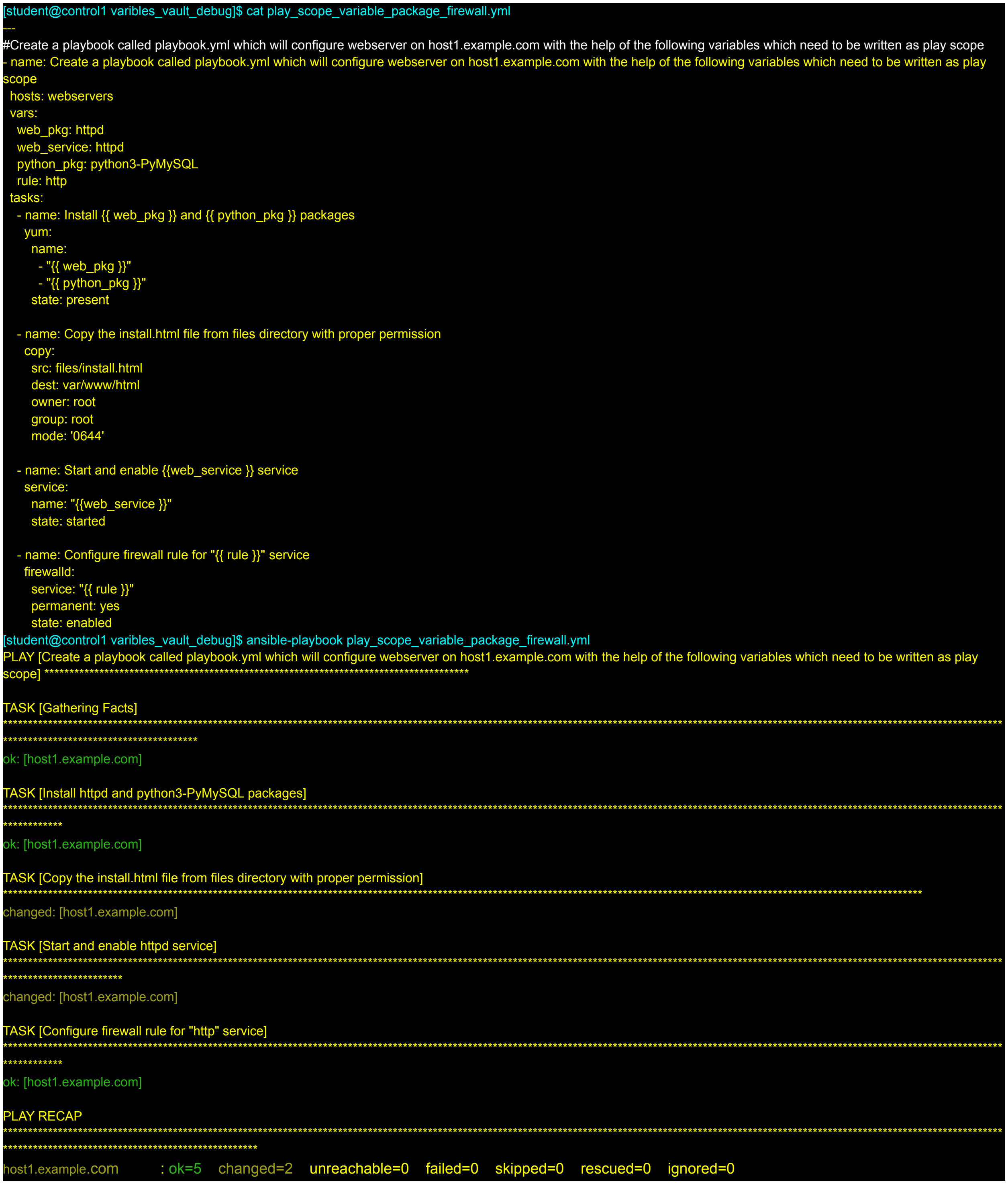
```
## Defining variables for inventories
[student@control1 varibles_vault_debug]$ cat createuser_using_inventory_variables.yml
---
#Creating user using inventory variables
- name: Create user using inventory variables
  hosts: webservers
  tasks:
    - name: Create user {{ user }}
      user:
        name: "{{ user }}"

[student@control1 varibles_vault_debug]$ cat inventory
[webservers]
host1.example.com user=shivram

[dbservers]
host2.example.com

[datacenters:children]
webservers
dbservers
[student@control1 varibles_vault_debug]$

[student@control1 varibles_vault_debug]$ ansible-playbook createuser_using_inventory_variables.yml
--syntax-check

playbook: createuser_using_inventory_variables.yml
[student@control1 varibles_vault_debug]$ ansible-playbook createuser_using_inventory_variables.yml

PLAY [Create user using inventory variables]
**************************************************************************************************************************************
**************************

TASK [Gathering Facts]
**************************************************************************************************************************************
************************************************
ok: [host1.example.com]

TASK [Create user shivram]
**************************************************************************************************************************************
********************************************
changed: [host1.example.com]

PLAY RECAP
**************************************************************************************************************************************
************************************************************
host1.example.com          : ok=2    changed=1    unreac
```

