Windows Frequently Asked Questions

Here are some commonly asked questions in regards to Ansible and Windows and their answers.

> **Note**
>
> This document covers questions about managing Microsoft Windows servers with Ansible. For questions about Ansible Core, please see the :ref:`general FAQ page <ansible_faq>`.
>
> > **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 9**); *backlink*
> >
> > Unknown interpreted text role "ref".

Does Ansible work with Windows XP or Server 2003?

Ansible does not work with Windows XP or Server 2003 hosts. Ansible does work with these Windows operating system versions:

- Windows Server 2008 [1]
- Windows Server 2008 R2 [1]
- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019
- Windows 7 [1]
- Windows 8.1
- Windows 10

1 - See the :ref:`Server 2008 FAQ <windows_faq_server2008>` entry for more details.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 27**); *backlink*
>
> Unknown interpreted text role "ref".

Ansible also has minimum PowerShell version requirements - please see :ref:`windows_setup` for the latest information.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 29**); *backlink*
>
> Unknown interpreted text role "ref".

Are Server 2008, 2008 R2 and Windows 7 supported?

Microsoft ended Extended Support for these versions of Windows on January 14th, 2020, and Ansible deprecated official support in the 2.10 release. No new feature development will occur targeting these operating systems, and automated testing has ceased. However, existing modules and features will likely continue to work, and simple pull requests to resolve issues with these Windows versions may be accepted.

Can I manage Windows Nano Server with Ansible?

Ansible does not currently work with Windows Nano Server, since it does not have access to the full .NET Framework that is used by the majority of the modules and internal components.

Can Ansible run on Windows?

No, Ansible can only manage Windows hosts. Ansible cannot run on a Windows host natively, though it can run under the Windows Subsystem for Linux (WSL).

> **Note**
>
> The Windows Subsystem for Linux is not supported by Ansible and should not be used for production systems.

To install Ansible on WSL, the following commands can be run in the bash terminal:

```
sudo apt-get update
sudo apt-get install python-pip git libffi-dev libssl-dev -y
pip install --user ansible pywinrm
```

To run Ansible from source instead of a release on the WSL, simply uninstall the pip installed version and then clone the git repo.

```
pip uninstall ansible -y
git clone https://github.com/ansible/ansible.git
source ansible/hacking/env-setup

# To enable Ansible on login, run the following
echo ". ~/ansible/hacking/env-setup -q' >> ~/.bashrc
```

If you encounter timeout errors when running Ansible on the WSL, this may be due to an issue with `sleep` not returning correctly. The following workaround may resolve the issue:

```
mv /usr/bin/sleep /usr/bin/sleep.orig
ln -s /bin/true /usr/bin/sleep
```

Another option is to use WSL 2 if running Windows 10 later than build 2004.

```
wsl --set-default-version 2
```

Can I use SSH keys to authenticate to Windows hosts?

You cannot use SSH keys with the WinRM or PSRP connection plugins. These connection plugins use X509 certificates for authentication instead of the SSH key pairs that SSH uses.

The way X509 certificates are generated and mapped to a user is different from the SSH implementation; consult the :ref:`windows_winrm` documentation for more information.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, line 96);** *backlink*
>
> Unknown interpreted text role "ref".

Ansible 2.8 has added an experimental option to use the SSH connection plugin, which uses SSH keys for authentication, for Windows servers. See :ref:`this question <windows_faq_ssh>` for more information.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, line 100);** *backlink*
>
> Unknown interpreted text role "ref".

Why can I run a command locally that does not work under Ansible?

Ansible executes commands through WinRM. These processes are different from running a command locally in these ways:

- Unless using an authentication option like CredSSP or Kerberos with credential delegation, the WinRM process does not have the ability to delegate the user's credentials to a network resource, causing `Access is Denied` errors.
- All processes run under WinRM are in a non-interactive session. Applications that require an interactive session will not work.
- When running through WinRM, Windows restricts access to internal Windows APIs like the Windows Update API and DPAPI, which some installers and programs rely on.

Some ways to bypass these restrictions are to:

- Use `become`, which runs a command as it would when run locally. This will bypass most WinRM restrictions, as Windows is unaware the process is running under WinRM when `become` is used. See the :ref:`become` documentation for more information.

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, line 125);** *backlink*

> Unknown interpreted text role "ref".

- Use a scheduled task, which can be created with `win_scheduled_task`. Like `become`, it will bypass all WinRM restrictions, but it can only be used to run commands, not modules.

- Use `win_psexec` to run a command on the host. PSExec does not use WinRM and so will bypass any of the restrictions.

- To access network resources without any of these workarounds, you can use CredSSP or Kerberos with credential delegation enabled.

