

Approx time to complete this assignment: 12 hours

I have a centralized sudoers file across all machines. This is what my sudoers file looks like:

```
#Host Aliases
Host_Alias MACHINEE = 10.21.32.2
Host_Alias DMZMACHINES = 100.64.26.2,100.64.26.3,100.64.26.4,100.64.26.6

#Command Aliases
Cmnd_Alias SHUTDOWN_CMDS=/usr/sbin/shutdown -h [1-9][2-9][0-9],
/usr/sbin/shutdown -h [2-9][0-9][0-9]
Cmnd_Alias CANCEL_CMDS=/usr/sbin/shutdown -c
Cmnd_Alias CHOWN_CMD=/usr/bin/chown [a-zA-Z0-9_/*, !/usr/bin/chown
*\:root *, !/usr/bin/chown root\:* *, !/usr/bin/chown root *

#Grant Dwight full access on all machines
dschrute ALL=(ALL) ALL

#Grant Michael access to shutdown on B,C,D,F machines
mscott DMZMACHINES=(ALL) SHUTDOWN_CMDS, CANCEL_CMDS

#Grant Jim access to chown on Machine E
jhalpert MACHINEE=(ALL) CHOWN_CMD
```

1. Set up password-less logins for Pam Beesly, Kelly Kapoor and Andy Bernard from Machine E to Machine C.

Created public and private keys (via `ssh-keygen -t rsa` command) for **pbeesly**, **kkapoor**, **abernard** and stored them in their respective home directories in `~/.ssh` folder.

Copied the public keys to Machine C by issuing

```
ssh-copy-id pbeesly@100.64.26.1
```

command. Ran the same for **kkapoor** and **abernard**.

-
2. Allow Dwight Schrute to run any command via sudo on any machine using a method other than adding him to the sudo or wheel group.

I ran this command on all machines:

```
for i in {1..6}; do [[ $i == 5 ]] && continue; ssh root@100.64.26.$i  
echo "dschrute ALL=(ALL) ALL" >> /etc/sudoers.d/custom; done
```

For FreeBSD:

```
ssh root@100.64.26.7 echo "dschrute ALL=(ALL) ALL" >>  
/usr/local/etc/sudoers.d/custom
```

I noticed that in `/etc/sudoers`, there's a line that mentions

```
#includedir /etc/sudoers.d
```

This implied that every file within the `/etc/sudoers.d` directory would be incorporated into `/etc/sudoers`. As a result, I generated a 'custom' file on all machines.

Post this, I checked for any syntax errors by running

```
visudo -c /etc/sudoers.d/custom
```

-
3. Allow Michael Scott to shut down any of the DMZ machines (B,C,D,F) with no less than two hours notice to other users. The shutdown command he will use is `sudo shutdown -h` followed by the number of minutes (an integer). As an upper limit you can assume that the shut down delay will be no more than 999 minutes. Michael should also be able to cancel the pending shutdown.

I created a command alias in the sudoers file which includes all the commands to shutdown between 120 minutes to 999 minutes

```
Cmnd_Alias SHUTDOWN_CMDS=/usr/sbin/shutdown -h [1-9][2-9][0-9],
```

```
/usr/sbin/shutdown -h [2-9][0-9][0-9]
```

And added a command alias to cancel the shutdown

```
Cmnd_Alias CANCEL_CMDS=/usr/sbin/shutdown -c
```

And granted mscott access to these command aliases to run on B,C,D,F machines.

```
mscott ALL=(ALL) SHUTDOWN_CMDS, CANCEL_CMDS
```

-
- 4. Allow Jim Halpert to change the ownership of files on Machine E to another user, but not the user root. So he should, for example, be able to make pbeesly the owner of a file, but not root.**

I added the following line on Machine E to grant chown access to jhalpert.

```
jhalpert ALL=(ALL) /usr/bin/chown [a-zA-Z0-9_]*, !/usr/bin/chown *\:root *, !/usr/bin/chown root\:* *, !/usr/bin/chown root *
```

The `[a-zA-Z0-9_]*` pattern encompasses any character string, including usernames with underscores or numbers.

The `:root *` and `root: *` patterns address situations where if jhalpert attempts commands like "sudo chown root:root <file>," "sudo chown jhalpert:root <file>," or "sudo chown root:jhalpert <file>," he should not be permitted to execute them. The "!" symbol negates these permissions.

The `root *` pattern tackles the scenario when jhalpert tries to execute "sudo chown root <file>". It disallows this action, and again, the "!" symbol negates it.

