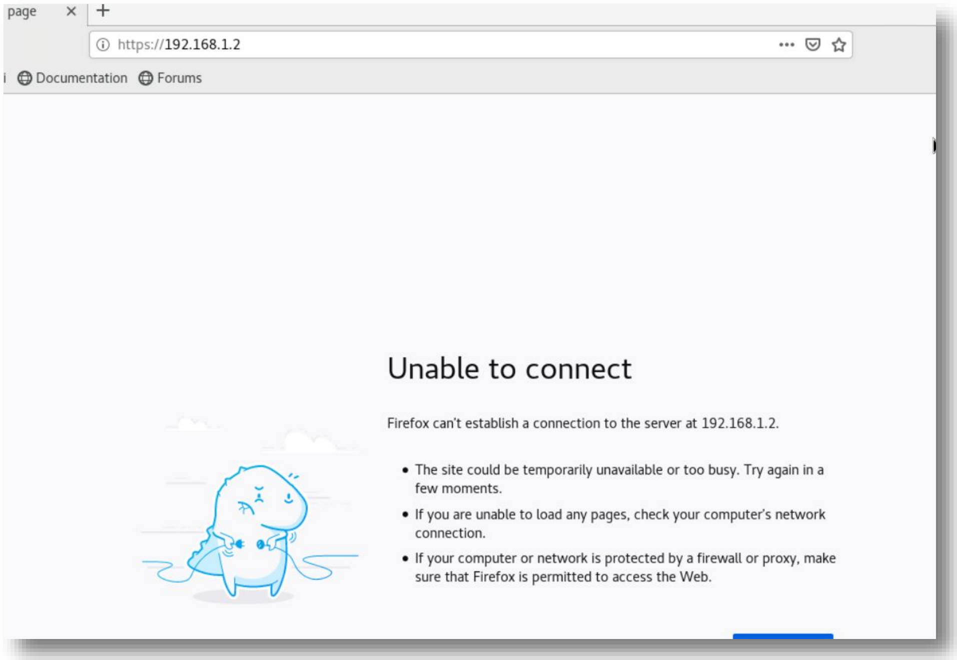


simvas2020@cis321-centos:~/playbooks
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos playbooks]$ nano iis_website.yml
[simvas2020@cis321-centos playbooks]$ ansible-playbook iis_website.yml --syntax-check

playbook: iis_website.yml
[simvas2020@cis321-centos playbooks]$ ansible-playbook iis_website.yml

PLAY [iisserver] ********************************************************************

TASK [Gathering Facts] **************************************************************
ok: [192.168.1.2]

TASK [Stop default website] *********************************************************
changed: [192.168.1.2]

PLAY RECAP **************************************************************************
192.168.1.2                : ok=2    changed=1    unreachable=0    failed=0    skippe
d=0    rescued=0    ignored=0

[simvas2020@cis321-centos playbooks]$ 
```

5. Check that you can no longer access the website. Go to CentOS, open a browser page, and type the following. You should not be able to connect. **Note: You may need to clear Firefox history to see this change.**

