Use Puppet's file server to distribute files to multiple nodes

Table of Contents

Introduction

Puppet's file server allows administrators to store and distribute files from the Puppet Master to multiple nodes. This simplifies the process of ensuring specific files or configurations are present and consistent across all managed nodes.

Problem Statement

Manually copying files to multiple nodes is time-consuming and error-prone. Using Puppet's file server, administrators can centralize file storage on the Puppet Master and automate the distribution of these files to Puppet Agents.

Prerequisites

Completion of all previous lab guides (up to Lab Guide-04) is required before proceeding with Lab Guide-05.

Software Requirements

- Puppet 3.8.7
- Puppet Master and Agent properly configured
- Files to be distributed (e.g., configuration files, scripts)

Hardware Requirements

- Puppet Master: 1GB RAM, 2 CPUs, 10GB Disk

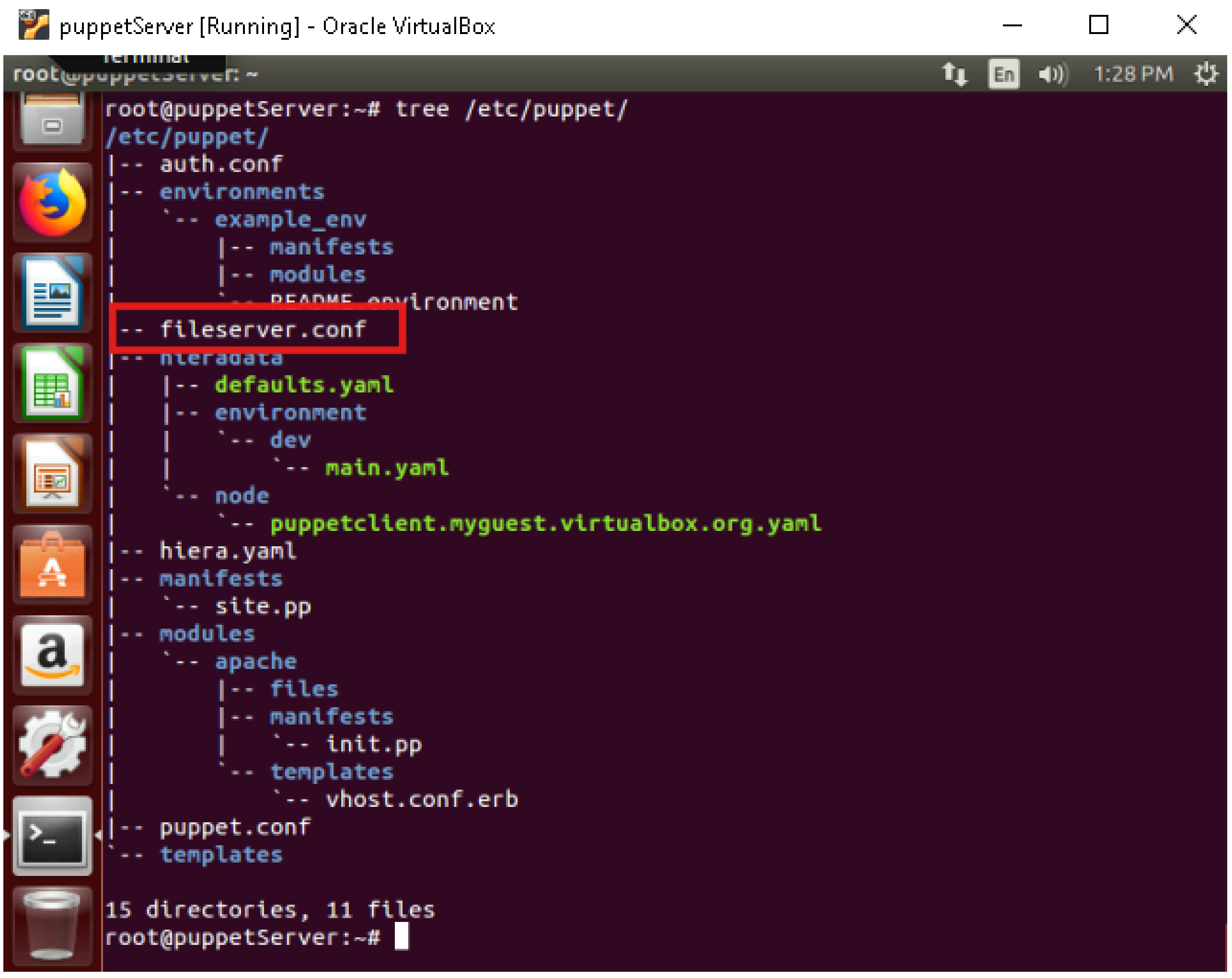
- Puppet Agent: 512MB RAM, 1 CPU, 5GB Disk

Implementation Steps

Step 1: Configure Puppet File Server

1. **Locate the `fileserver.conf` File**:

   The configuration file is typically located at `/etc/puppet/fileserver.conf`.

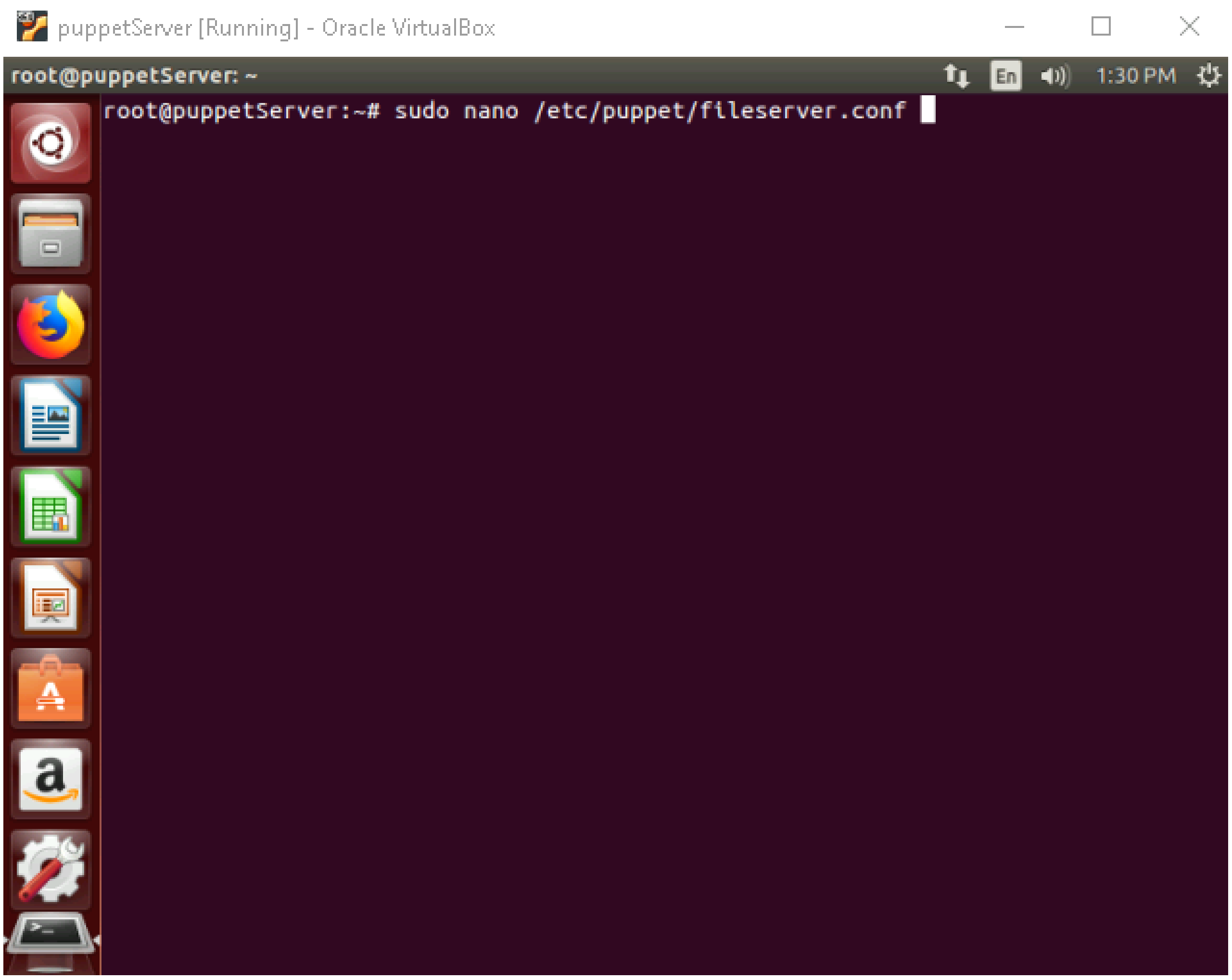


```
root@puppetServer:~# tree /etc/puppet/
/etc/puppet/
|-- auth.conf
|-- environments
|   `-- example_env
|       |-- manifests
|       |-- modules
|       `-- README.environment
|-- fileserver.conf
|-- hieradata
|   |-- defaults.yaml
|   |-- environment
|   |   `-- dev
|   |       `-- main.yaml
|   `-- node
|       `-- puppetclient.myguest.virtualbox.org.yaml
|-- hiera.yaml
|-- manifests
|   `-- site.pp
|-- modules
|   `-- apache
|       |-- files
|       |-- manifests
|       |   `-- init.pp
|       `-- templates
|           `-- vhost.conf.erb
|-- puppet.conf
`-- templates

15 directories, 11 files
root@puppetServer:~#
```

