Writing Multiple Plays

Introduction:

A playbook is a YAML file containing a list of one or more plays. Remember that a single play is an ordered list of tasks to execute against hosts selected from the inventory. Therefore, if a playbook contains multiple plays, each play may apply its tasks to a separate set of hosts. This can be very useful when orchestrating a complex deployment which may involve different tasks on different hosts. You can write a playbook that runs one play against one set of hosts, and when that finish runs another play against another set of hosts.

Objectives:

- Writing multiple plays in a single playbook

In this Lab you will create a playbook containing **Multiple Plays** then use it to perform configuration tasks on managed hosts.

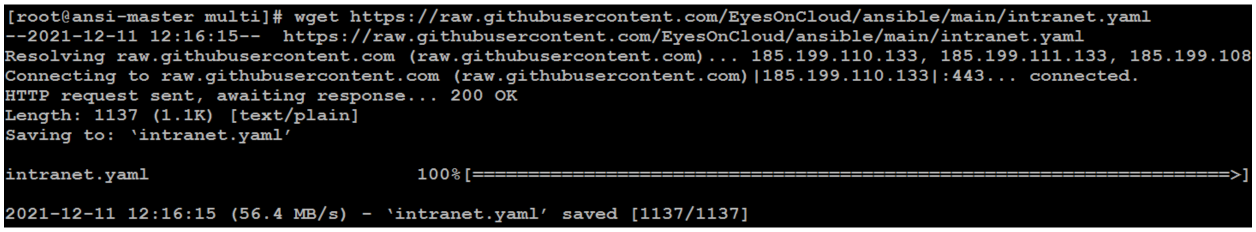
1.1 Create a new playbook **/root/playbook/multi/intranet.yaml**, and add lines needed to start the first play. it should target the managed host **ansi-node1.example.com** and enable privilege escalation.

```
#  mkdir multi
#  cd multi
```

Let us download the yaml from GitHub

```
#  wget
https://raw.githubusercontent.com/EyesOnCloud/ansible/main/i
ntranet.yaml
```

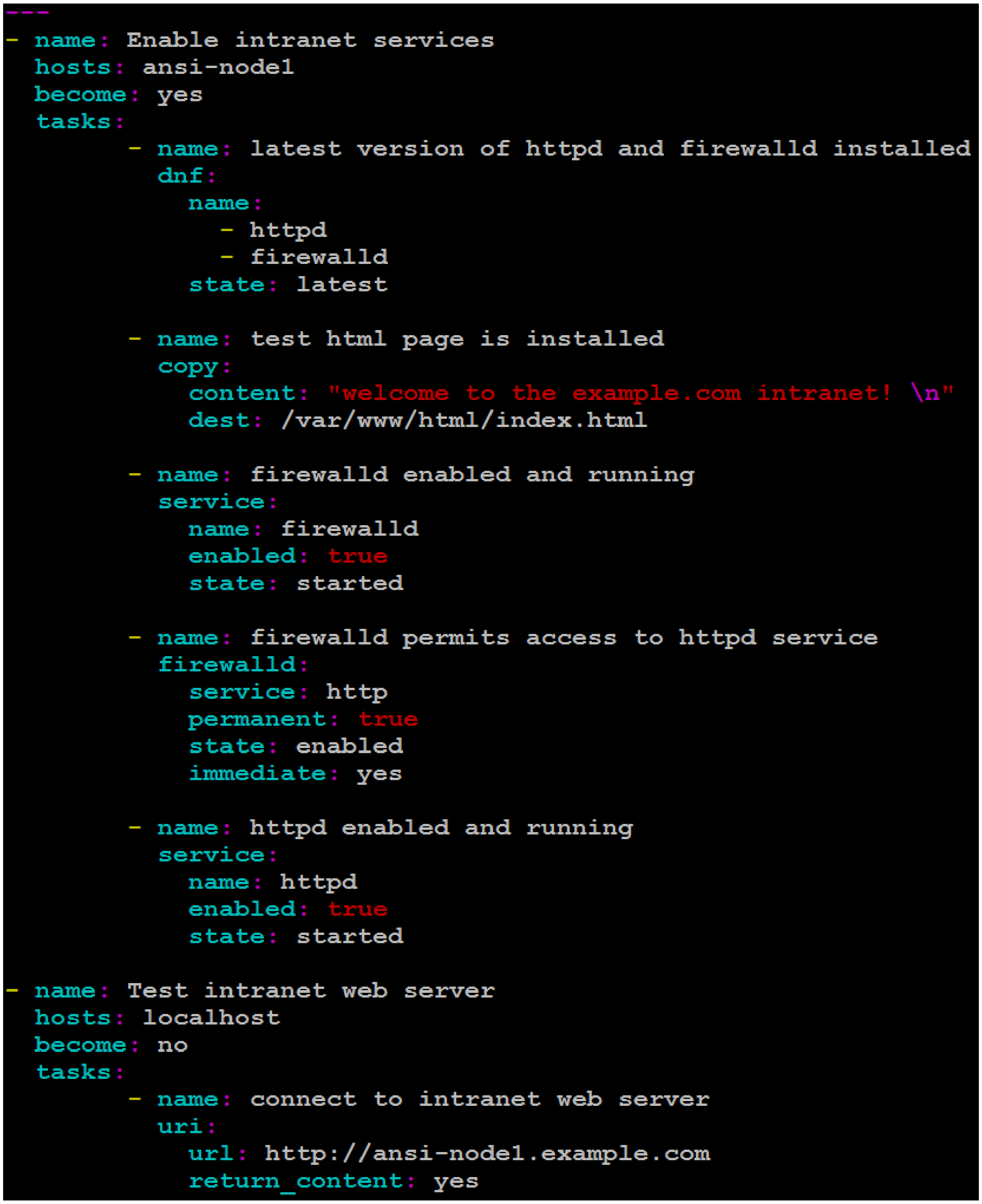
Output:



```
[root@ansi-master multi]# wget https://raw.githubusercontent.com/EyesOnCloud/ansible/main/intranet.yaml
--2021-12-11 12:16:15--  https://raw.githubusercontent.com/EyesOnCloud/ansible/main/intranet.yaml
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.110.133, 185.199.111.133, 185.199.108
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.110.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1137 (1.1K) [text/plain]
Saving to: ‘intranet.yaml’

intranet.yaml                  100%[===================================================================>]

2021-12-11 12:16:15 (56.4 MB/s) - ‘intranet.yaml’ saved [1137/1137]
```

```
# cat -n intranet.yaml
```

