

Day 16

Control Node Setup:

- Install Ansible on the control node.

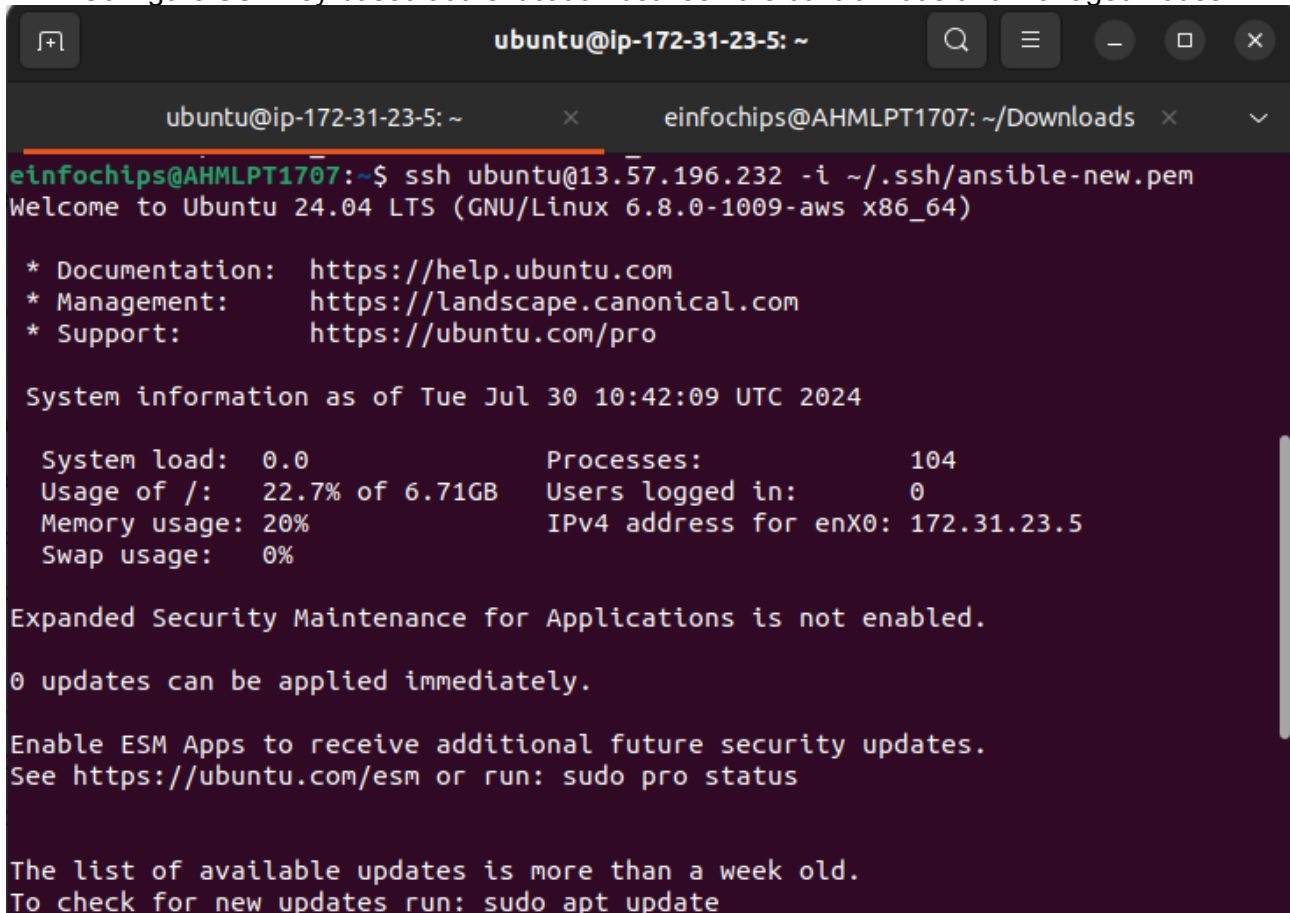
```
sudo apt update
```

```
sudo apt install software-properties-common
```

```
sudo add-apt-repository --yes --update ppa:ansible/ansible
```

```
sudo apt install ansible
```

Configure SSH key-based authentication between the control node and managed nodes.



