Version : **2022.01**

Dernière mise-à-jour : 2020/01/30 03:45

DOF405 - Puppet en mode Agent/Maître

Contenu du Module

Préparation

Les trois machines virtuelles **PuppetMaster**, **PuppetSlave01** et **PuppetSlave02** ont été configurées selon le tableau ci-dessous :

Machine	Nom d'hôte	Adresse IP	OS	RAM
PuppetMaster	master.i2tch.loc	10.0.2.59	Ubuntu 18.04	4096 Mo
PuppetSlave01	slave01.i2tch.loc	10.0.2.68	Ubuntu 18.04	2048 Mo
PuppetSlave02	slave02.i2tch.loc	10.0.2.69	CentOS 7	1024 Mo

Les noms d'utilisateurs et les mots de passe sont identiques pour chaque machine :

Utilisateur	Mot de Passe
trainee	trainee
root	fenestros

Configuration du Fuseau d'Horaire

Configurez les trois machines virtuelles pour qu'elles soient sur le même fuseau d'horaire :

```
trainee@master:~$ su -
Password: fenestros
root@master:~# dpkg-reconfigure tzdata

Current default time zone: 'Europe/Paris'
Local time is now:      Wed Feb 12 14:11:40 CET 2020.
Universal Time is now:  Wed Feb 12 13:11:40 UTC 2020.
```

```
trainee@slave01:~$ su -
Password: fenestros
root@slave01:~# dpkg-reconfigure tzdata

Current default time zone: 'Europe/Paris'
```

```
Local time is now:      Wed Feb 12 14:12:21 CET 2020.
Universal Time is now:  Wed Feb 12 13:12:21 UTC 2020.
```

```
[trainee@slave02 ~]$ su -
Mot de passe : fenestros
Dernière connexion : mercredi 13 mars 2019 à 12:55:24 CET sur tty1
[root@slave02 ~]# timedatectl set-timezone 'Europe/Paris'
[root@slave02 ~]# date
Wed 12 Feb 14:12:51 CET 2020
```

Désactiver SELinux dans puppetslave02

```
[root@slave02 ~]# vi /etc/sysconfig/selinux
[root@slave02 ~]# cat /etc/sysconfig/selinux

# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=permissive
# SELINUXTYPE= can take one of three two values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted


[root@slave02 ~]# setenforce permissive
```

LAB #1 - Installer et Configurer Puppet Server

Installer puppetserver

Installez Puppet dans la machine virtuelle **PuppetMaster** :

```
root@master:~# wget https://apt.puppetlabs.com/puppet-release-bionic.deb
--2020-02-12 14:13:20--  https://apt.puppetlabs.com/puppet-release-bionic.deb
Resolving apt.puppetlabs.com (apt.puppetlabs.com)... 13.225.38.129, 13.225.38.76, 13.225.38.45, ...
Connecting to apt.puppetlabs.com (apt.puppetlabs.com)|13.225.38.129|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11724 (11K) [application/x-debian-package]
Saving to: ‘puppet-release-bionic.deb’

puppet-release-bionic.deb       100%[=================================================>]  11.45K  --.-KB/s
in 0s

2020-02-12 14:13:21 (346 MB/s) - ‘puppet-release-bionic.deb’ saved [11724/11724]

root@master:~# dpkg -i puppet-release-bionic.deb
Selecting previously unselected package puppet-release.
(Reading database ... 128539 files and directories currently installed.)
Preparing to unpack puppet-release-bionic.deb ...
Unpacking puppet-release (1.0.0-7bionic) ...
Setting up puppet-release (1.0.0-7bionic) ...

root@master:~# apt update
Hit:1 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelease
Get:5 http://apt.puppetlabs.com bionic InRelease [85.3 kB]
```

```
Get:6 http://apt.puppetlabs.com bionic/puppet amd64 Packages [39.9 kB]
Get:7 http://apt.puppetlabs.com bionic/puppet i386 Packages [16.4 kB]
Get:8 http://apt.puppetlabs.com bionic/puppet all Packages [16.4 kB]
Fetched 158 kB in 2s (83.9 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
624 packages can be upgraded. Run 'apt list --upgradable' to see them.

root@master:~# apt install puppetserver
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java java-common net-tools openjdk-8-jre-headless puppet-agent
Suggested packages:
  default-jre fonts-dejavu-extra fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei fonts-wqy-zenhei
The following NEW packages will be installed:
  ca-certificates-java java-common net-tools openjdk-8-jre-headless puppet-agent puppetserver
0 upgraded, 6 newly installed, 0 to remove and 624 not upgraded.
Need to get 110 MB of archives.
After this operation, 290 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Configurer puppetserver

Utilisez la commande **puppet config** pour définir la valeur de la variable **dns_alt_names** :