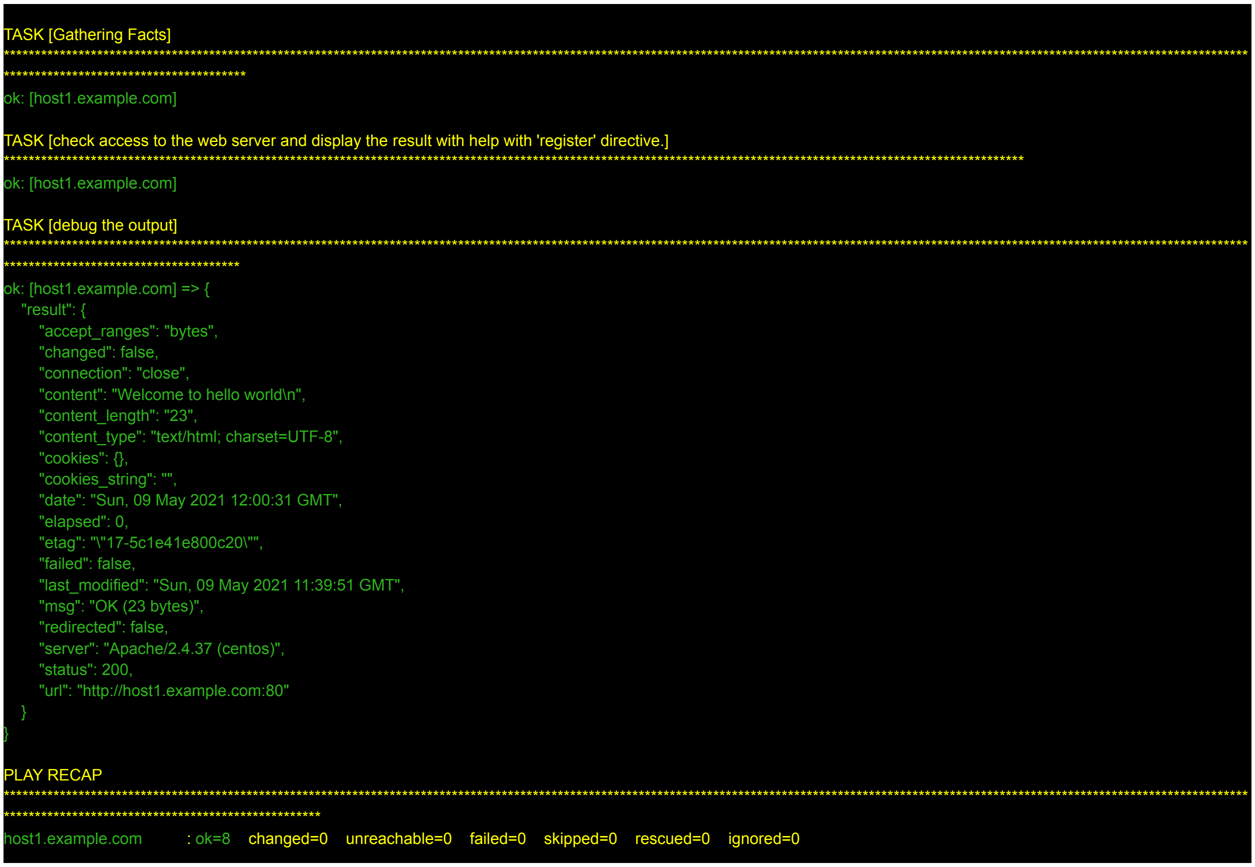
```
### Defining Variables in Playbooks
[student@control1 varibles_vault_debug]$ cat createuser_using_variables_in_playbook.yml
---
### Defining Variables in Playbooks
- name: Create a user account
  hosts: webservers
  vars:
    username: dharani
    homepath: /home/dharani
  tasks:
    - name: Create the user {{ username }}
      user:
        name: "{{ username }}"
        home: "{{ homepath }}"

#Create user using variable file
[student@control1 varibles_vault_debug]$ cat createuser_using_variable_file.yml
---
#Create user using variable file
- name: Create user account using variable file
  hosts: webservers
  vars_files:
    - user_info.yml
  tasks:
    - name: Create a user {{ user }}
      user:
        name: "{{ user }}"
        home: "{{ home }}"
[student@control1 varibles_vault_debug]$

[student@control1 varibles_vault_debug]$ cat user_info.yml
user: alex
home: /home/alex
0
[student@control1 varibles_vault_debug]$ ansible-playbook createuser_using_variable_file.yml

PLAY [Create user account using variable file]
*********************************************************************************************************************************************************
*******

TASK [Gathering Facts]
*********************************************************************************************************************************************************
*******************************
ok: [host1.example.com]

TASK [Create a user alex]
*********************************************************************************************************************************************************
****************************
changed: [host1.example.com]

PLAY RECAP
*********************************************************************************************************************************************************
********************************************
host1.example.com          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
#Creating user using inventory variables (group vars)
[student@control1 varibles_vault_debug]$ cat createuser_using_inventory_variables.yml
---
#Creating user using inventory variables
- name: Create user using inventory variables
  hosts: datacenters