The image shows a terminal window with two tabs. The active tab is titled 'ubuntu@ip-172-31-23-5: ~'. The other tab is titled 'einfochips@AHMLPT1707: ~/Downloads'. The terminal output shows an SSH session from 'einfochips@AHMLPT1707' to 'ubuntu@13.57.196.232' using a private key. The output includes a welcome message for Ubuntu 24.04 LTS, system information, and security notices.

```
ubuntu@ip-172-31-23-5: ~
einfochips@AHMLPT1707:~$ ssh ubuntu@13.57.196.232 -i ~/.ssh/ansible-new.pem
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1009-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

System information as of Tue Jul 30 10:42:09 UTC 2024

System load:  0.0               Processes:            104
Usage of /:   22.7% of 6.71GB   Users logged in:     0
Memory usage: 20%              IPv4 address for enX0: 172.31.23.5
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
```

```
ubuntu@ip-172-31-23-5: ~
Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-23-5:~$
```

Managed Nodes Configuration:

- Ensure all managed nodes are properly configured to be controlled by Ansible.

```
ubuntu@ip-172-31-23-5: ~
ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: ena>
  Drop-In: /usr/lib/systemd/system/ssh.service.d
           └─ec2-instance-connect.conf
  Active: active (running) since Tue 2024-07-30 10:25:20 UTC; 22min ago
  TriggeredBy: ● ssh.socket
  Docs: man:sshd(8)
        man:sshd_config(5)
  Process: 1012 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
  Main PID: 1014 (sshd)
    Tasks: 1 (limit: 1130)
  Memory: 4.0M (peak: 6.5M)
    CPU: 218ms
  CGroup: /system.slice/ssh.service
          └─1014 "sshd: /usr/sbin/sshd -D -o AuthorizedKeysCommand /usr/shar>

Jul 30 10:25:25 ip-172-31-23-5 sshd[1016]: Connection reset by 198.235.24.82 po>
Jul 30 10:40:11 ip-172-31-23-5 sshd[1027]: AuthorizedKeysCommand /usr/share/ec2>
Jul 30 10:40:11 ip-172-31-23-5 sshd[1027]: Connection closed by authenticating >
Jul 30 10:40:23 ip-172-31-23-5 sshd[1048]: AuthorizedKeysCommand /usr/share/ec2>
Jul 30 10:40:23 ip-172-31-23-5 sshd[1048]: Connection closed by authenticating >
Jul 30 10:41:46 ip-172-31-23-5 sshd[1064]: AuthorizedKeysCommand /usr/share/ec2>
Jul 30 10:41:46 ip-172-31-23-5 sshd[1064]: Connection closed by authenticating >
lines 1-23
```