```
root@master:~# /opt/puppetlabs/bin/puppet config set dns_alt_names 'master,master.i2tch.loc' --section main
root@master:~# cat /etc/puppetlabs/puppet/puppet.conf | grep dns_alt_names
dns_alt_names = master,master.i2tch.loc
```

Utilisez la commande **puppet config** pour définir la valeur de la variable **server** :

```
root@master:~# /opt/puppetlabs/bin/puppet config set server 'master.i2tch.loc'
```

Ajoutez la ligne **export PATH=/opt/puppetlabs/bin:$PATH** au fichier **~/.bashrc** :

```
root@master:~# vi .bashrc
root@master:~# tail .bashrc
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
#if [ -f /etc/bash_completion ] && ! shopt -oq posix; then
#    . /etc/bash_completion
#fi

export PATH=/opt/puppetlabs/bin:$PATH
```

Saisissez la commande suivante :

```
root@master:~# export PATH=/opt/puppetlabs/bin:$PATH
```

Activez et démarrez le serveur Puppet :

```
root@master:~# systemctl enable puppetserver
Synchronizing state of puppetserver.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable puppetserver

root@master:~# systemctl start puppetserver
oot@master:~# systemctl status puppetserver.service
● puppetserver.service - puppetserver Service
   Loaded: loaded (/lib/systemd/system/puppetserver.service; enabled; vendor pre
   Active: active (running) since Wed 2020-02-12 14:43:22 CET; 8min ago
  Process: 746 ExecStart=/opt/puppetlabs/server/apps/puppetserver/bin/puppetserv
```

```
 Main PID: 838 (java)
    Tasks: 46 (limit: 4915)
   CGroup: /system.slice/puppetserver.service
           └─838 /usr/bin/java -Xms2g -Xmx2g -Djruby.logger.class=com.puppetlabs

Feb 12 14:42:34 master.i2tch.loc systemd[1]: Starting puppetserver Service...
Feb 12 14:43:22 master.i2tch.loc systemd[1]: Started puppetserver Service.
```

Dernièrement, vérifiez que le fichier **/etc/hosts** contient des entrées pour **master.i2tch.loc**, **slave01.i2tch.loc** et **slave02.i2tch.loc** :

```
root@master:~# vi /etc/hosts
root@master:~# cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       master.i2tch.loc        master
10.0.2.59        master.i2tch.loc  master
10.0.2.68        slave01.i2tch.loc       slave01
10.0.2.69        slave02.i2tch.loc       slave02

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

LAB #2 - Installer et Configurer puppet-agent sur les Machines Virtuelles Esclaves