  tasks:
    - name: Create user {{ user }}
      user:
        name: "{{ user }}"
[student@control1 varibles_vault_debug]$ cat inventory
[webservers]
host1.example.com

[dbservers]
host2.example.com

[datacenters:children]
webservers
dbservers

[datacenters:vars]
user=ganesh
[student@control1 varibles_vault_debug]$ ansible-playbook createuser_using_inventory_variables.yml

PLAY [Create user using inventory variables]
**************************************************************************************************************************************
**************************

TASK [Gathering Facts]
**************************************************************************************************************************************
************************************************
ok: [host2.example.com]
ok: [host1.example.com]

TASK [Create user ganesh]
**************************************************************************************************************************************
*********************************************
changed: [host2.example.com]
changed: [host1.example.com]

PLAY RECAP
**************************************************************************************************************************************
************************************************************
host1.example.com          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
host2.example.com          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```



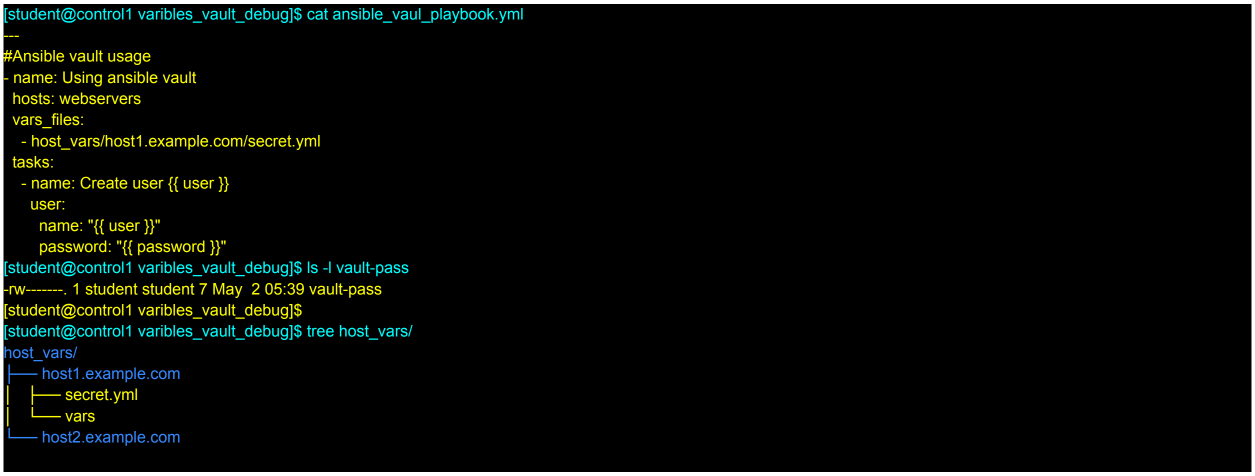
```
[student@control1 varibles_vault_debug]$ cat play_scope_variable_package_firewall.yml
---
#Create a playbook called playbook.yml which will configure webserver on host1.example.com with the help of the following variables which need to be written as play scope
- name: Create a playbook called playbook.yml which will configure webserver on host1.example.com with the help of the following variables which need to be written as play scope
  hosts: webservers
  vars:
    web_pkg: httpd
    web_service: httpd
    python_pkg: python3-PyMySQL
    rule: http
  tasks:
    - name: Install {{ web_pkg }} and {{ python_pkg }} packages
      yum:
        name:
          - "{{ web_pkg }}"
          - "{{ python_pkg }}"
        state: present