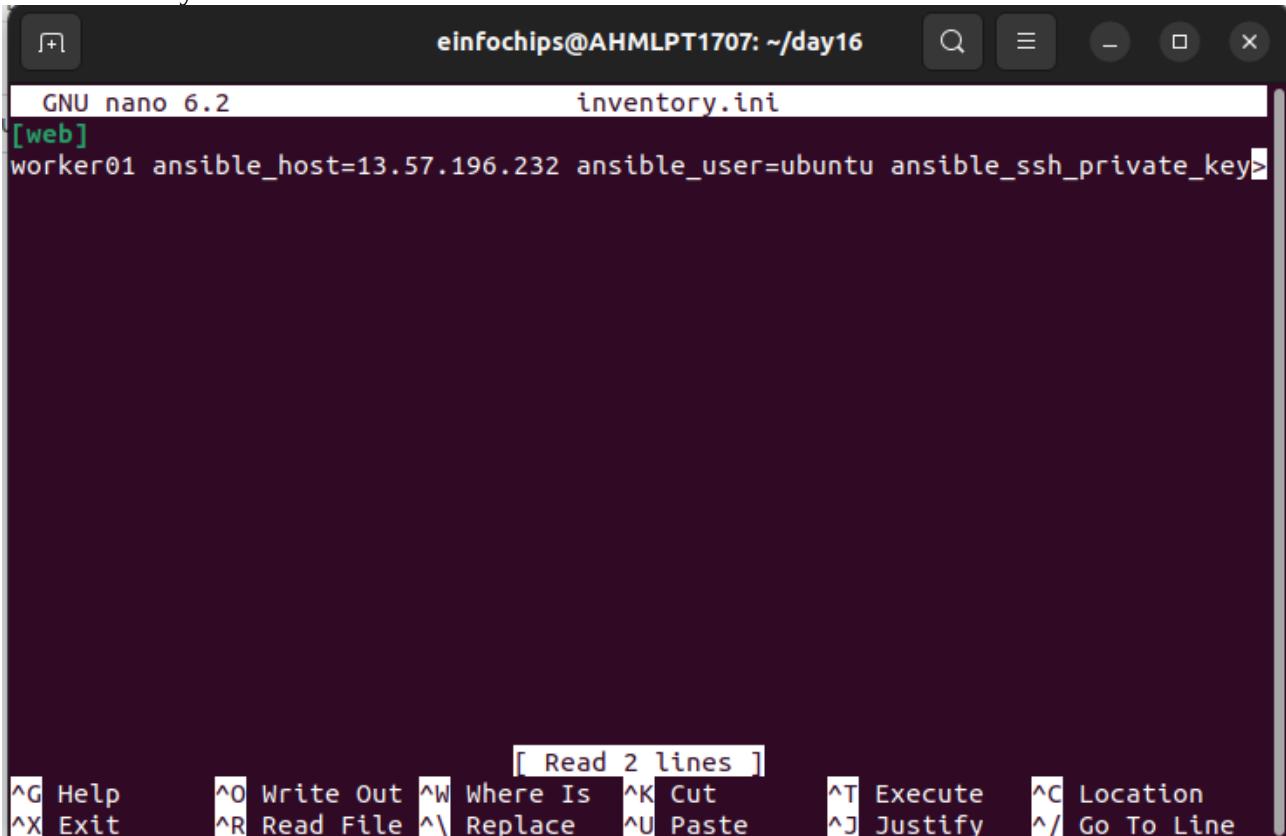
Configure Ansible on the Control Node

3.1. Set Up the Ansible Inventory File

The inventory file lists all managed nodes and their groups. By default, it is located at `/etc/ansible/hosts`. You can edit it or create a new inventory file in your project directory.

Creating a new inventory file

`nano inventory.ini`



```
einfochips@AHMLPT1707: ~/day16
GNU nano 6.2 inventory.ini
[web]
worker01 ansible_host=13.57.196.232 ansible_user=ubuntu ansible_ssh_private_key=
[ Read 2 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut      ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace  ^U Paste    ^J Justify  ^_ Go To Line
```

Verify connectivity and proper setup between the control node and managed nodes.

`ansible -i ~/day16/inventory.ini web -m ping`

This command should return `pong` from each managed node if everything is set up correctly.

```
einfochips@AHMLPT1707: ~/day16
'/usr/share/ansible/plugins/modules']
ansible python module location = /home/einfochips/.local/lib/python3.10/site-packages/ansible
ansible collection location = /home/einfochips/.ansible/collections:/usr/share/ansible/collections
executable location = /home/einfochips/.local/bin/ansible
python version = 3.10.12 (main, Mar 22 2024, 16:50:05) [GCC 11.4.0] (/usr/bin/python3)
jinja version = 3.1.4
libyaml = True
einfochips@AHMLPT1707:~/day16$ ansible -i ~/day16/inventory.ini web -m ping
[WARNING]: Platform linux on host worker01 is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
worker01 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.12"
  },
  "changed": false,
  "ping": "pong"
}
einfochips@AHMLPT1707:~/day16$
```

Ad-Hoc Ansible Commands

Problem Statement: Your organization needs to perform frequent, one-off administrative tasks across a fleet of servers. These tasks include checking disk usage, restarting services, and updating packages. You are required to use Ansible ad-hoc commands to accomplish these tasks efficiently.