Installer puppet-agent

```
root@slave01:~# wget https://apt.puppetlabs.com/puppet-release-bionic.deb
--2020-02-12 14:27:25--  https://apt.puppetlabs.com/puppet-release-bionic.deb
```

```
Resolving apt.puppetlabs.com (apt.puppetlabs.com)... 143.204.226.21, 143.204.226.18, 143.204.226.112, ...
Connecting to apt.puppetlabs.com (apt.puppetlabs.com)|143.204.226.21|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11724 (11K) [application/x-debian-package]
Saving to: ‘puppet-release-bionic.deb’

puppet-release-bionic.deb       100%[=================================================>]  11.45K  --.-KB/s
in 0s

2020-02-12 14:27:26 (97.4 MB/s) - ‘puppet-release-bionic.deb’ saved [11724/11724]

root@slave01:~# dpkg -i puppet-release-bionic.deb
Selecting previously unselected package puppet-release.
(Reading database ... 128539 files and directories currently installed.)
Preparing to unpack puppet-release-bionic.deb ...
Unpacking puppet-release (1.0.0-7bionic) ...
Setting up puppet-release (1.0.0-7bionic) ...

root@slave01:~# apt update
Hit:1 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Get:3 http://apt.puppetlabs.com bionic InRelease [85.3 kB]
Hit:4 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:5 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelease
Get:6 http://apt.puppetlabs.com bionic/puppet i386 Packages [16.4 kB]
Get:7 http://apt.puppetlabs.com bionic/puppet amd64 Packages [39.9 kB]
Get:8 http://apt.puppetlabs.com bionic/puppet all Packages [16.4 kB]
Fetched 158 kB in 2s (81.3 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
619 packages can be upgraded. Run 'apt list --upgradable' to see them.

root@slave01:~# apt install puppet-agent
```

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  puppet-agent
0 upgraded, 1 newly installed, 0 to remove and 619 not upgraded.
Need to get 20.2 MB of archives.
After this operation, 116 MB of additional disk space will be used.
Get:1 http://apt.puppetlabs.com bionic/puppet amd64 puppet-agent amd64 6.12.0-1bionic [20.2 MB]
Fetched 20.2 MB in 3s (6,846 kB/s)
Selecting previously unselected package puppet-agent.
(Reading database ... 128544 files and directories currently installed.)
Preparing to unpack .../puppet-agent_6.12.0-1bionic_amd64.deb ...
Unpacking puppet-agent (6.12.0-1bionic) ...
Setting up puppet-agent (6.12.0-1bionic) ...
Created symlink /etc/systemd/system/multi-user.target.wants/puppet.service → /lib/systemd/system/puppet.service.
Created symlink /etc/systemd/system/multi-user.target.wants/pxp-agent.service → /lib/systemd/system/pxp-
agent.service.
Removed /etc/systemd/system/multi-user.target.wants/pxp-agent.service.
Processing triggers for libc-bin (2.27-3ubuntu1) ...
```

```
[root@slave02 ~]# rpm -Uvh https://yum.puppet.com/puppet/puppet-release-el-7.noarch.rpm
Retrieving https://yum.puppet.com/puppet/puppet-release-el-7.noarch.rpm
warning: /var/tmp/rpm-tmp.PYL8Aj: Header V4 RSA/SHA256 Signature, key ID ef8d349f: NOKEY
Preparing...                          ################################# [100%]
Updating / installing...
   1:puppet-release-1.0.0-3.el7       ################################# [100%]

[root@slave02 ~]# yum install puppet-agent
Loaded plugins: fastestmirror, langpacks
Determining fastest mirrors
 * base: mirrors.ircam.fr
 * extras: miroir.univ-lorraine.fr
 * updates: mirrors.standaloneinstaller.com
```

```
base                                                                                                     | 3.6 kB
00:00:00
extras                                                                                                   | 2.9 kB
00:00:00
puppet                                                                                                   | 2.5 kB
00:00:00
updates                                                                                                  | 2.9 kB
00:00:00
(1/5): base/7/x86_64/group_gz                                                                            | 165 kB
00:00:00
(2/5): extras/7/x86_64/primary_db                                                                        | 159 kB
00:00:00
(3/5): puppet/x86_64/primary_db                                                                          | 256 kB
00:00:02
(4/5): base/7/x86_64/primary_db                                                                          | 6.0 MB
00:00:02
(5/5): updates/7/x86_64/primary_db                                                                       | 6.7 MB
00:00:02
Resolving Dependencies
--> Running transaction check
---> Package puppet-agent.x86_64 0:6.12.0-1.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