    - name: Copy the install.html file from files directory with proper permission
      copy:
        src: files/install.html
        dest: var/www/html
        owner: root
        group: root
        mode: '0644'

    - name: Start and enable {{web_service }} service
      service:
        name: "{{web_service }}"
        state: started

    - name: Configure firewall rule for "{{ rule }}" service
      firewalld:
        service: "{{ rule }}"
        permanent: yes
        state: enabled
[student@control1 varibles_vault_debug]$ ansible-playbook play_scope_variable_package_firewall.yml
PLAY [Create a playbook called playbook.yml which will configure webserver on host1.example.com with the help of the following variables which need to be written as play scope] ***************************************************************************************

TASK [Gathering Facts]
*************************************************************************************************************************************************************************************************************************************************
ok: [host1.example.com]

TASK [Install httpd and python3-PyMySQL packages]
****************************************************************************************************************************************************************************************************************************
ok: [host1.example.com]

TASK [Copy the install.html file from files directory with proper permission]
********************************************************************************************************************************************************************************
changed: [host1.example.com]

TASK [Start and enable httpd service]
*******************************************************************************************************************************************************************************************************************************
changed: [host1.example.com]

TASK [Configure firewall rule for "http" service]
****************************************************************************************************************************************************************************************************************************
ok: [host1.example.com]

PLAY RECAP
****************************************************************************************************************************************************************************************************************************************************************
host1.example.com          : ok=5    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[student@control1 varibles_vault_debug]$ cat play_scope_variable_package_firewall.yml
---
#Create a playbook called playbook.yml which will configure webserver on host1.example.com with the help of the following variables which need to be written as play scope
- name: Create a playbook called playbook.yml which will configure webserver on host1.example.com with the help of the following variables which need to be written as play scope
  hosts: webservers
  vars:
    web_pkg: httpd
    web_service: httpd
    python_pkg: python3-PyMySQL
    rule: http
  tasks:
    - name: Install {{ web_pkg }} and {{ python_pkg }} packages
      yum:
        name:
          - "{{ web_pkg }}"
          - "{{ python_pkg }}"
        state: present