```
192.168.1.2
```



Unable to connect

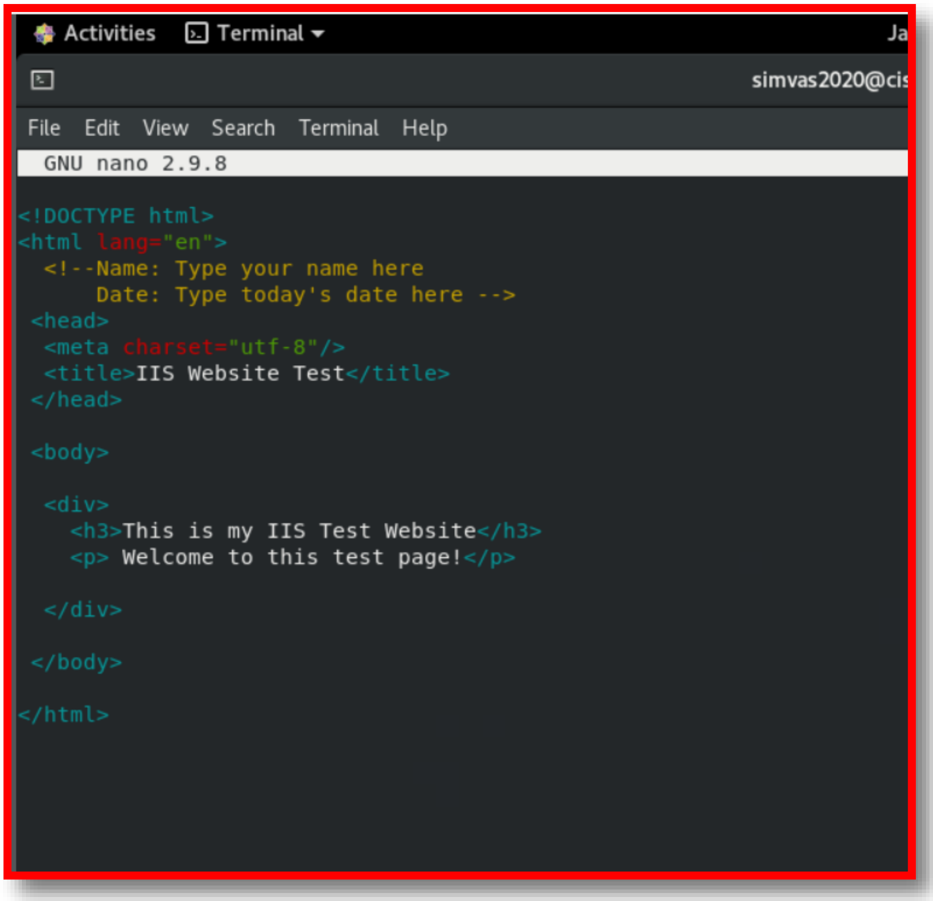
Firefox can't establish a connection to the server at 192.168.1.2.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

Create a Test Webpage

1. On CentOS, use a text editor to create a file **index.html,** and type the following content. After you have typed the lines below, save the file, and close the editor.

```
<!DOCTYPE html>
<html lang="en">
  <!--Name: Type your name here
      Date: Type today's date here -->
 <head>
  <meta charset="utf-8"/>
  <title>IIS Website Test</title>
 </head>

 <body>

  <div>
    <h3>This is my IIS Test Website</h3>
    <p> Welcome to this test page!</p>

  </div>

 </body>

</html>
```



```
 GNU nano 2.9.8

<!DOCTYPE html>
<html lang="en">
  <!--Name: Type your name here
      Date: Type today's date here -->
 <head>
  <meta charset="utf-8"/>
  <title>IIS Website Test</title>
 </head>

 <body>

  <div>
    <h3>This is my IIS Test Website</h3>
    <p> Welcome to this test page!</p>

  </div>

 </body>

</html>
```

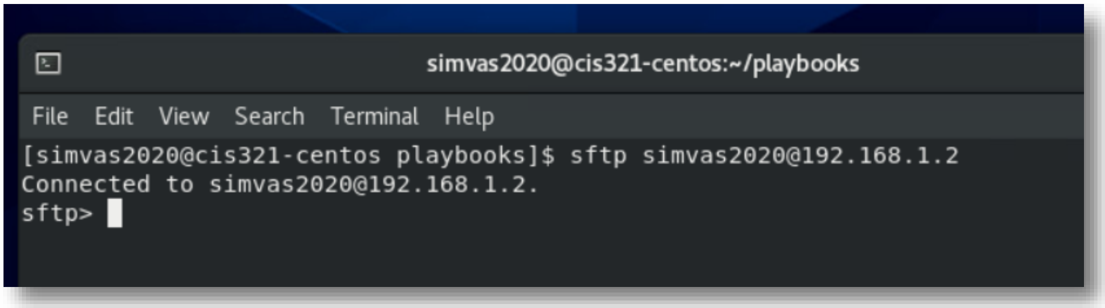
Take a screenshot like the one above for your Lab Report.

2. Given that you configured password less ssh earlier in this Guided Practice, you should easily be able to connect to the Windows machine. In a terminal, type:

```
sftp yourusername@windowsIPaddress
```

For example, in the screenshot below we used:

```
sftp simvas2020@192.168.1.2
```


simvas2020@cis321-centos:~/playbooks
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos playbooks]$ sftp simvas2020@192.168.1.2
Connected to simvas2020@192.168.1.2.
sftp> 
```