Execute commands to check disk usage across all managed nodes.

```
ansible all -i /path/to/your/inventory.ini -m command -a "df -h"
```

```
einfochips@AHMLPT1707:~/day16$ ansible all -i inventory.ini -m command -a "df -h"
[WARNING]: Platform linux on host worker01 is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
worker01 | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        6.8G  1.6G  5.2G  23% /
tmpfs            479M   0  479M   0% /dev/shm
tmpfs            192M  868K  191M   1% /run
tmpfs            5.0M   0   5.0M   0% /run/lock
/dev/xvda16      881M   76M  744M  10% /boot
/dev/xvda15      105M   6.1M   99M   6% /boot/efi
tmpfs            96M   12K   96M   1% /run/user/1000
einfochips@AHMLPT1707:~/day16$
```

Restart a specific service on all managed nodes.

ansible all -i inventory.ini -m apt -a "name=nginx state=present" --become

```
elInFochips@AHMLPT1707: ~/day1$ ansible all -i inventory.ini -m apt -a "name=nginx state=present" --become
[WARNING]: Platform linux on host worker01 is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of another Python interpreter could change the meaning of that path.
See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
worker01 | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.12"
  },
  "cache_update_time": 1719049974,
  "cache_updated": false,
  "changed": true,
  "stderr": "",
  "stderr_lines": [],
  "stdout": "Reading package lists...
Building dependency tree...
Reading state information...
The following additional packages will be installed:
  nginx-common
Suggested packages:
  fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  nginx-common
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 552 kB of archives.
After this operation
 1596 kB of additional disk space will be used.
Get:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 nginx-common all 1.24.0-2ubuntu7 [31.2 kB]
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 nginx amd64 1.24.0-2ubuntu7 [521 kB]
Preconfiguring packages ...
Fetched 552 kB in 0s (12.7 MB/s)
Selecting previously unselected package nginx-common.
(Reading database ... 58%(Reading database ... 100%(Reading database ... 158%(Reading database ... 208%(Reading database ... 253%(Reading database ... 308%(Reading database ... 353%(Reading database ... 408%(Reading database ... 453%(Reading database ... 503%(Reading database ... 553%(Reading database ... 603%(Reading database ... 653%(Reading database ... 703%(Reading database ... 753%(Reading database ... 803%(Reading database ... 853%(Reading database ... 903%(Reading database ... 953%(Reading database ... 1003%(Reading database ... 67739 files and directories currently installed.)
Preparing to unpack .../nginx-common-1.24.0-2ubuntu7_all.deb ...
Unpacking nginx-common (1.24.0-2ubuntu7) ...
Selecting previously unselected package nginx.
Preparing to unpack .../nginx-1.24.0-2ubuntu7_amd64.deb ...
Unpacking nginx (1.24.0-2ubuntu7) ...
Setting up nginx-common (1.24.0-2ubuntu7) ...
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.
Processing triggers for ufw (0.36-2.6) ...
Processing triggers for man-db (2.12.0-4build2) ...
Running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
"
  "stdout_lines": [
    "Reading package lists...",
    "Building dependency tree...",
    "Reading state information...",
    "The following additional packages will be installed:",
    "  nginx-common",
    "Suggested packages:",
    "  fcgiwrap nginx-doc ssl-cert",
    "The following NEW packages will be installed:",
    "  nginx-common",
    "0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.",
    "Need to get 552 kB of archives.",