2. **Define a File Mount Point**:

   Open the file and configure a mount point for your files:

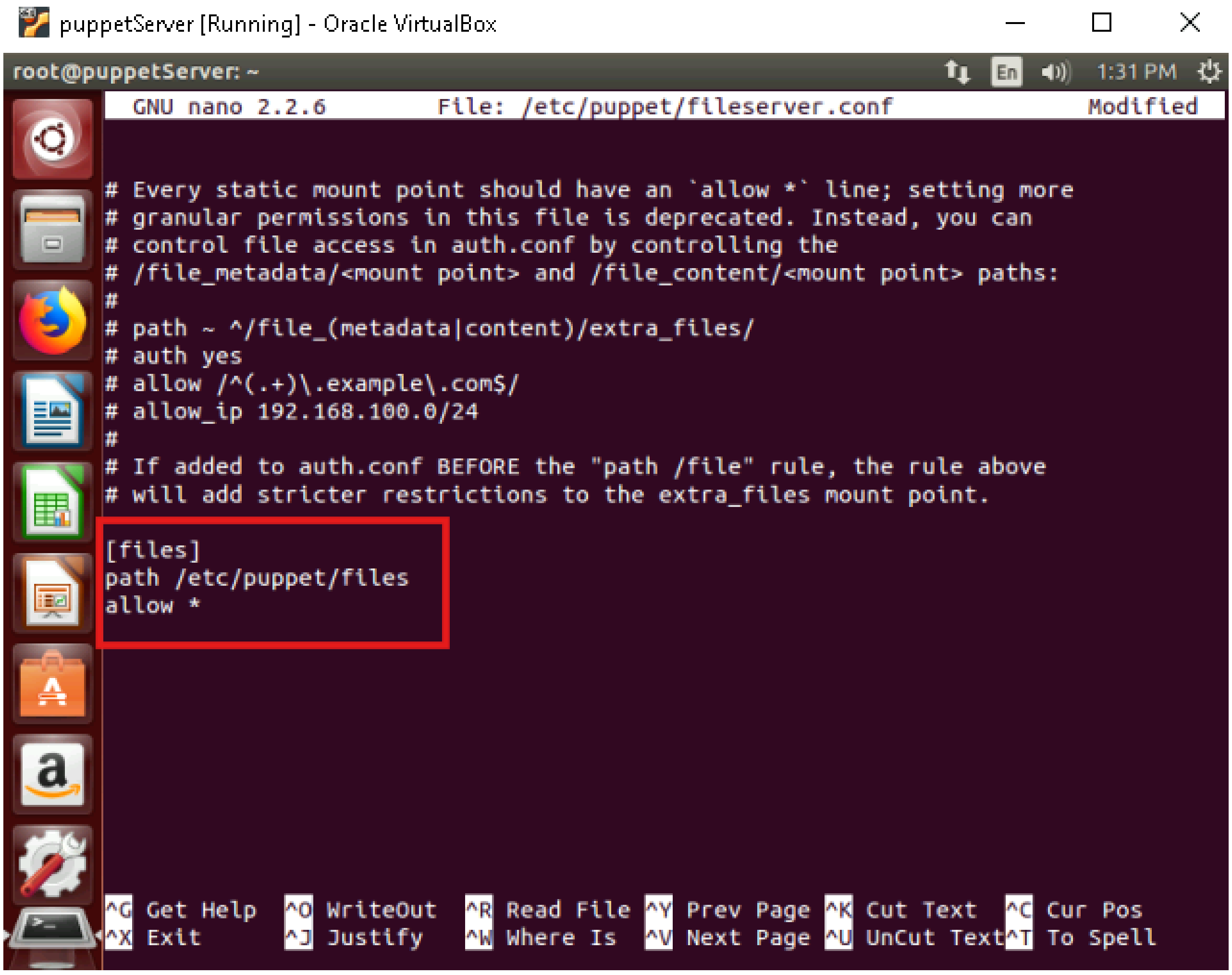
```
sudo nano /etc/puppet/fileserver.conf
```



```
root@puppetServer:~# sudo nano /etc/puppet/fileserver.conf
```

Add the following:

```
[files]
path /etc/puppet/files
allow *
```



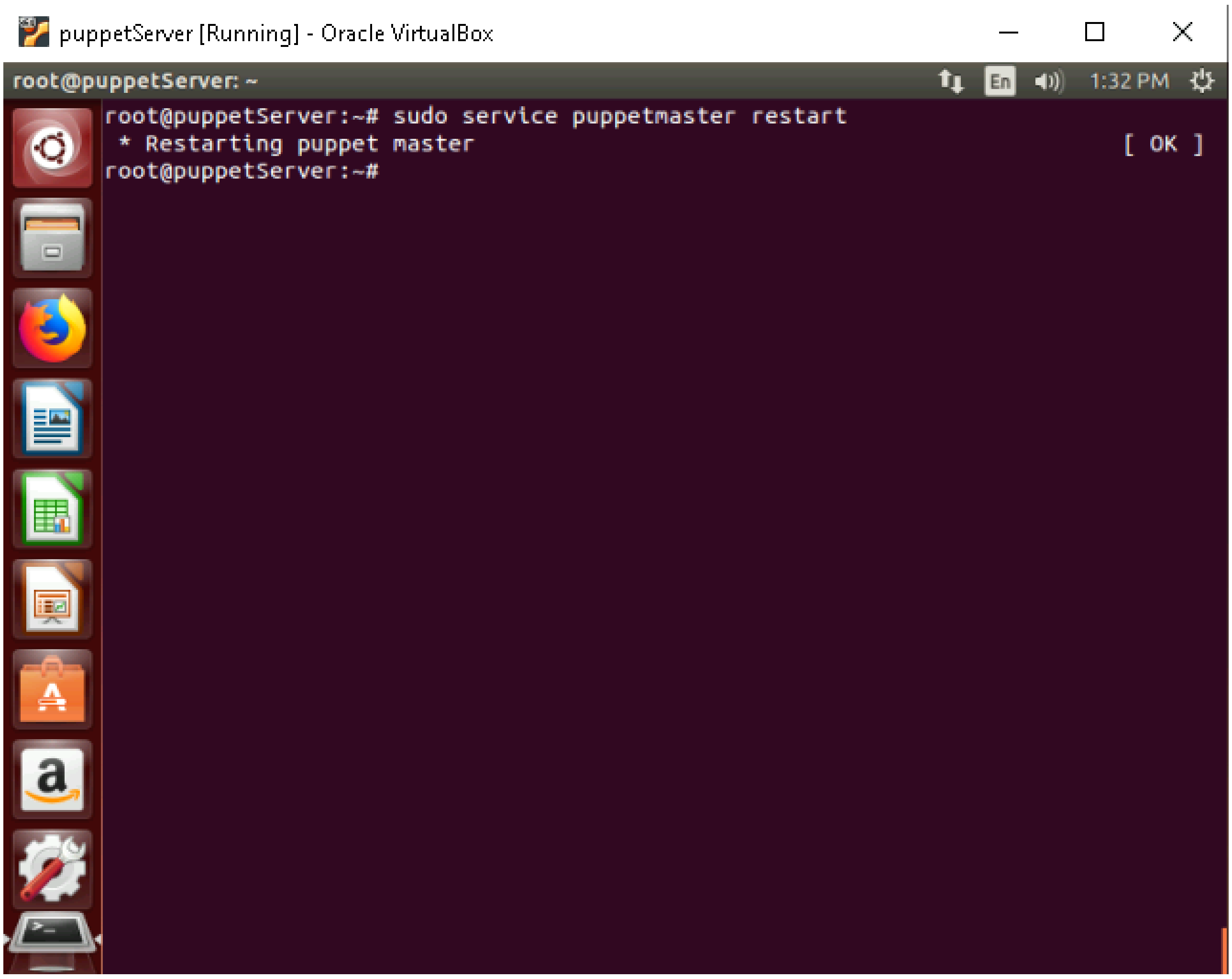
```
puppetServer [Running] - Oracle VirtualBox
root@puppetServer: ~                                              1:31 PM
  GNU nano 2.2.6          File: /etc/puppet/fileserver.conf             Modified


# Every static mount point should have an `allow *` line; setting more
# granular permissions in this file is deprecated. Instead, you can
# control file access in auth.conf by controlling the
# /file_metadata/<mount point> and /file_content/<mount point> paths:
#
# path ~ ^/file_(metadata|content)/extra_files/
# auth yes
# allow /^(.+)\.example\.com$/
# allow_ip 192.168.100.0/24
#
# If added to auth.conf BEFORE the "path /file" rule, the rule above
# will add stricter restrictions to the extra_files mount point.

[files]
path /etc/puppet/files
allow *

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

- **Explanation**:
  - `[files]`: The mount point name.
  - `path`: Directory where files to be distributed are stored.
  - `allow *`: Allows all nodes to access this mount point.

3. **Restart the Puppet Server**:

   After making changes, restart the Puppet Server to apply them:

```
sudo service puppetmaster restart
```



```
root@puppetServer:~# sudo service puppetmaster restart
 * Restarting puppet master                                              [ OK ]
root@puppetServer:~#
```

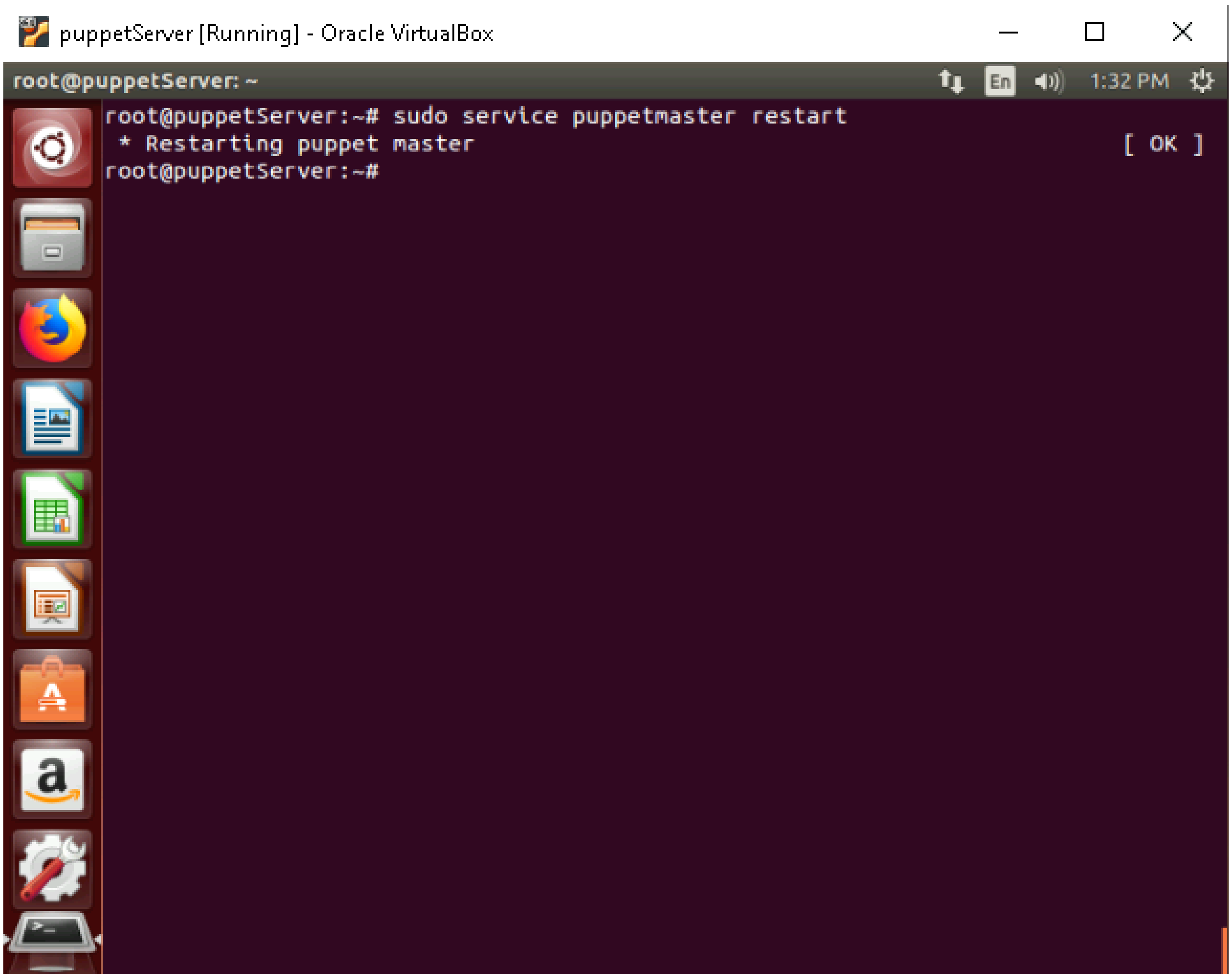
- **Explanation**: The Puppet Server reloads the configuration files, including the file server configuration.

Step 2: Place Files on Puppet Master

1. **Create the Directory for Files**:

   Ensure the directory specified in `fileserver.conf` exists:

   ```
   sudo mkdir -p /etc/puppet/files
   ```



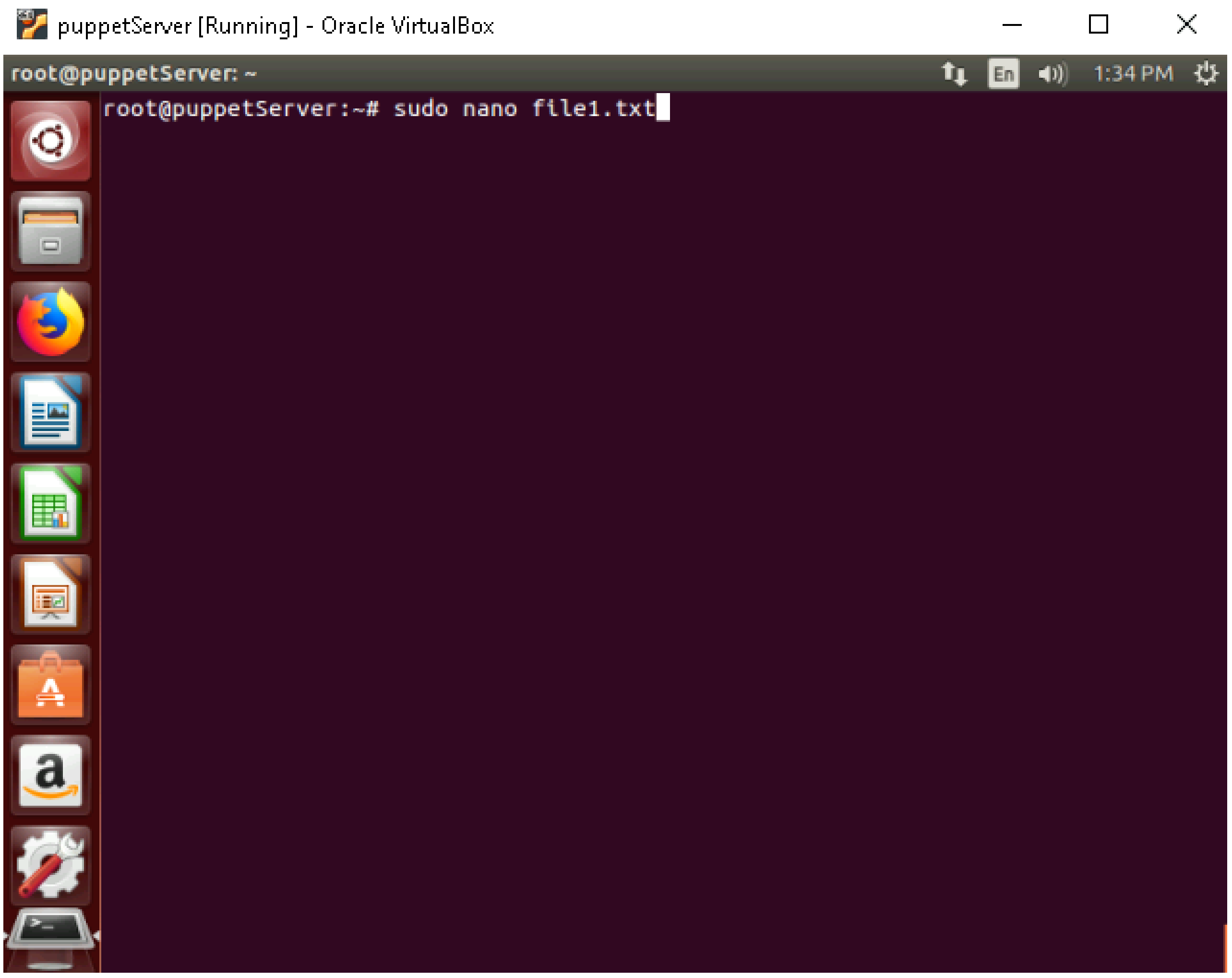
```
root@puppetServer:~# sudo service puppetmaster restart
 * Restarting puppet master                                              [ OK ]
root@puppetServer:~#
```

2. **Create Files to Distribute and Copy to Puppet Master file Directory**:

   Create the files:

   file1.txt:

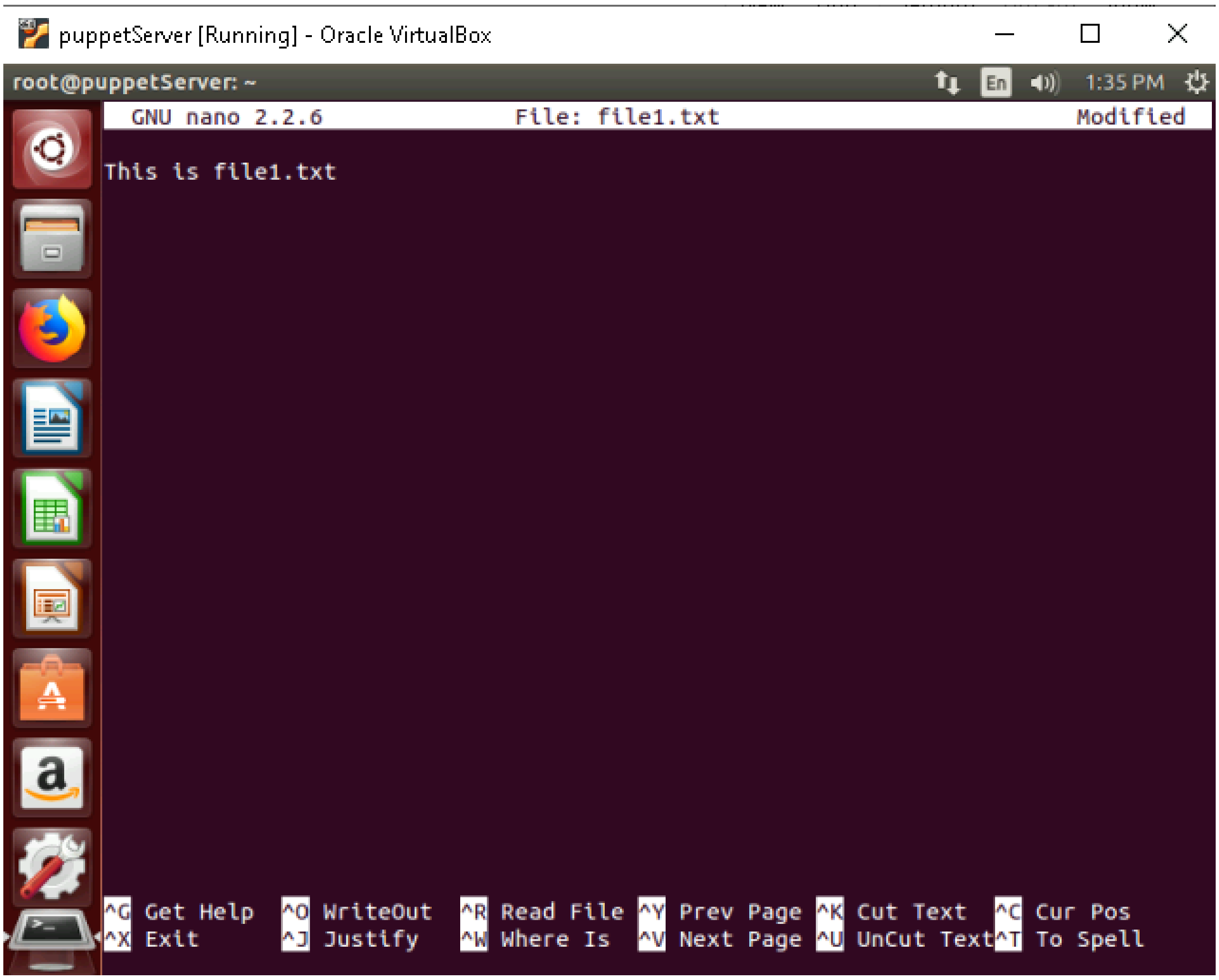
   ```
   sudo nano file1.txt
   ```