See :ref:`become` more info on how to use become. The limitations section at :ref:`windows_winrm` has more details around WinRM limitations.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-`
> `devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)`
> `(user_guide)windows_faq.rst`, **line 140);** *backlink*
>
> Unknown interpreted text role "ref".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-`
> `devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)`
> `(user_guide)windows_faq.rst`, **line 140);** *backlink*
>
> Unknown interpreted text role "ref".

This program won't install on Windows with Ansible

See :ref:`this question <windows_faq_winrm>` for more information about WinRM limitations.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-`
> `devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)`
> `(user_guide)windows_faq.rst`, **line 145);** *backlink*
>
> Unknown interpreted text role "ref".

What Windows modules are available?

Most of the Ansible modules in Ansible Core are written for a combination of Linux/Unix machines and arbitrary web services. These modules are written in Python and most of them do not work on Windows.

Because of this, there are dedicated Windows modules that are written in PowerShell and are meant to be run on Windows hosts. A list of these modules can be found :ref:`here <windows_modules>`.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-`
> `devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)`
> `(user_guide)windows_faq.rst`, **line 153);** *backlink*
>
> Unknown interpreted text role "ref".

In addition, the following Ansible Core modules/action-plugins work with Windows:

- add_host
- assert
- async_status
- debug
- fail
- fetch
- group_by
- include
- include_role
- include_vars
- meta
- pause
- raw
- script
- set_fact
- set_stats

- setup
- slurp
- template (also: win_template)
- wait_for_connection

Ansible Windows modules exist in the :ref:`plugins_in_ansible.windows`, :ref:`plugins_in_community.windows`, and :ref:`plugins_in_chocolatey.chocolatey` collections.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 180**); *backlink*
>
> Unknown interpreted text role "ref".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 180**); *backlink*
>
> Unknown interpreted text role "ref".

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 180**); *backlink*
>
> Unknown interpreted text role "ref".

Can I run Python modules on Windows hosts?

No, the WinRM connection protocol is set to use PowerShell modules, so Python modules will not work. A way to bypass this issue to use `delegate_to: localhost` to run a Python module on the Ansible controller. This is useful if during a playbook, an external service needs to be contacted and there is no equivalent Windows module available.

Can I connect to Windows hosts over SSH?

Ansible 2.8 has added an experimental option to use the SSH connection plugin to manage Windows hosts. To connect to Windows hosts over SSH, you must install and configure the Win32-OpenSSH fork that is in development with Microsoft on the Windows host(s). While most of the basics should work with SSH, `Win32-OpenSSH` is rapidly changing, with new features added and bugs fixed in every release. It is highly recommend you install the latest release of `Win32-OpenSSH` from the GitHub Releases page when using it with Ansible on Windows hosts.

To use SSH as the connection to a Windows host, set the following variables in the inventory:

```
ansible_connection=ssh

# Set either cmd or powershell not both
ansible_shell_type=cmd
# ansible_shell_type=powershell
```

The value for `ansible_shell_type` should either be `cmd` or `powershell`. Use `cmd` if the `DefaultShell` has not been configured on the SSH service and `powershell` if that has been set as the `DefaultShell`.

Why is connecting to a Windows host via SSH failing?

Unless you are using `Win32-OpenSSH` as described above, you must connect to Windows hosts using :ref:`windows_winrm`. If your Ansible output indicates that SSH was used, either you did not set the connection vars properly or the host is not inheriting them correctly.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 220**); *backlink*
>
> Unknown interpreted text role "ref".

Make sure `ansible_connection: winrm` is set in the inventory for the Windows host(s).

Why are my credentials being rejected?

This can be due to a myriad of reasons unrelated to incorrect credentials.

See HTTP 401/Credentials Rejected at :ref:`windows_setup` for a more detailed guide of this could mean.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 231);** *backlink*

Unknown interpreted text role "ref".

Why am I getting an error SSL CERTIFICATE_VERIFY_FAILED?

When the Ansible controller is running on Python 2.7.9+ or an older version of Python that has backported SSLContext (like Python 2.7.5 on RHEL 7), the controller will attempt to validate the certificate WinRM is using for an HTTPS connection. If the certificate cannot be validated (such as in the case of a self signed cert), it will fail the verification process.

To ignore certificate validation, add `ansible_winrm_server_cert_validation: ignore` to inventory for the Windows host.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\user_guide\(ansible-devel)(docs)(docsite)(rst)(user_guide)windows_faq.rst`, **line 246)**

Unknown directive type "seealso".

```
.. seealso::

   :ref:`windows`
       The Windows documentation index
   :ref:`about_playbooks`
       An introduction to playbooks
   :ref:`playbooks_best_practices`
       Tips and tricks for playbooks
   `User Mailing List <https://groups.google.com/group/ansible-project>`_
       Have a question?  Stop by the google group!
   :ref:`communication_irc`
       How to join Ansible chat channels
```