3. Type:

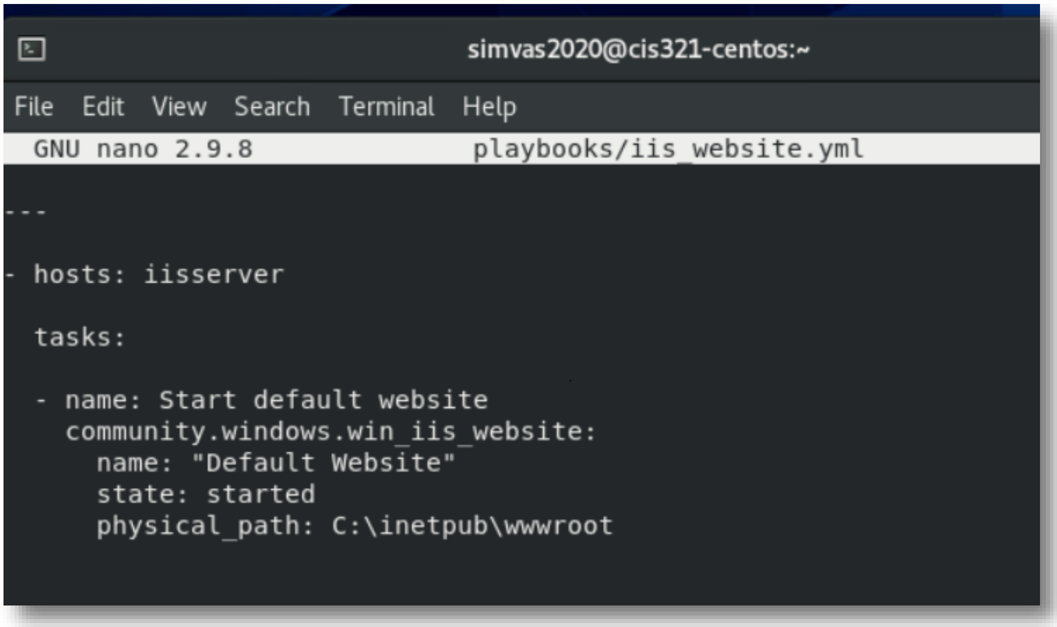
```
put index.html /C:/inetpub/wwwroot
```

after you copy index.html. At the sftp prompt, type:

```
quit
```

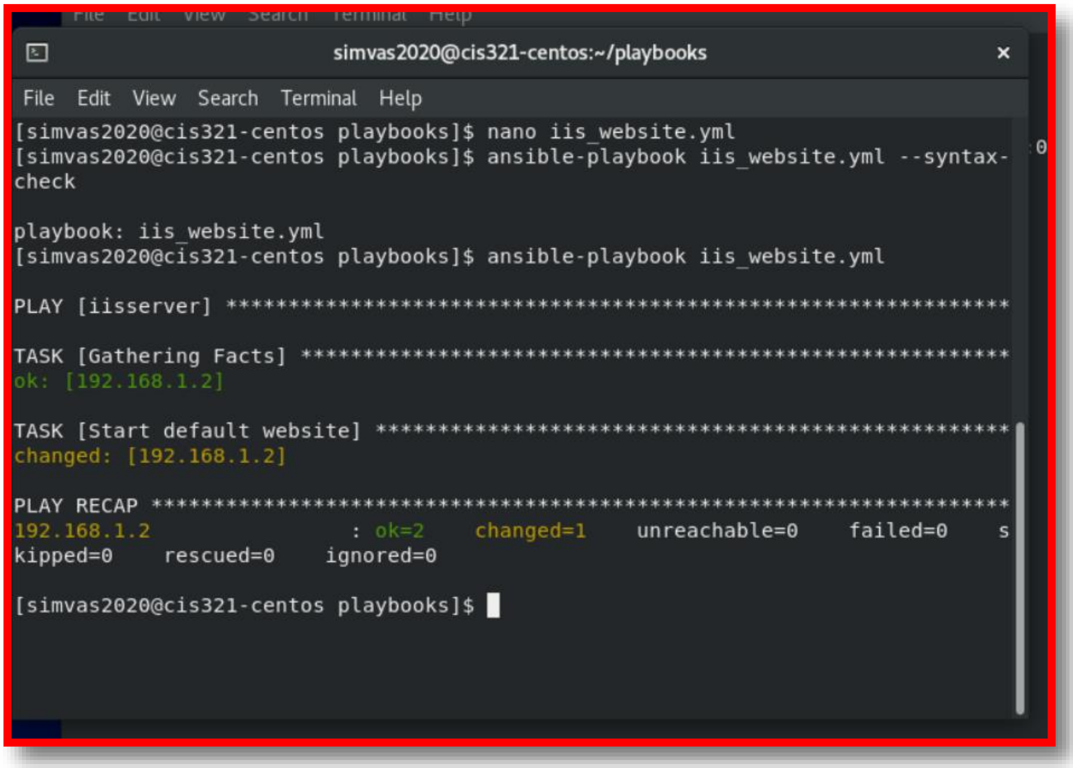
4. Change your iis_website.yml playbook to start the default website again. Edit iis_website.yml, and change name: **Stop default website** to name: **Start default website** and state: **stopped** to state: **started**.

```
---


- hosts: iisserver


  tasks:


  - name: Start default website
    community.windows.win_iis_website:
      name: "Default Website"
      state: started
      physical_path: C:\inetpub\wwwroot
```