```
root@puppetServer:~# sudo nano file1.txt
```

Add the following content:

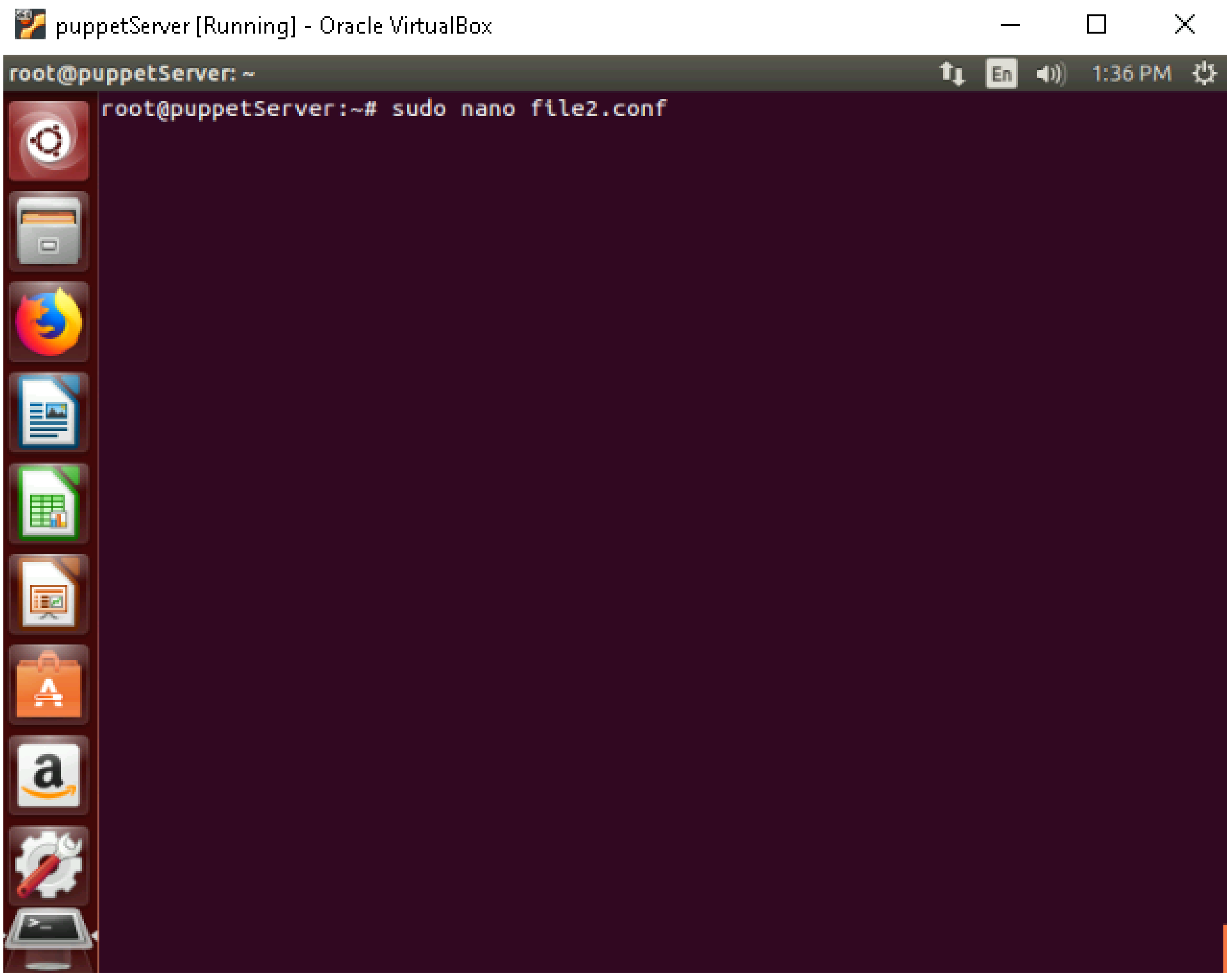
```
This is file1.txt
```



```
  GNU nano 2.2.6              File: file1.txt                          Modified

This is file1.txt

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

file2.conf:

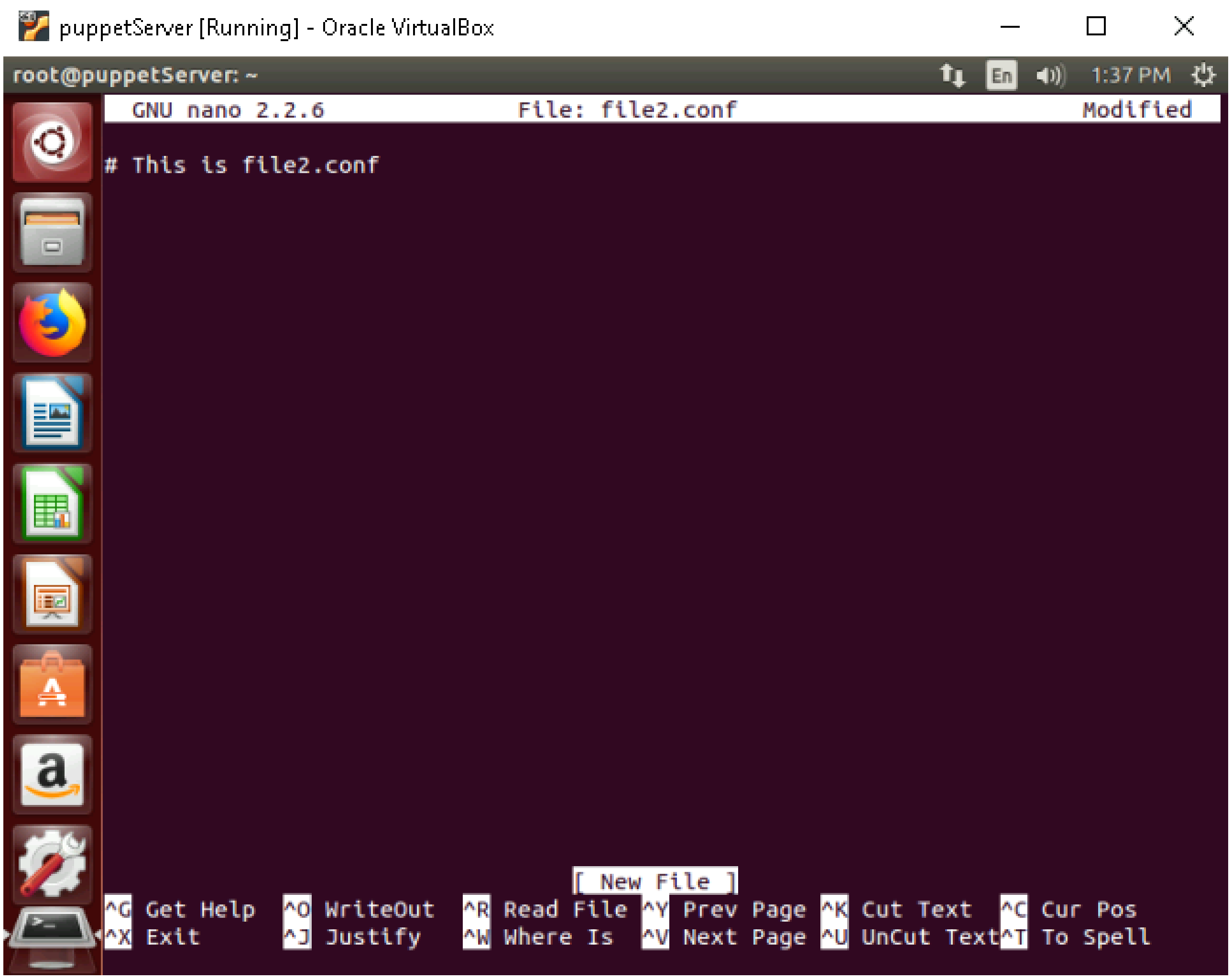
```
sudo nano file2.conf
```



```
root@puppetServer:~# sudo nano file2.conf
```