```

```
elInFochips@AHMLPT1707: ~/day1$ ansible all -i inventory.ini -m service -a "name=nginx state=restarted" --become
[WARNING]: Platform linux on host worker01 is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of another Python interpreter could change the meaning of that path.
See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
worker01 | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.12"
  },
  "changed": true,
  "name": "nginx",
  "state": "started",
  "status": {
    "ActiveEnterTimestamp": "Tue 2024-07-30 11:57:17 UTC",
    "ActiveEnterTimestampMonotonic": "6508729939",
    "ActiveExitTimestampMonotonic": "0",
    "ActiveState": "active",
    "After": "nss-lookup.target systemd-journald.socket basic.target sysinit.target remote-fs.target system.slice network-online.target",
    "AllowIsolate": "no",
    "AssertResult": "yes",
    "AssertTimestamp": "Tue 2024-07-30 11:57:17 UTC",
    "AssertTimestampMonotonic": "6508703837",
    "Before": "shutdown.target multi-user.target",
    "BlockIOAccounting": "no",
    "BlockIOWeight": "[not set]",
    "CPUAccounting": "yes",
    "CPUAffinityFromNUMA": "no",
    "CPUQuotaPerSecUSec": "infinity",
    "CPUQuotaPerPeriodUSec": "infinity",
    "CPUSchedulingPolicy": "0",
    "CPUSchedulingPriority": "0",
    "CPUSchedulingResetOnFork": "no",
    "CPUShares": "[not set]",
    "CPUUsageSec": "12015000",
    "CPUWeight": "[not set]",
    "CacheDirectoryMode": "0755",
    "CanFreeze": "yes",
    "CanRestart": "yes",

```

```

    "StateDirectoryMode": "0755",
    "StatusError": "0",
    "StopWhenUnneeded": "no",
    "SubState": "running",
    "SuccessAction": "none",
    "SurviveFinalKillSignal": "no",
    "SyslogFacility": "3",
    "SyslogLevel": "6",
    "SyslogLevelPrefix": "yes",
    "SyslogPriority": "30",
    "SystemCallErrorNumber": "2147483646",
    "TTYReset": "no",
    "TTYVHangup": "no",
    "TTYVDDeallocate": "no",
    "TasksAccounting": "yes",
    "TasksCurrent": "12",
    "TasksMax": "1130",
    "TimeoutAbortUSec": "5s",
    "TimeoutCleanUSec": "infinity",
    "TimeoutStartFailureMode": "terminate",
    "TimeoutStartUSec": "1min 30s",
    "TimeoutStopFailureMode": "terminate",
    "TimeoutStopUSec": "5s",
    "TierslackNSec": "50000",
    "Transient": "no",
    "Type": "forking",
    "UID": "[not set]",
    "UMask": "0022",
    "UnitFilePresets": "enabled",
    "UnitFileState": "enabled",
    "UtmpMode": "init",
    "WantedBy": "multi-user.target",
    "Wants": "network-online.target",
    "WatchdogSignal": "0",
    "WatchdogTimestampMonotonic": "0",
    "WatchdogUSec": "0",

```

Update all packages on a subset of managed nodes.


```
einfochips@AHMLPT1707: ~/day16
[WARNING]: no hosts matched, nothing to do
einfochips@AHMLPT1707:~/day16$ ansible all -i inventory.ini -m package -a "name=* state=latest" --become
[WARNING]: Platform linux on host worker01 is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
worker01 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.12"
  },
  "changed": false,
  "msg": "Reading package lists...\nBuilding dependency tree...\nReading state information...\nCalculating upgrade...\n0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.\n",
  "stderr": "",
  "stderr_lines": [],
  "stdout": "Reading package lists...\nBuilding dependency tree...\nReading state information...\nCalculating upgrade...\n0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.\n",
  "stdout_lines": [
    "Reading package lists...",
    "Building dependency tree...",
    "Reading state information..."
  ]
}
```