    - name: Copy the install.html file from files directory with proper permission
      copy:
        src: files/install.html
        dest: /var/www/html/index.html
        owner: root
        group: root
        mode: '0644'

    - name: Start and enable {{web_service }} service
      service:
        name: "{{web_service }}"
        state: started

    - name: Configure firewall rule for "{{ rule }}" service
      firewalld:
        service: "{{ rule }}"
        permanent: yes
        state: enabled
- name: Create a second play in the same playbook which will check access to the web server and display the result with help with 'register' directive.
  hosts: webservers
  tasks:
    - name: check access to the web server and display the result with help with 'register' directive.
      uri:
        url: http://host1.example.com:80
        return_content: True
      register: result

    - name: debug the output
      debug:
        var: result
[student@control1 varibles_vault_debug]$ ansible-playbook play_scope_variable_package_firewall.yml
PLAY [Create a playbook called playbook.yml which will configure webserver on host1.example.com with the help of the following variables which need to be written as play scope] ********************************************************************************************

TASK [Gathering Facts]
************************************************************************************************************************************************************************************************************************************************************************

ok: [host1.example.com]

TASK [Install httpd and python3-PyMySQL packages]
*****************************************************************************************************************************************************************************************************************************

ok: [host1.example.com]

TASK [Copy the install.html file from files directory with proper permission]
*******************************************************************************************************************************************************************************

ok: [host1.example.com]

TASK [Start and enable httpd service]
******************************************************************************************************************************************************************************************************************************************

ok: [host1.example.com]

TASK [Configure firewall rule for "http" service]
*****************************************************************************************************************************************************************************************************************************

ok: [host1.example.com]

PLAY [Create a second play in the same playbook which will check access to the web server and display the result with help with 'register' directive.]
************************************************************************************************************************
```

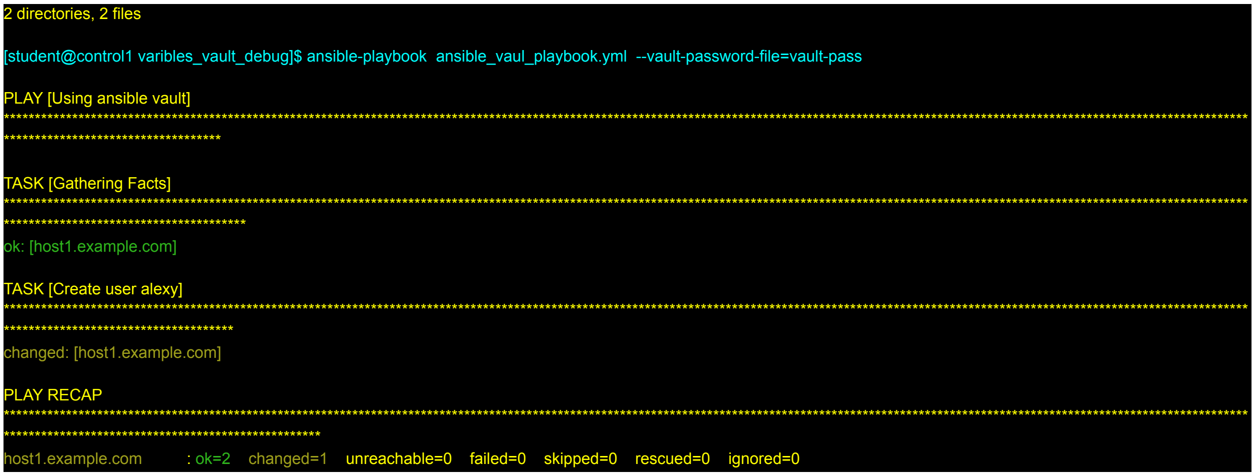


```
TASK [Gathering Facts] ********************************************************************************************************************************************************************************************
*****************************************
ok: [host1.example.com]

