

# Ansible Stuff

Random ansible code, for various things, doing fun stuff. Nah, just a bunch of code I would like to keep track of.

- [Enable WinRM On Windows Server](#)
- [Configure Ansible To Use WinRM](#)

# Enable WinRM On Windows Server

You can enable WinRM on the Windows Servers by following these steps:

1. Open an elevated PowerShell console on the Windows Server.
2. Run the following command to enable WinRM:

```
winrm quickconfig
```

3. When prompted, type ☐ Y to allow WinRM to create a listener and set the firewall rules.
4. If you want to enable HTTPS for WinRM, run the following command:

```
winrm quickconfig -transport:https
```

5. When prompted, type ☐ Y to create a self-signed certificate.
6. Open the Windows Firewall with Advanced Security.
7. Create a new inbound rule for port 5985 (for HTTP) or 5986 (for HTTPS) to allow incoming connections.

Once you have completed these steps, WinRM will be enabled on the Windows Server and you can connect to it using Ansible's WinRM connection plugin.

# Configure Ansible To Use WinRM

To configure Ansible to connect to Windows Servers from a Red Hat Enterprise Linux 8 server, you need to install the following prerequisites:

1. Install the `pywinrm` module:

```
pip3 install pywinrm
```

2. Install the `xmlltodict` module:

```
pip3 install xmlltodict
```

3. Enable WinRM on the Windows Server and configure the firewall to allow incoming connections on the WinRM port (5985 for HTTP, 5986 for HTTPS).
4. Create a user account on the Windows Server with administrative privileges.

Once you have completed these prerequisites, you can configure Ansible to connect to the Windows Servers by adding the following to your `ansible.cfg` file:

```
[defaults]
inventory = /path/to/inventory.yml
remote_user = username
ansible_connection = winrm
ansible_winrm_server_cert_validation = ignore
```

Where `/path/to/inventory.yml` is the path to your inventory file containing the list of Windows Servers to manage, `username` is the name of the user account you created on the Windows Servers, and `ansible_winrm_server_cert_validation` is set to `ignore` to bypass certificate validation.

You can also specify the connection details for individual hosts in the inventory file using the following format:

```
[windows_servers]
windows_server_hostname ansible_host=ip_address ansible_user=username ansible_password=password
```

Where `windows_server_hostname` is the name of the Windows Server, `ip_address` is the IP address of the Windows Server, `username` is the name of the user account you created on the Windows Server, and `password` is the password for the user account.

Once you have configured Ansible to connect to the Windows Servers, you can use the `win_command`, `win_shell`, and other Windows-specific Ansible modules to manage the servers.