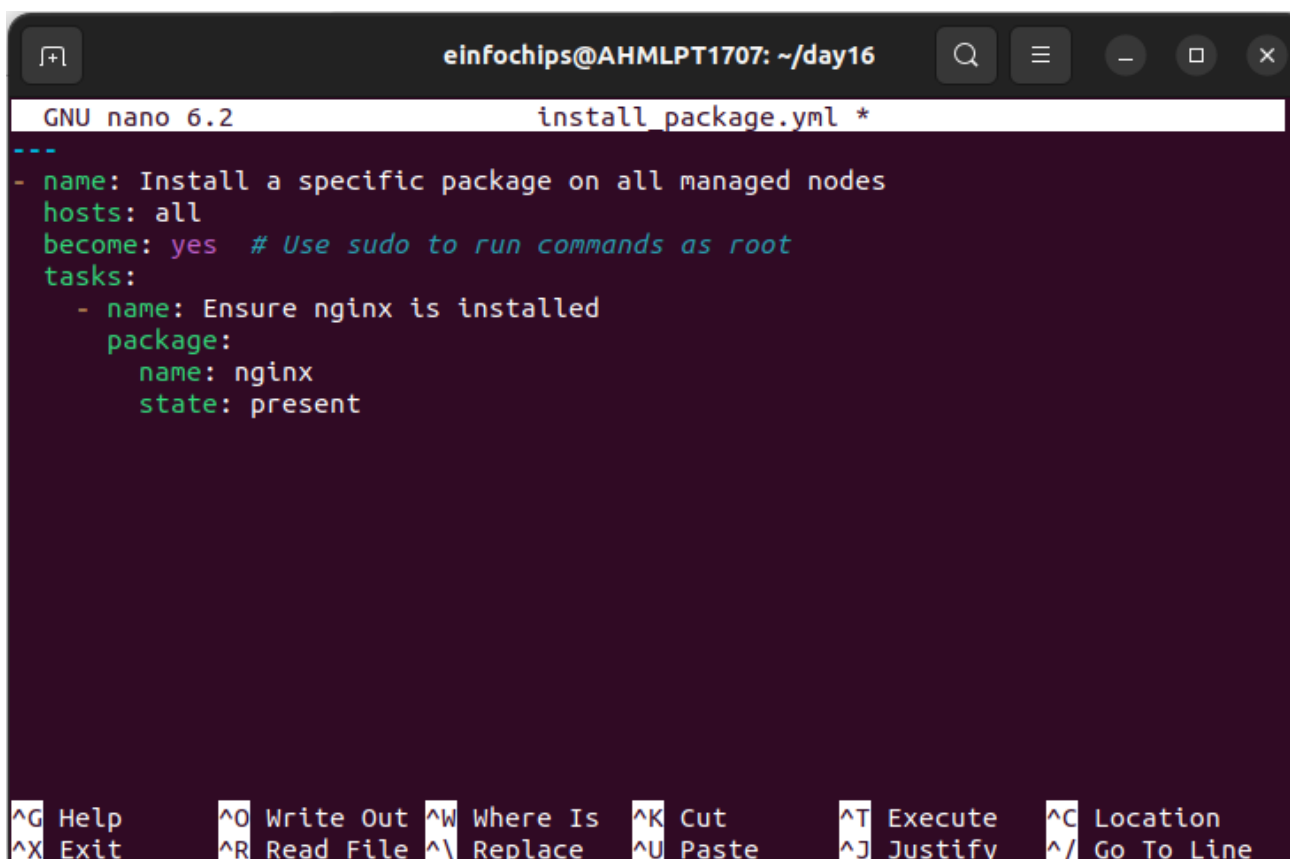
```
einfochips@AHMLPT1707: ~/day16
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
worker01 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.12"
  },
  "changed": false,
  "msg": "Reading package lists...\nBuilding dependency tree...\nReading state information...\nCalculating upgrade...\n0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.\n",
  "stderr": "",
  "stderr_lines": [],
  "stdout": "Reading package lists...\nBuilding dependency tree...\nReading state information...\nCalculating upgrade...\n0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.\n",
  "stdout_lines": [
    "Reading package lists...",
    "Building dependency tree...",
    "Reading state information...",
    "Calculating upgrade...",
    "0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded."
  ]
}
einfochips@AHMLPT1707:~/day16$
```

Playbook Creation:

- Write a playbook to install a specific package on all managed nodes.
- Create a playbook to configure a service with specific parameters.
- Develop a playbook to manage files, such as creating, deleting, and modifying files on managed nodes.