Add the following content:

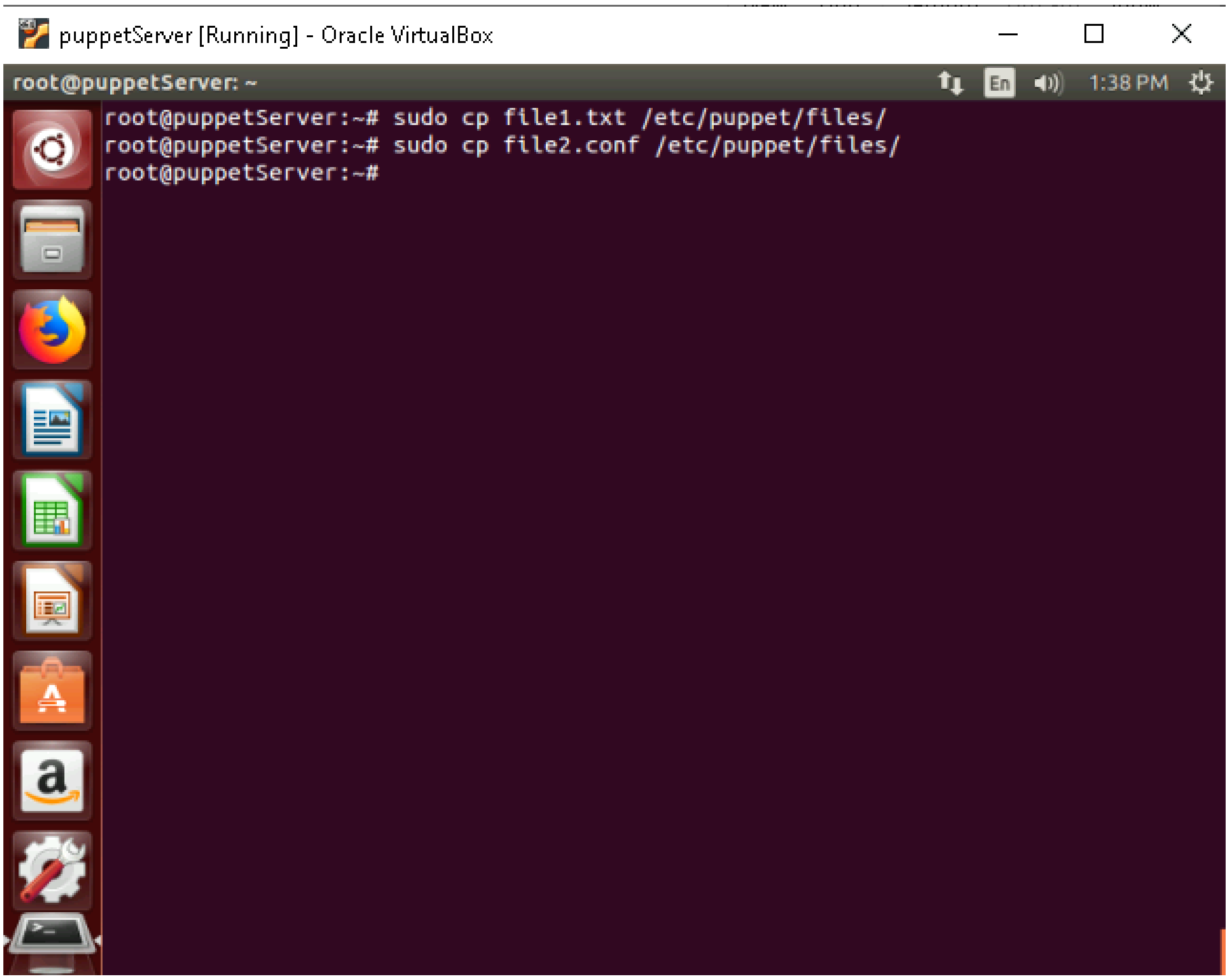
```
# This is file2.conf
```



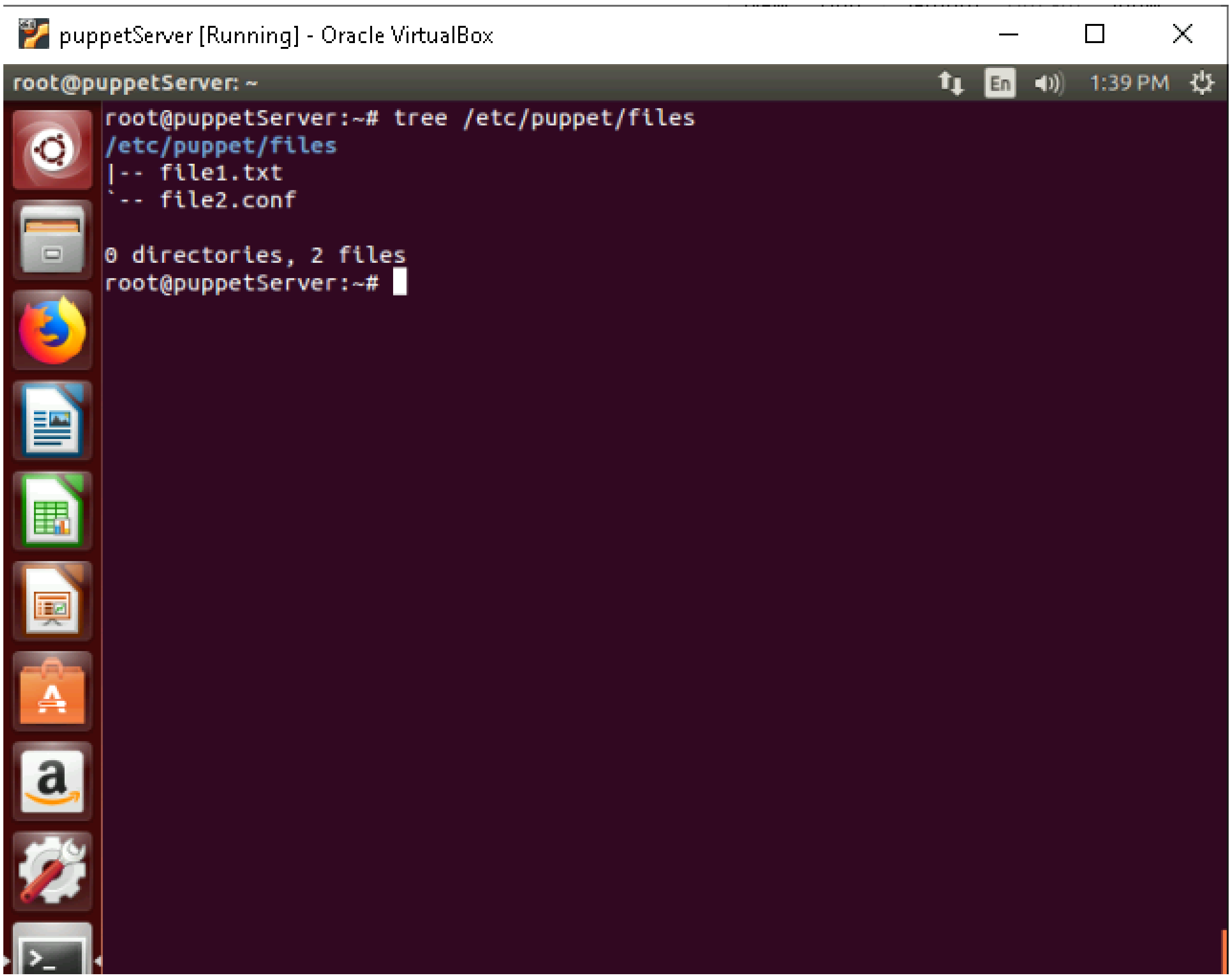
```
  GNU nano 2.2.6              File: file2.conf                        Modified

# This is file2.conf

                                 [ New File ]
^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text^T To Spell
```