==================================================================================================================
=============
 Package                          Arch                       Version                                  Repository
Size
==================================================================================================================
=============
Installing:
 puppet-agent                     x86_64                     6.12.0-1.el7                             puppet
23 M
```

```
Transaction Summary
================================================================================================================================
=============
Install  1 Package

Total download size: 23 M
Installed size: 23 M
Is this ok [y/d/N]: y
```

Configurer puppet-agent

Utilisez la commande **puppet config** sur chaque nœud pour définir la valeur de la variable **server** :

```
root@slave01:~# /opt/puppetlabs/bin/puppet config set server 'master.i2tch.loc' --section main
root@slave01:~# cat /etc/puppetlabs/puppet/puppet.conf | grep server
server = master.i2tch.loc
```

```
[root@slave02 ~]# /opt/puppetlabs/bin/puppet config set server 'master.i2tch.loc' --section main
[root@slave02 ~]# cat /etc/puppetlabs/puppet/puppet.conf | grep server
server = master.i2tch.loc
```

Utilisez la commande **puppet resource** afin d'activer et de démarrer le service de puppet agent :

```
root@slave01:~# /opt/puppetlabs/bin/puppet resource service puppet ensure=running enable=true
Notice: /Service[puppet]/ensure: ensure changed 'stopped' to 'running'
service { 'puppet':
  ensure => 'running',
  enable => 'true',
}
```

```
[root@slave02 ~]# /opt/puppetlabs/bin/puppet resource service puppet ensure=running enable=true
Notice: /Service[puppet]/ensure: ensure changed 'stopped' to 'running'
```

```
service { 'puppet':
  ensure => 'running',
  enable => 'true',
}
```

Retournez sur la machine virtuelle **PuppetMaster** et lister les certificats en attente de validation :

```
root@master:~# /opt/puppetlabs/bin/puppetserver ca list
Requested Certificates:
    slave01.i2tch.loc   (SHA256)
81:02:B3:C7:6F:BE:DB:48:93:9E:1A:A5:87:CA:AF:E5:DB:14:09:11:2D:43:60:39:1C:BE:6F:A1:CF:C0:BD:31
    slave02.i2tch.loc   (SHA256)
EF:BF:00:84:F1:F0:3B:C0:5F:A8:6F:49:98:E5:73:FA:39:B6:16:8E:8D:B3:0E:38:04:76:4D:2E:BF:BE:53:57
```

Validez les certificats en attente :

```
root@master:~# /opt/puppetlabs/bin/puppetserver ca sign --certname slave01.i2tch.loc,slave02.i2tch.loc
Successfully signed certificate request for slave01.i2tch.loc
Successfully signed certificate request for slave02.i2tch.loc
```

Si vous ne voyez pas de certificats ou seulement un certificat sur deux, arrêtez le service puppet sur le(s) noeud(s) concerné(s) :

```
root@slave01:~# systemctl stop puppet
root@slave02:~# systemctl stop puppet
```

Supprimez les certificats existants :

```
root@master:~# puppetserver ca clean --certname slave01.i2tch.loc
root@master:~# puppetserver ca clean --certname slave02.i2tch.loc
```

Lancez ensuite les commandes suivantes dans les deux esclaves :