2. Testing and Verification:

- Test the playbooks to ensure they run successfully and perform the intended tasks.
- Validate the changes made by the playbooks on the managed nodes.



The screenshot shows a terminal window with a dark background. At the top, the window title is 'einfochips@AHMLPT1707: ~/day16'. Below the title bar, the nano editor's status line shows 'GNU nano 6.2' and the filename 'install_package.yml *'. The main content area displays a YAML playbook snippet:

```
---
- name: Install a specific package on all managed nodes
  hosts: all
  become: yes # Use sudo to run commands as root
  tasks:
    - name: Ensure nginx is installed
      package:
        name: nginx
        state: present
```

At the bottom of the terminal, a row of keyboard shortcuts is displayed:

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location
^X Exit	^R Read File	^\\ Replace	^U Paste	^J Justify	^/ Go To Line

```
einfochips@AHMLPT1707: ~/day16
GNU nano 6.2      configure_service.yml *
```

```
---
- name: Configure nginx service with specific parameters
  hosts: all
  become: yes
  tasks:
    - name: Ensure nginx is installed (if not already installed)
      package:
        name: nginx
        state: present

    - name: Deploy custom nginx configuration
      copy:
        src: files/nginx.conf
        dest: /etc/nginx/nginx.conf
        owner: root
        group: root
        mode: '0644'

    - name: Restart nginx service to apply configuration
      service:
```

```
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

```
einfochips@AHMLPT1707: ~/day16
GNU nano 6.2      manage_files.yml *
```

```
---
- name: Manage files on managed nodes
  hosts: all
  become: yes
  tasks:
    - name: Create a directory
      file:
        path: /opt/mydir
        state: directory
        owner: root
        group: root
        mode: '0755'

    - name: Create a file with content
      copy:
        dest: /opt/mydir/myfile.txt
        content: |
          This is a sample file.
          Created by Ansible playbook.
        owner: root
```

```
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```



```
einfochips@AHMLPT1707: ~/day16
einfochips@AHMLPT1707:~/day16$ nano install_package.yml
einfochips@AHMLPT1707:~/day16$ nano configure_service.yml
einfochips@AHMLPT1707:~/day16$ nano manage_files.yml
einfochips@AHMLPT1707:~/day16$ ansible-playbook -i inventory.ini install_package.yml

PLAY [Install a specific package on all managed nodes] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host worker01 is using the discovered Python
interpreter at /usr/bin/python3.12, but future installation of another Python
interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [worker01]

TASK [Ensure nginx is installed] *****
ok: [worker01]

PLAY RECAP *****
worker01 : ok=2 changed=0 unreachable=0 failed=0 s
kipped=0 rescued=0 ignored=0

einfochips@AHMLPT1707:~/day16$
```

```
einfochips@AHMLPT1707: ~/day16
PLAY RECAP *****
worker01 : ok=2 changed=0 unreachable=0 failed=0 s
kipped=0 rescued=0 ignored=0

einfochips@AHMLPT1707:~/day16$ ansible-playbook -i inventory.ini configure_servi
ce.yml

PLAY [Configure nginx service with specific parameters] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host worker01 is using the discovered Python
interpreter at /usr/bin/python3.12, but future installation of another Python
interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [worker01]

TASK [Ensure nginx is installed (if not already installed)] *****
ok: [worker01]

TASK [Deploy custom nginx configuration] *****
changed: [worker01]

TASK [Restart nginx service to apply configuration] *****
```

```
einfochips@AHMLPT1707: ~/day16
einfochips@AHMLPT1707:~/day16$ ansible-playbook -i inventory.ini manage_files.yml

PLAY [Manage files on managed nodes] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host worker01 is using the discovered Python
interpreter at /usr/bin/python3.12, but future installation of another Python
interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [worker01]

TASK [Create a directory] *****
changed: [worker01]

TASK [Create a file with content] *****
changed: [worker01]

TASK [Ensure a file is absent] *****
ok: [worker01]

TASK [Modify a file's content (append text)] *****
changed: [worker01]

PLAY RECAP *****
worker01 : ok=5    changed=3    unreachable=0    failed=0    skipped=0    rescued=0
         ignored=0
```

Playbook with Error Handling:

- Write a playbook that includes tasks likely to fail, such as starting a non-existent service or accessing a non-existent file.