Copy the files to the Puppet Master file directory:

```
sudo cp file1.txt /etc/puppet/files/
sudo cp file2.conf /etc/puppet/files/
```



```
root@puppetServer:~# sudo cp file1.txt /etc/puppet/files/
root@puppetServer:~# sudo cp file2.conf /etc/puppet/files/
root@puppetServer:~#
```

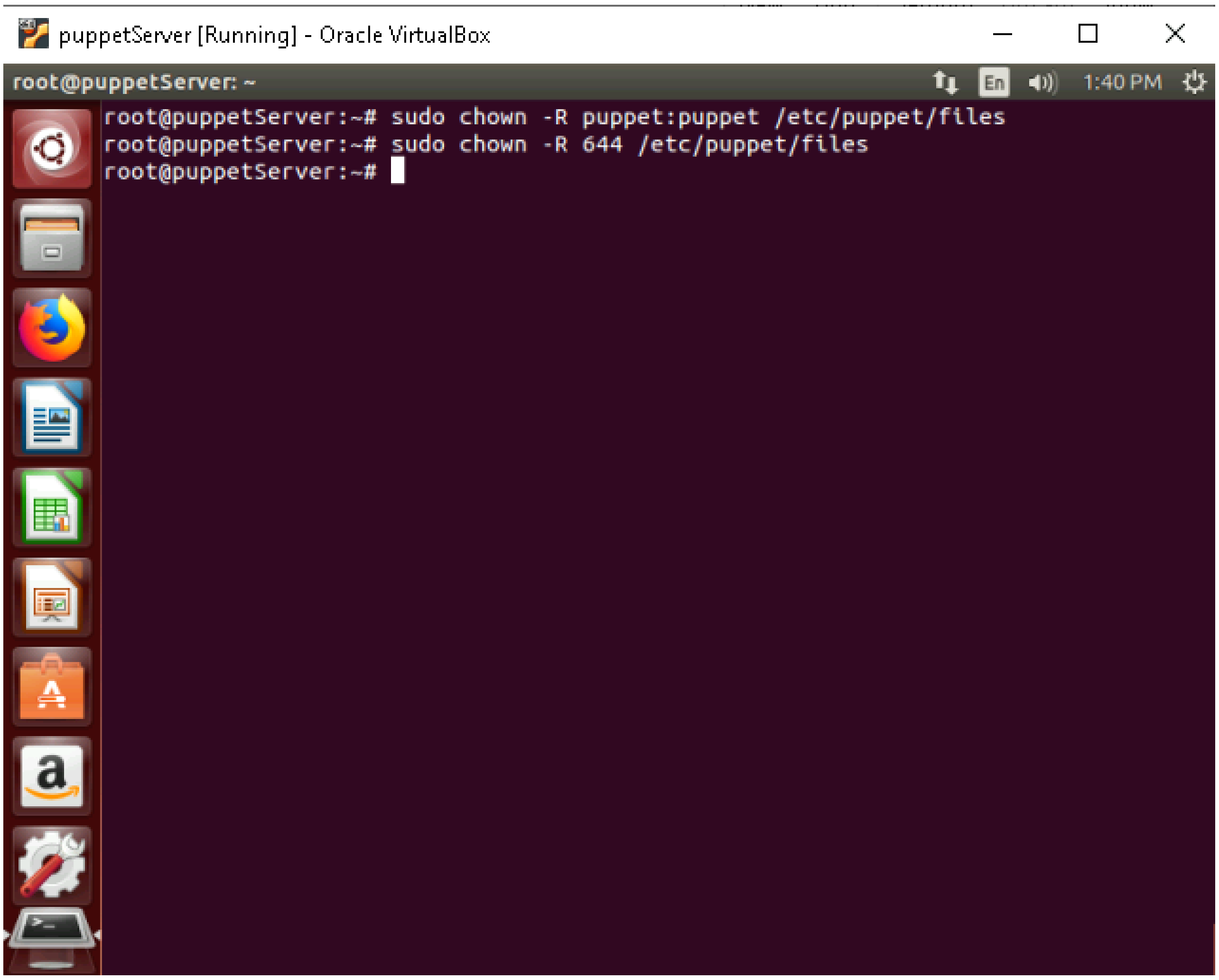


```
root@puppetServer:~# tree /etc/puppet/files
/etc/puppet/files
|-- file1.txt
`-- file2.conf

0 directories, 2 files
root@puppetServer:~#
```

3. **Set Permissions**:

   Ensure the files and directories are readable by the Puppet user:

   ```
   sudo chown -R puppet:puppet /etc/puppet/files
   sudo chmod -R 644 /etc/puppet/files
   ```



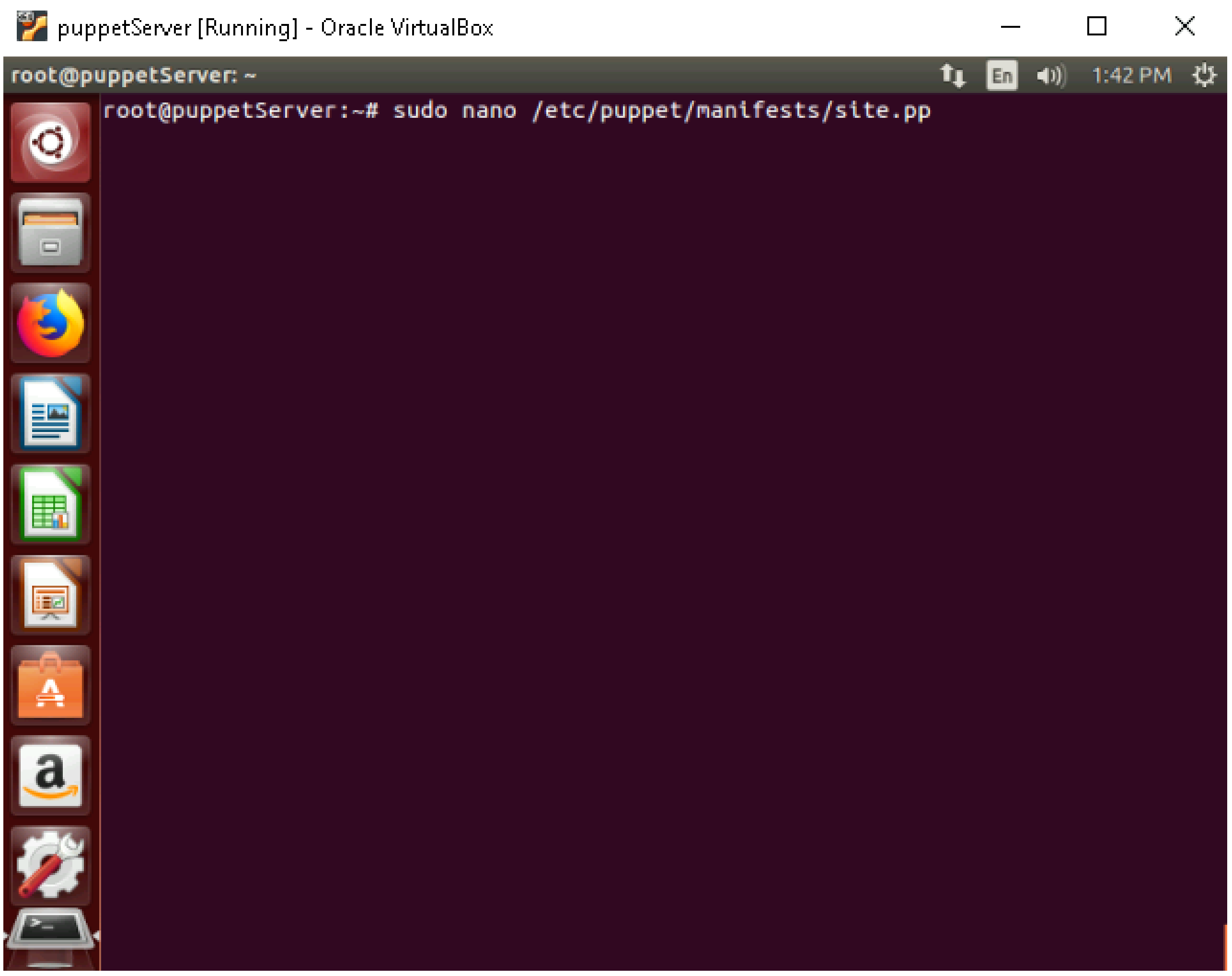
```
root@puppetServer:~# sudo chown -R puppet:puppet /etc/puppet/files
root@puppetServer:~# sudo chown -R 644 /etc/puppet/files
root@puppetServer:~# 
```

Step 3: Write a Puppet Manifest to Distribute Files

1. **Create or Edit the Manifest File**:

   Create a new manifest file or modify an existing one:

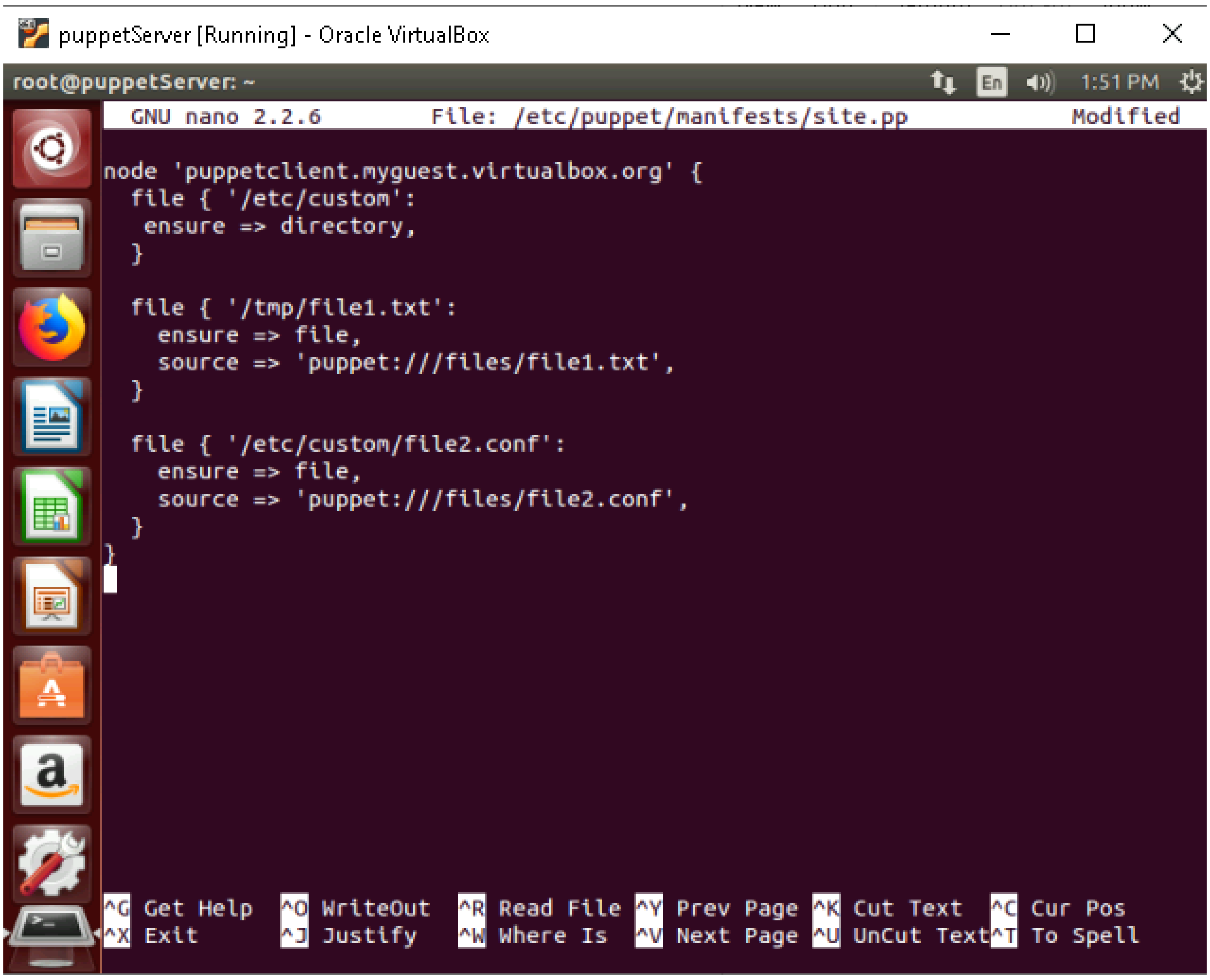
   ```
   sudo nano /etc/puppet/manifests/site.pp
   ```



```
root@puppetServer:~# sudo nano /etc/puppet/manifests/site.pp
```

2. **Add a File Resource**: Use the `file` resource to manage the distribution of files:

```
node 'puppetclient.myguest.virtualbox.org' {
  file { '/etc/custom':
    ensure => directory,
  }

  file { '/tmp/file1.txt':
    ensure => file,
    source => 'puppet:///files/file1.txt',
  }

  file { '/etc/custom/file2.conf':
    ensure => file,
    source => 'puppet:///files/file2.conf',
  }
}
```



```
  GNU nano 2.2.6          File: /etc/puppet/manifests/site.pp          Modified

node 'puppetclient.myguest.virtualbox.org' {
  file { '/etc/custom':
   ensure => directory,
  }

  file { '/tmp/file1.txt':
    ensure => file,
    source => 'puppet:///files/file1.txt',
  }

  file { '/etc/custom/file2.conf':
    ensure => file,
    source => 'puppet:///files/file2.conf',
  }
}

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

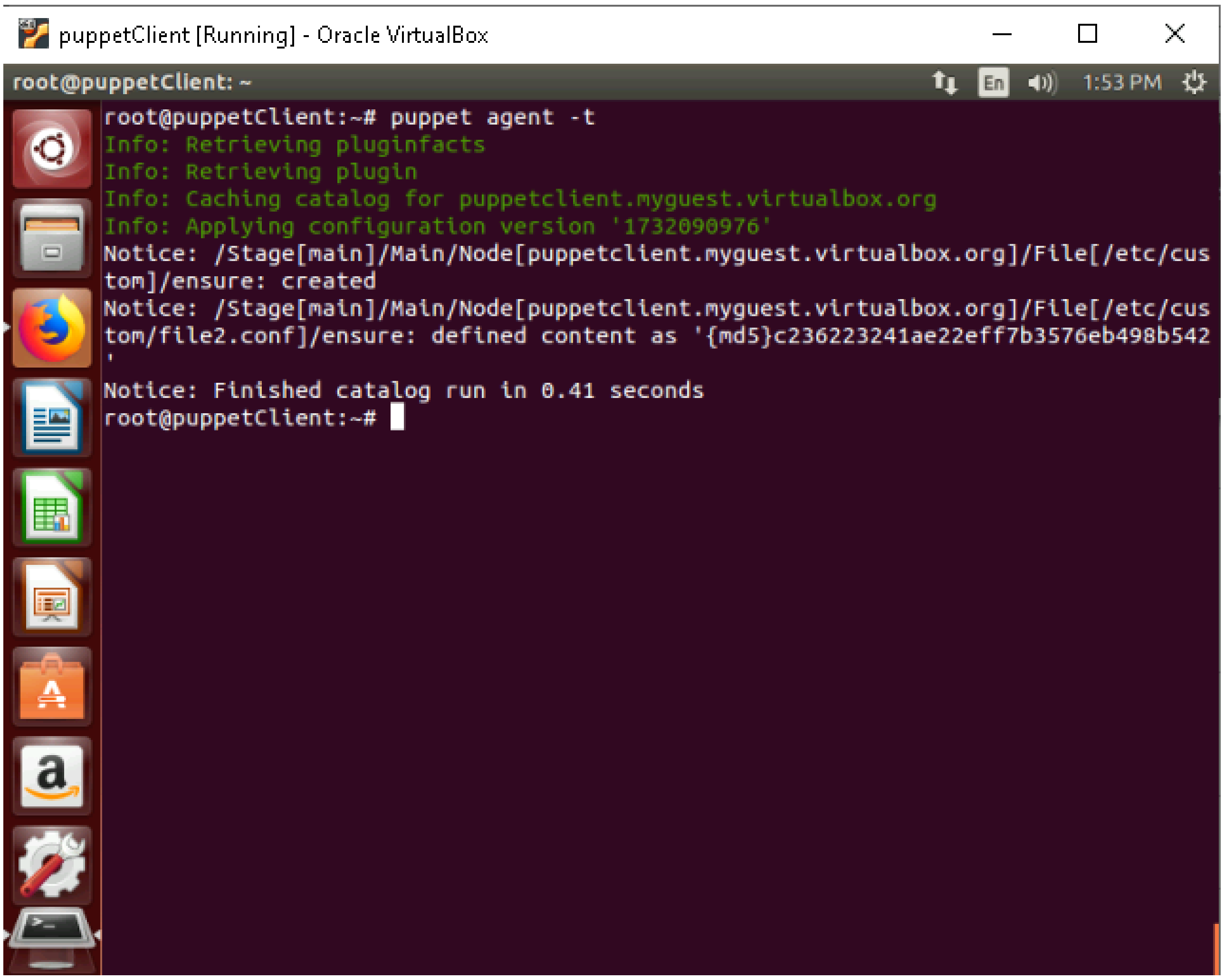
- **Explanation**:
  - `ensure`: Ensures the file exists.
  - `source`: Specifies the file's location on the Puppet Master using the `puppet:///` URI.

Step 4: Apply the Configuration

1. **Trigger Puppet Agent Run**:

   On the Puppet Agent node, run:

   ```
   puppet agent -t
   ```