```
root@slave01:~# rm -rf /etc/puppetlabs/puppet/ssl/
```

```
root@slave01:~# puppet agent --test
Info: Creating a new RSA SSL key for slave01.i2tch.loc
Info: csr_attributes file loading from /etc/puppetlabs/puppet/csr_attributes.yaml
Info: Creating a new SSL certificate request for slave01.i2tch.loc
Info: Certificate Request fingerprint (SHA256):
81:02:B3:C7:6F:BE:DB:48:93:9E:1A:A5:87:CA:AF:E5:DB:14:09:11:2D:43:60:39:1C:BE:6F:A1:CF:C0:BD:31
Info: Certificate for slave01.i2tch.loc has not been signed yet
Couldn't fetch certificate from CA server; you might still need to sign this agent's certificate
(slave01.i2tch.loc).
Exiting now because the waitforcert setting is set to 0.
```

```
[root@slave02 ~]# rm -rf /etc/puppetlabs/puppet/ssl/
[root@slave02 ~]# puppet agent --test
Info: Creating a new RSA SSL key for slave02.i2tch.loc
Info: csr_attributes file loading from /etc/puppetlabs/puppet/csr_attributes.yaml
Info: Creating a new SSL certificate request for slave02.i2tch.loc
Info: Certificate Request fingerprint (SHA256):
EF:BF:00:84:F1:F0:3B:C0:5F:A8:6F:49:98:E5:73:FA:39:B6:16:8E:8D:B3:0E:38:04:76:4D:2E:BF:BE:53:57
Info: Certificate for slave02.i2tch.loc has not been signed yet
Couldn't fetch certificate from CA server; you might still need to sign this agent's certificate
(slave02.i2tch.loc).
Exiting now because the waitforcert setting is set to 0.
```

Retournez ensuite à la machine virtuelle **master** et validez les certificats en attente :

```
root@master:~# /opt/puppetlabs/bin/puppetserver ca sign --certname slave01.i2tch.loc,slave02.i2tch.loc
Successfully signed certificate request for slave01.i2tch.loc
Successfully signed certificate request for slave02.i2tch.loc
```

Retournez sur les esclaves et exécutez la commande **puppet agent** :

```
root@slave01:~# /opt/puppetlabs/bin/puppet agent -t
Info: csr_attributes file loading from /etc/puppetlabs/puppet/csr_attributes.yaml
Info: Creating a new SSL certificate request for slave01.i2tch.loc
```

```
Info: Certificate Request fingerprint (SHA256):
81:02:B3:C7:6F:BE:DB:48:93:9E:1A:A5:87:CA:AF:E5:DB:14:09:11:2D:43:60:39:1C:BE:6F:A1:CF:C0:BD:31
Info: Downloaded certificate for slave01.i2tch.loc from https://master.i2tch.loc:8140/puppet-ca/v1
Info: Using configured environment 'production'
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Retrieving locales
Info: Caching catalog for slave01.i2tch.loc
Info: Applying configuration version '1581520502'
Notice: Applied catalog in 0.01 seconds
```

```
[root@slave02 ~]# /opt/puppetlabs/bin/puppet agent -t
Info: Using configured environment 'production'
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Retrieving locales
Info: Caching catalog for slave02.i2tch.loc
Info: Applying configuration version '1581520616'
Notice: Applied catalog in 0.03 seconds
```

LAB #3 - Création d'un Utilisateur

Placez-vous dans le répertoire **/etc/puppetlabs/code/environments/production/modules/** et créez le répertoire **accounts** :

```
root@master:~# cd /etc/puppetlabs/code/environments/production/modules/
root@master:/etc/puppetlabs/code/environments/production/modules# mkdir accounts
```

Placez-vous dans le répertoire **/etc/puppetlabs/code/environments/production/modules/accounts** et créez les répertoires **examples**, **files**, **manifests**, **templates** :

```
root@master:/etc/puppetlabs/code/environments/production/modules# cd accounts
```

