

# PASSEXAM 問題集

更に上のクオリティ 更に上のサービス



1年で無料進級することに提供する  
<http://www.passexam.jp>

**Exam** : **EX417**

**Title** : Red Hat Certified Specialist  
in Microsoft Windows  
Automation with Ansible  
Exam

**Version** : DEMO

1.You are setting up your Ansible Automation Platform environment to manage Windows systems. Create a static inventory file named windows\_hosts.ini under /home/ansible/inventories/, define a group called [win\_servers] with two Windows hosts (winhost1 and winhost2) using their IP addresses. Configure connection variables such as ansible\_connection, ansible\_user, and ansible\_password, assuming both systems support WinRM over HTTP.

A. See the Explanation.

**Answer: A**

**Explanation:**

1. Create the inventory directory if it doesn't exist: `mkdir -p /home/ansible/inventories`

2. Create the inventory file:

```
nano /home/ansible/inventories/windows_hosts.ini
```

3. Add the following content:

```
[win_servers]
```

```
winhost1 ansible_host=192.168.1.10 ansible_user=Administrator ansible_password='P@ssw0rd!' ansible_connection=winrm ansible_winrm_transport=basic ansible_port=5985
```

```
winhost2 ansible_host=192.168.1.11 ansible_user=Administrator ansible_password='P@ssw0rd!' ansible_connection=winrm ansible_winrm_transport=basic ansible_port=5985
```

```
winhost2 ansible_host=192.168.1.11 ansible_user=Administrator ansible_password='P@ssw0rd!' ansible_connection=winrm ansible_winrm_transport=basic ansible_port=5985
```

```
winhost2 ansible_host=192.168.1.11 ansible_user=Administrator ansible_password='P@ssw0rd!' ansible_connection=winrm ansible_winrm_transport=basic ansible_port=5985
```

4. Save and close the file. Ansible will now be able to use this inventory to manage Windows hosts.

2.You need to verify connectivity to the Windows hosts defined in your static inventory file using the win\_ping module. Describe the command you would run and explain what a successful response looks like.

A. See the Explanation.

**Answer: A**

**Explanation:**

1. Run the ansible ad-hoc command:

```
ansible win_servers -i /home/ansible/inventories/windows_hosts.ini -m win_ping
```

2. Ansible will return:

```
winhost1 | SUCCESS => {"ping": "pong"}
```

```
winhost2 | SUCCESS => {"ping": "pong"}
```

3. This confirms successful WinRM communication from the control node to the Windows hosts.

3.You've received a requirement to switch from static to dynamic inventories using a script or plugin for Microsoft Azure. Describe the steps to set up Azure as a dynamic inventory source in Ansible.

A. See the Explanation.

**Answer: A**

**Explanation:**

1. Install the Azure inventory plugin: `pip install ansible[azure]`

2. Create a configuration file:

```
mkdir -p /home/ansible/inventories
```

```
nano /home/ansible/inventories/azure_rm.yml
```

3. Add:

```
plugin: azure_rm
```

```
include_vm_resource_groups:
```

- myResourceGroup auth\_source: auto

4. Test with:

```
ansible-inventory -i /home/ansible/inventories/azure_rm.yml --list
```

5. The Azure VMs will now be dynamically discovered and used as inventory.

4. You need to configure a Windows Server 2019 system to be managed by Ansible using WinRM.

Explain how to configure WinRM and the firewall on the Windows host.

A. See the Explanation.

**Answer: A**

**Explanation:**

1. Log into the Windows machine.

2. Open PowerShell as Administrator and run:

```
winrm quickconfig
```

```
winrm set winrm/config/service @{AllowUnencrypted="true"} winrm set winrm/config/service/auth  
@{Basic="true"}
```

```
Set-NetFirewallRule -Name "WINRM-HTTP-In-TCP-PUBLIC" -RemoteAddress Any
```

3. Enable the WinRM service:

```
Set-Service -Name WinRM -StartupType Automatic
```

```
Start-Service -Name WinRM
```

5. You need to verify that the WinRM port (5985) is open on a remote Windows host from your Ansible control node. Describe how to test this using telnet or nc.

A. See the Explanation.

**Answer: A**

**Explanation:**

1. Use telnet:

```
telnet 192.168.1.10 5985
```

or nc:

```
nc -zv 192.168.1.10 5985
```

2. If successful, it confirms WinRM port is open and accessible.