TASK [check access to the web server and display the result with help with 'register' directive.] ****************************************************************************************************************
ok: [host1.example.com]

TASK [debug the output] *******************************************************************************************************************************************************************************************
****************************************
ok: [host1.example.com] => {
    "result": {
        "accept_ranges": "bytes",
        "changed": false,
        "connection": "close",
        "content": "Welcome to hello world\n",
        "content_length": "23",
        "content_type": "text/html; charset=UTF-8",
        "cookies": {},
        "cookies_string": "",
        "date": "Sun, 09 May 2021 12:00:31 GMT",
        "elapsed": 0,
        "etag": "\"17-5c1e41e800c20\"",
        "failed": false,
        "last_modified": "Sun, 09 May 2021 11:39:51 GMT",
        "msg": "OK (23 bytes)",
        "redirected": false,
        "server": "Apache/2.4.37 (centos)",
        "status": 200,
        "url": "http://host1.example.com:80"
    }
}

PLAY RECAP ********************************************************************************************************************************************************************************************************
*****************************************************
host1.example.com          : ok=8    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

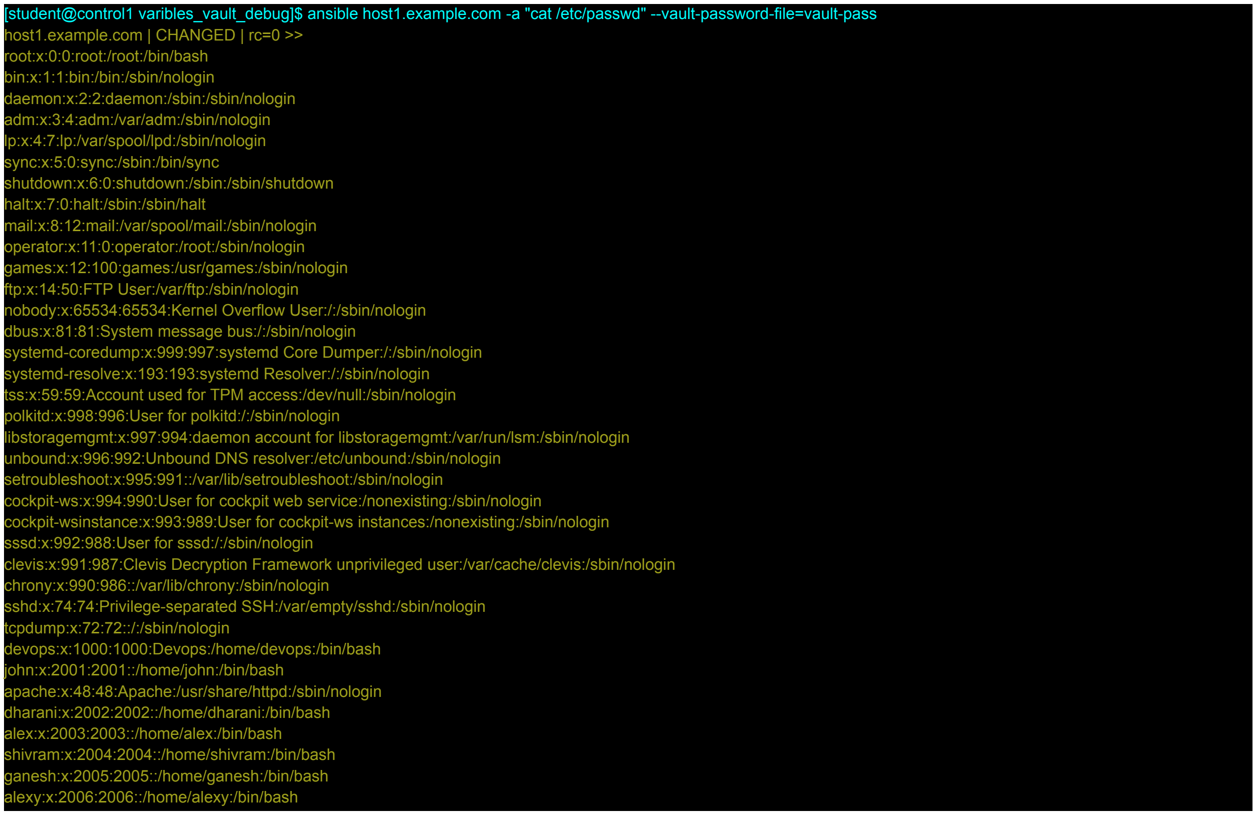
Task: Create a new playbook called newusers.yml which will create a new user called alex with a password 'redhat'. Use 'openssl passwd -6' command to generate the SHA512 encrypted hash for the password redhat. Use the username and and the password hash as two diffent variables kept in secret.yml file which is encrypted with ansible-vault. The playbook should use the variables from the secret.yml file. Also while getting executed it should not ask for any password.



```
[student@control1 varibles_vault_debug]$ cat ansible_vaul_playbook.yml
---
#Ansible vault usage
- name: Using ansible vault
  hosts: webservers
  vars_files:
    - host_vars/host1.example.com/secret.yml
  tasks:
    - name: Create user {{ user }}
      user:
        name: "{{ user }}"
        password: "{{ password }}"
[student@control1 varibles_vault_debug]$ ls -l vault-pass
-rw-------. 1 student student 7 May  2 05:39 vault-pass
[student@control1 varibles_vault_debug]$
[student@control1 varibles_vault_debug]$ tree host_vars/
host_vars/
├── host1.example.com
│   ├── secret.yml
│   └── vars
└── host2.example.com
```



```
2 directories, 2 files

[student@control1 varibles_vault_debug]$ ansible-playbook  ansible_vaul_playbook.yml  --vault-password-file=vault-pass

PLAY [Using ansible vault] ********************************************************************************************************************************************************************
************************************

TASK [Gathering Facts] ********************************************************************************************************************************************************************
****************************************
ok: [host1.example.com]

TASK [Create user alexy] ********************************************************************************************************************************************************************
**************************************
changed: [host1.example.com]

PLAY RECAP ********************************************************************************************************************************************************************
****************************************************
host1.example.com          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```