```
SNAPSHOT1

root@master:/etc/puppetlabs/code/environments/production/modules/accounts# mkdir
{examples,files,manifests,templates}
```

Placez-vous dans le répertoire **/etc/puppetlabs/code/environments/production/modules/accounts/manifests** et créez le fichier init.pp :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts# cd manifests
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# vi init.pp
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# cat init.pp
class accounts {

  $rootgroup = $osfamily ? {
    'Debian'  => 'sudo',
    'RedHat'  => 'wheel',
    default   => warning('This distribution is not supported by the Accounts module'),
  }

  include accounts::groups

  user { 'toto':
    ensure      => present,
    home        => '/home/toto',
    shell       => '/bin/bash',
    managehome  => true,
    gid         => 'toto',
    groups      => "$rootgroup",
  }

}
```

Créez ensuite le fichier **/etc/puppetlabs/code/environments/production/modules/accounts/manifests/groups.pp** afin de créer le groupe **toto** :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# vi groups.pp
```

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# cat groups.pp
class accounts::groups {

  group { 'toto':
    ensure  => present,
  }

}
```

Créez le mot de passe au format SHA1 pour l'utilisateur **toto** :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# openssl passwd -1
Password: toto
Verifying - Password: toto
$1$1lyYx3kO$sb0z34V28E7b7kYQb3Wjz.
```

Mettez à jour le fichier **/etc/puppetlabs/code/environments/production/modules/accounts/manifests/init.pp** pour inclure le mot de passe de **toto** :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# vi init.pp
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# cat  init.pp
class accounts {

  $rootgroup = $osfamily ? {
    'Debian'  => 'sudo',
    'RedHat'  => 'wheel',
    default   => warning('This distribution is not supported by the Accounts module'),
  }

  include accounts::groups

  user { 'toto':
    ensure      => present,
    home        => '/home/toto',
```

```
    shell       => '/bin/bash',
    managehome  => true,
    gid         => 'toto',
    groups      => "$rootgroup",
    password    => '$1$1lyYx3kO$sb0z34V28E7b7kYQb3Wjz.',
  }

}
```

Validez la syntaxe du fichier :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# /opt/puppetlabs/bin/puppet
parser validate init.pp
```

Naviguez vers le répertoire **/etc/puppetlabs/code/environments/production/modules/accounts/examples** :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests# cd ../examples/
```

Créez le fichier **/etc/puppetlabs/code/environments/production/modules/accounts/examples/init.pp** :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/examples# vi init.pp
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/examples# cat init.pp
include accounts
```

Testez ce manifest :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/examples# /opt/puppetlabs/bin/puppet
apply --noop init.pp
Notice: Compiled catalog for master.i2tch.loc in environment production in 0.10 seconds
Notice: /Stage[main]/Accounts::Groups/Group[toto]/ensure: current_value 'absent', should be 'present' (noop)
Notice: Class[Accounts::Groups]: Would have triggered 'refresh' from 1 event
Notice: /Stage[main]/Accounts/User[toto]/ensure: current_value 'absent', should be 'present' (noop)
Notice: Class[Accounts]: Would have triggered 'refresh' from 1 event
Notice: Stage[main]: Would have triggered 'refresh' from 2 events
```

```
Notice: Applied catalog in 0.39 seconds
```

Appliquez maintenant le manifest :

```
root@master:/etc/puppetlabs/code/environments/production/modules/accounts/examples# /opt/puppetlabs/bin/puppet
apply init.pp
Notice: Compiled catalog for master.i2tch.loc in environment production in 0.04 seconds
Notice: /Stage[main]/Accounts::Groups/Group[toto]/ensure: created
Notice: /Stage[main]/Accounts/User[toto]/ensure: created
Notice: Applied catalog in 0.37 seconds
```

Déconnectez-vous et reconnectez-vous en tant que l'utilisateur **toto** :

```
trainee@traineeXX:~$ ssh -l toto 10.0.2.59
toto@localhost's password: toto
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-29-generic x86_64)

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


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

636 packages can be updated.
380 updates are security updates.


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.

toto@master:~$
```

LAB #4 - Configuration de ssh

Naviguez vers le répertoire **/etc/puppetlabs/code/environements/production/modules/accounts/files** :

```
toto@master:~$ cd /etc/puppetlabs/code/environments/production/modules/accounts/files
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/files$
```