```
root@puppetClient:~# puppet agent -t
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Caching catalog for puppetclient.myguest.virtualbox.org
Info: Applying configuration version '1732090976'
Notice: /Stage[main]/Main/Node[puppetclient.myguest.virtualbox.org]/File[/etc/cus
tom]/ensure: created
Notice: /Stage[main]/Main/Node[puppetclient.myguest.virtualbox.org]/File[/etc/cus
tom/file2.conf]/ensure: defined content as '{md5}c236223241ae22eff7b3576eb498b542
'
Notice: Finished catalog run in 0.41 seconds
root@puppetClient:~#
```

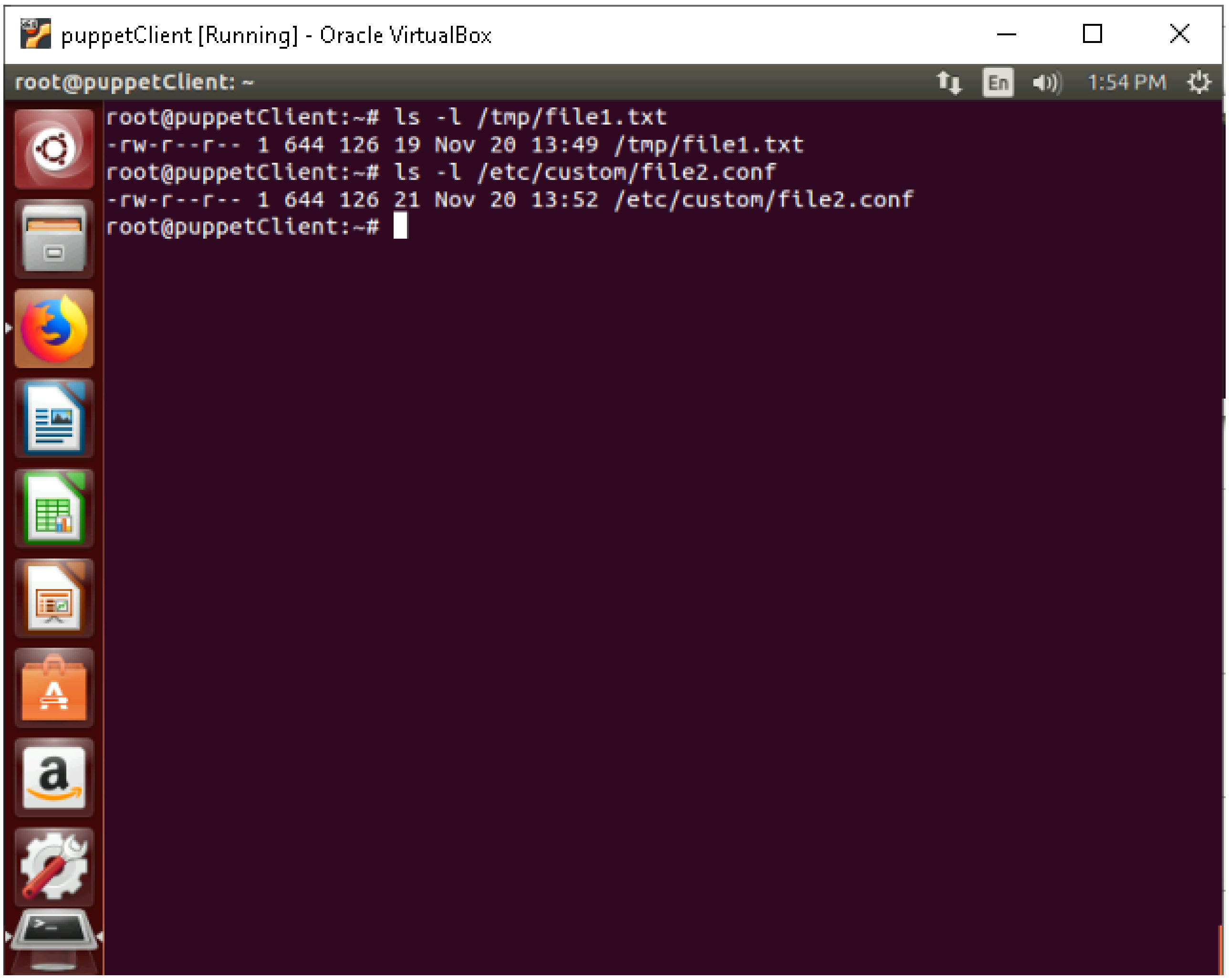
- **Explanation**: The agent fetches the catalog from the Puppet Master and applies the configuration, including downloading the specified files.

Step 5: Verify File Distribution

1. **Check Files on the Agent**:

   Verify that the files are distributed to the correct locations:

   ```
   ls -l /tmp/file1.txt
   ls -l /etc/custom/file2.conf
   ```

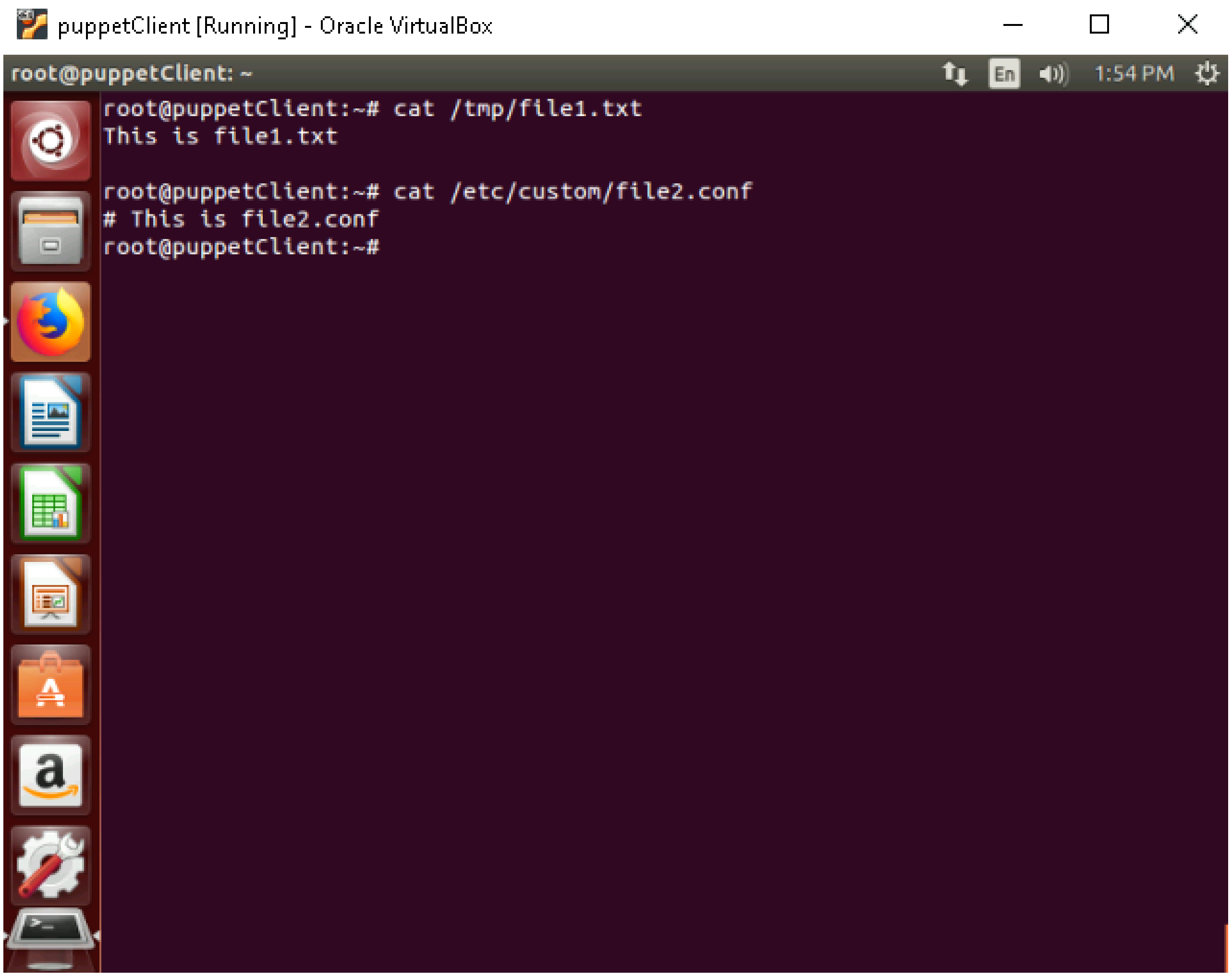


```
root@puppetClient:~# ls -l /tmp/file1.txt
-rw-r--r-- 1 644 126 19 Nov 20 13:49 /tmp/file1.txt
root@puppetClient:~# ls -l /etc/custom/file2.conf
-rw-r--r-- 1 644 126 21 Nov 20 13:52 /etc/custom/file2.conf
root@puppetClient:~#
```

2. **Compare File Contents**:

   Confirm the file contents match the originals:

```
cat /tmp/file1.txt
cat /etc/custom/file2.conf
```



```
root@puppetClient:~# cat /tmp/file1.txt
This is file1.txt

root@puppetClient:~# cat /etc/custom/file2.conf
# This is file2.conf
root@puppetClient:~#
```

References

- Puppet File Server
- Puppet File Server-2