Output:

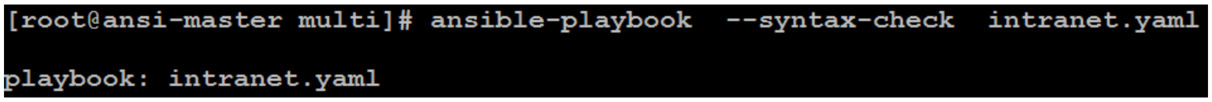


```
---
- name: Enable intranet services
  hosts: ansi-node1
  become: yes
  tasks:
        - name: latest version of httpd and firewalld installed
          dnf:
            name:
              - httpd
              - firewalld
            state: latest

        - name: test html page is installed
          copy:
            content: "welcome to the example.com intranet! \n"
            dest: /var/www/html/index.html

        - name: firewalld enabled and running
          service:
            name: firewalld
            enabled: true
            state: started

        - name: firewalld permits access to httpd service
          firewalld:
            service: http
            permanent: true
            state: enabled
            immediate: yes

        - name: httpd enabled and running
          service:
            name: httpd
            enabled: true
            state: started

- name: Test intranet web server
  hosts: localhost
  become: no
  tasks:
        - name: connect to intranet web server
          uri:
            url: http://ansi-node1.example.com
            return_content: yes
```

1.2 Run the ansible-playbook –syntax-check command to verify the syntax of the /home/devops/multi/intranet.yml playbook:

```
# ansible-playbook  --syntax-check intranet.yaml
```

Output:

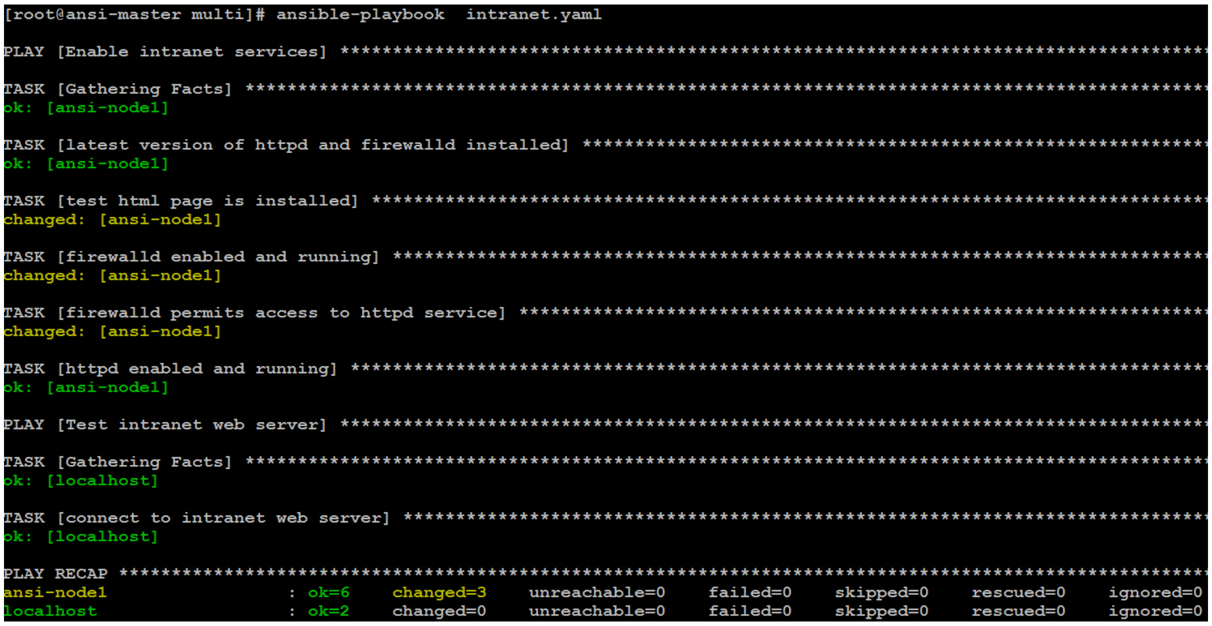


```
[root@ansi-master multi]# ansible-playbook  --syntax-check  intranet.yaml

playbook: intranet.yaml
```