simvas2020@cis321-centos:~
File Edit View Search Terminal Help

```
  GNU nano 2.9.8              playbooks/iis_website.yml

---

- hosts: iisserver

  tasks:

  - name: Start default website
    community.windows.win_iis_website:
      name: "Default Website"
      state: started
      physical_path: C:\inetpub\wwwroot
```

6. Save the playbook, check its syntax, correct any errors, and run it.

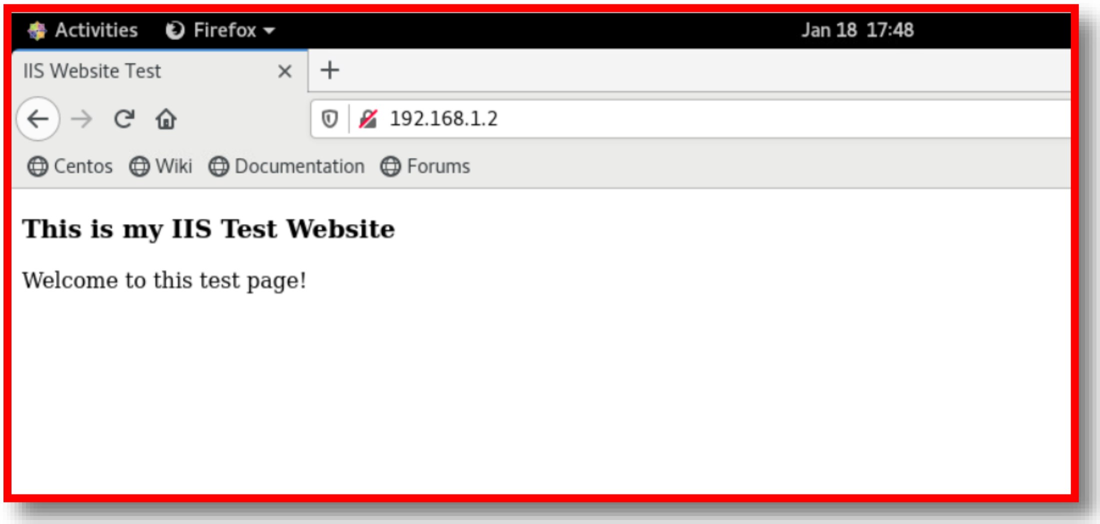

simvas2020@cis321-centos:~/playbooks
File Edit View Search Terminal Help

```
[simvas2020@cis321-centos playbooks]$ nano iis_website.yml
[simvas2020@cis321-centos playbooks]$ ansible-playbook iis_website.yml --syntax-
check

playbook: iis_website.yml
[simvas2020@cis321-centos playbooks]$ ansible-playbook iis_website.yml

PLAY [iisserver] ***************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.1.2]

TASK [Start default website] ***************************************************
changed: [192.168.1.2]

PLAY RECAP *********************************************************************
192.168.1.2                : ok=2    changed=1    unreachable=0    failed=0    s
kipped=0    rescued=0    ignored=0

[simvas2020@cis321-centos playbooks]$ 
```

Take a screenshot like the one above for your Lab Report.

7. Go to CentOS, open a browser page, and type the following. The new website should be displayed. **Note: You may need to clear Firefox history to see this change.**

```
192.168.1.2
```



This is my IIS Test Website

Welcome to this test page!

Take a screenshot like the one above for your Lab Report.