Implement error handling strategies using modules like **block**, **rescue**, and **always**.

```
einfochips@AHMLPT1707: ~/day16
GNU nano 6.2 error_handling_playbook.yml *
--
- name: Playbook with Error Handling
  hosts: all
  become: yes
  tasks:
    - name: Ensure a specific service is started
      block:
        - name: Start a non-existent service
          service:
            name: non_existent_service
            state: started
      rescue:
        - name: Log error for non-existent service
          debug:
            msg: "The service 'non_existent_service' could not be started because it does not exist."

    - name: Manage files with error handling
      block:
        - name: Access a non-existent file
          command: cat /path/to/non_existent_file.txt
      rescue:
        - name: Log error for non-existent file
          debug:
            msg: "The file '/path/to/non_existent_file.txt' could not be accessed because it does not exist."

  always:

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
```

```
einfochips@AHMLPT1707: ~/day16
einfochips@AHMLPT1707:~/day16$ ansible-playbook -i inventory.ini error_handling_playbook.yml

PLAY [Playbook with Error Handling] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host worker01 is using the discovered Python interpreter at
/usr/bin/python3.12, but future installation of another Python interpreter could change the meaning
of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [worker01]

TASK [Start a non-existent service] *****
fatal: [worker01]: FAILED! => {"changed": false, "msg": "Could not find the requested service non_exi
stent_service: host"}

TASK [Log error for non-existent service] *****
ok: [worker01] => {
  "msg": "The service 'non_existent_service' could not be started because it does not exist."
}

TASK [Access a non-existent file] *****
fatal: [worker01]: FAILED! => {"changed": true, "cmd": ["cat", "/path/to/non_existent_file.txt"], "de
lta": "0:00:00.003409", "end": "2024-07-30 12:22:38.627751", "msg": "non-zero return code", "rc": 1,
"start": "2024-07-30 12:22:38.624342", "stderr": "cat: /path/to/non_existent_file.txt: No such file o
r directory", "stderr_lines": ["cat: /path/to/non_existent_file.txt: No such file or directory"], "st
dout": "", "stdout_lines": []}

TASK [Log error for non-existent file] *****
ok: [worker01] => {
  "msg": "The file '/path/to/non existent file.txt' could not be accessed because it does not exist"
}
```

```
einfochips@AHMLPT1707: ~/day16

TASK [Log error for non-existent service] *****
ok: [worker01] => {
  "msg": "The service 'non_existent_service' could not be started because it does not exist."
}

TASK [Access a non-existent file] *****
fatal: [worker01]: FAILED! => {"changed": true, "cmd": ["cat", "/path/to/non_existent_file.txt"], "delta": "0:00:00.003409", "end": "2024-07-30 12:22:38.627751", "msg": "non-zero return code", "rc": 1, "start": "2024-07-30 12:22:38.624342", "stderr": "cat: /path/to/non_existent_file.txt: No such file or directory", "stderr_lines": ["cat: /path/to/non_existent_file.txt: No such file or directory"], "stdout": "", "stdout_lines": []}

TASK [Log error for non-existent file] *****
ok: [worker01] => {
  "msg": "The file '/path/to/non_existent_file.txt' could not be accessed because it does not exist."
}

TASK [Clean up or perform a final action] *****
ok: [worker01] => {
  "msg": "Error handling block has completed execution, regardless of success or failure of previous tasks."
}

PLAY RECAP *****
worker01 : ok=4 changed=0 unreachable=0 failed=0 skipped=0 rescued=2 ignored=0

einfochips@AHMLPT1707:~/day16$
```