Let’s run the play book

```
# ansible-playbook  intranet.yaml
```

Output:

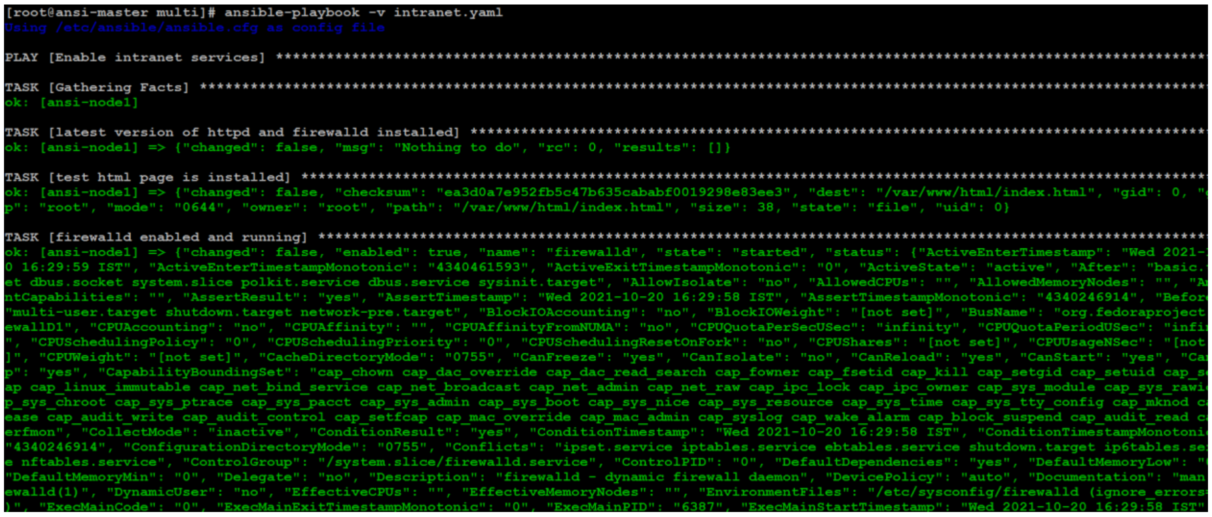


```
[root@ansi-master multi]# ansible-playbook  intranet.yaml

PLAY [Enable intranet services] ************************************************************************

TASK [Gathering Facts] *********************************************************************************
ok: [ansi-node1]

TASK [latest version of httpd and firewalld installed] *************************************************
ok: [ansi-node1]

TASK [test html page is installed] *********************************************************************
changed: [ansi-node1]

TASK [firewalld enabled and running] *******************************************************************
changed: [ansi-node1]

TASK [firewalld permits access to httpd service] *******************************************************
changed: [ansi-node1]

TASK [httpd enabled and running] ***********************************************************************
ok: [ansi-node1]

PLAY [Test intranet web server] ************************************************************************

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

TASK [connect to intranet web server] ******************************************************************
ok: [localhost]

PLAY RECAP *********************************************************************************************
ansi-node1                 : ok=6    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
localhost                  : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

1.6 Lets execute the playbook using the -v option to output detailed results for each task.