```
[student@control1 varibles_vault_debug]$ ansible host1.example.com -a "cat /etc/passwd" --vault-password-file=vault-pass
host1.example.com | CHANGED | rc=0 >>
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
systemd-coredump:x:999:997:systemd Core Dumper:/:/sbin/nologin
systemd-resolve:x:193:193:systemd Resolver:/:/sbin/nologin
tss:x:59:59:Account used for TPM access:/dev/null:/sbin/nologin
polkitd:x:998:996:User for polkitd:/:/sbin/nologin
libstoragemgmt:x:997:994:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
unbound:x:996:992:Unbound DNS resolver:/etc/unbound:/sbin/nologin
setroubleshoot:x:995:991::/var/lib/setroubleshoot:/sbin/nologin
cockpit-ws:x:994:990:User for cockpit web service:/nonexisting:/sbin/nologin
cockpit-wsinstance:x:993:989:User for cockpit-ws instances:/nonexisting:/sbin/nologin
sssd:x:992:988:User for sssd:/:/sbin/nologin
clevis:x:991:987:Clevis Decryption Framework unprivileged user:/var/cache/clevis:/sbin/nologin
chrony:x:990:986::/var/lib/chrony:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
devops:x:1000:1000:Devops:/home/devops:/bin/bash
john:x:2001:2001::/home/john:/bin/bash
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
dharani:x:2002:2002::/home/dharani:/bin/bash
alex:x:2003:2003::/home/alex:/bin/bash
shivram:x:2004:2004::/home/shivram:/bin/bash
ganesh:x:2005:2005::/home/ganesh:/bin/bash
alexy:x:2006:2006::/home/alexy:/bin/bash
```