```
# ansible-playbook -v intranet.yaml
```

Output:

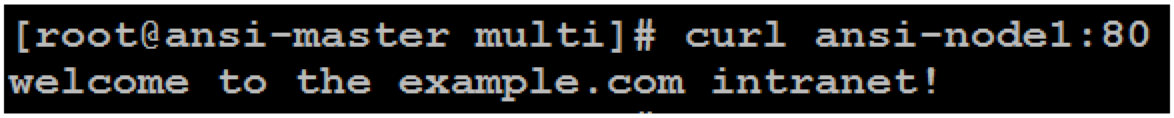


```
[root@ansi-master multi]# ansible-playbook -v intranet.yaml
Using /etc/ansible/ansible.cfg as config file

PLAY [Enable intranet services] ******************************************************************************

TASK [Gathering Facts] ***************************************************************************************
ok: [ansi-node1]

TASK [latest version of httpd and firewalld installed] *******************************************************
ok: [ansi-node1] => {"changed": false, "msg": "Nothing to do", "rc": 0, "results": []}

TASK [test html page is installed] ***************************************************************************
ok: [ansi-node1] => {"changed": false, "checksum": "ea3d0a7e952fb5c47b635cababf0019298e83ee3", "dest": "/var/www/html/index.html", "gid": 0, "
p": "root", "mode": "0644", "owner": "root", "path": "/var/www/html/index.html", "size": 38, "state": "file", "uid": 0}

TASK [firewalld enabled and running] *************************************************************************
ok: [ansi-node1] => {"changed": false, "enabled": true, "name": "firewalld", "state": "started", "status": {"ActiveEnterTimestamp": "Wed 2021-
0 16:29:59 IST", "ActiveEnterTimestampMonotonic": "4340461593", "ActiveExitTimestampMonotonic": "0", "ActiveState": "active", "After": "basic.
et dbus.socket system.slice polkit.service dbus.service sysinit.target", "AllowIsolate": "no", "AllowedCPUs": "", "AllowedMemoryNodes": "", "A
ntCapabilities": "", "AssertResult": "yes", "AssertTimestamp": "Wed 2021-10-20 16:29:58 IST", "AssertTimestampMonotonic": "4340246914", "Befor
"multi-user.target shutdown.target network-pre.target", "BlockIOAccounting": "no", "BlockIOWeight": "[not set]", "BusName": "org.fedoraproject
ewallD1", "CPUAccounting": "no", "CPUAffinity": "", "CPUAffinityFromNUMA": "no", "CPUQuotaPerSecUSec": "infinity", "CPUQuotaPeriodUSec": "infi
", "CPUSchedulingPolicy": "0", "CPUSchedulingPriority": "0", "CPUSchedulingResetOnFork": "no", "CPUShares": "[not set]", "CPUUsageNSec": "[not
]", "CPUWeight": "[not set]", "CacheDirectoryMode": "0755", "CanFreeze": "yes", "CanIsolate": "no", "CanReload": "yes", "CanStart": "yes", "Ca
p": "yes", "CapabilityBoundingSet": "cap_chown cap_dac_override cap_dac_read_search cap_fowner cap_fsetid cap_kill cap_setgid cap_setuid cap_s
ap cap_linux_immutable cap_net_bind_service cap_net_broadcast cap_net_admin cap_net_raw cap_ipc_lock cap_ipc_owner cap_sys_module cap_sys_rawi
p_sys_chroot cap_sys_ptrace cap_sys_pacct cap_sys_admin cap_sys_boot cap_sys_nice cap_sys_resource cap_sys_time cap_sys_tty_config cap_mknod c
ease cap_audit_write cap_audit_control cap_setfcap cap_mac_override cap_mac_admin cap_syslog cap_wake_alarm cap_block_suspend cap_audit_read c
erfmon", "CollectMode": "inactive", "ConditionResult": "yes", "ConditionTimestamp": "Wed 2021-10-20 16:29:58 IST", "ConditionTimestampMonotoni
"4340246914", "ConfigurationDirectoryMode": "0755", "Conflicts": "ipset.service iptables.service ebtables.service shutdown.target ip6tables.se
e nftables.service", "ControlGroup": "/system.slice/firewalld.service", "ControlPID": "0", "DefaultDependencies": "yes", "DefaultMemoryLow": "
"DefaultMemoryMin": "0", "Delegate": "no", "Description": "firewalld - dynamic firewall daemon", "DevicePolicy": "auto", "Documentation": "man
ewalld(1)", "DynamicUser": "no", "EffectiveCPUs": "", "EffectiveMemoryNodes": "", "EnvironmentFiles": "/etc/sysconfig/firewalld (ignore_errors
)", "ExecMainCode": "0", "ExecMainExitTimestampMonotonic": "0", "ExecMainPID": "6387", "ExecMainStartTimestamp": "Wed 2021-10-20 16:29:58 IST"
```

1.7 Use the curl command to verify that ansi-node1 is configured as an HTTPD server.

```
# curl ansi-node1:80
```

Output:



```
[root@ansi-master multi]# curl ansi-node1:80
welcome to the example.com intranet!
```