Copiez le fichier **/etc/sshd_config** vers le répertoire **/etc/puppetlabs/code/environments/production/modules/accounts/files** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/files$ sudo mv /etc/ssh/sshd_config .
[sudo] password for toto: toto
```

Ajoutez la directive **PermitRootLogin no** au fichier **/etc/puppetlabs/code/environments/production/modules/accounts/files/sshd_config** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/files$ sudo vi sshd_config
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/files$ cat sshd_config
...
# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
```

```
#MaxSessions 10

#PubkeyAuthentication yes
...
```

Retournez maintenant vers le répertoire **/etc/puppetlabs/code/environments/production/modules/accounts/manifests** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/files$ cd ../manifests/
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$
```

Créez un manifest appelé **ssh.pp** qui utilise un attribut **file** pour remplacer le fichier sshd_config par défaut avec celui de Puppet, démarrer le service ssh et re-démarrer le service ssh en cas de besoin :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$ sudo vi ssh.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$ cat ssh.pp
class accounts::ssh {

  $sshname = $osfamily ? {
    'Debian'  => 'ssh',
    'RedHat'  => 'sshd',
    default   => warning('This distribution is not supported by the Accounts module'),
  }

  file { '/etc/ssh/sshd_config':
    ensure  => present,
    source  => 'puppet:///modules/accounts/sshd_config',
    notify  => Service["$sshname"],
  }

  service { "$sshname":
    hasrestart  => true,
  }
```

```
}
```

Modifiez maintenant le fichier **/etc/puppetlabs/code/environments/production/modules/accounts/manifests/init.pp** afin d'inclure la classe **accounts::ssh** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$ sudo vi init.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$ cat init.pp
class accounts {

  $rootgroup = $osfamily ? {
    'Debian'  => 'sudo',
    'RedHat'  => 'wheel',
    default   => warning('This distribution is not supported by the Accounts module'),
  }

  include accounts::groups
  include accounts::ssh

  user { 'toto':
    ensure      => present,
    home        => '/home/toto',
    shell       => '/bin/bash',
    managehome  => true,
    gid         => 'toto',
    groups      => "$rootgroup",
    password    => '$1$1lyYx3kO$sb0z34V28E7b7kYQb3Wjz.',
  }

}
```

Testez ensuite la syntaxe des fichiers **ssh.pp** et **init.pp** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$ sudo
/opt/puppetlabs/bin/puppet parser validate ssh.pp
```

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$ sudo
/opt/puppetlabs/bin/puppet parser validate init.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$
```

Naviguez vers le répertoire **/etc/puppetlabs/code/environments/production/modules/accounts** et installez l'utilitaire **tree** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/manifests$ cd ..
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts$ sudo apt install tree
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  tree
0 upgraded, 1 newly installed, 0 to remove and 574 not upgraded.
Need to get 40.7 kB of archives.
After this operation, 105 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu bionic/universe amd64 tree amd64 1.7.0-5 [40.7 kB]
Fetched 40.7 kB in 1s (72.3 kB/s)
Selecting previously unselected package tree.
(Reading database ... 147924 files and directories currently installed.)
Preparing to unpack .../tree_1.7.0-5_amd64.deb ...
Unpacking tree (1.7.0-5) ...
Setting up tree (1.7.0-5) ...
Processing triggers for man-db (2.8.3-2) ...
```

Vérifiez l'arborescence du module **accounts** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts$ tree
.
├── examples
│   └── init.pp
├── files
│   └── sshd_config
```

```
├── manifests
│   ├── groups.pp
│   ├── init.pp
│   └── ssh.pp
└── templates

4 directories, 5 files
```

Naviguez vers le répertoire **/etc/puppetlabs/code/environments/production/modules/accounts/examples** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts$ cd examples/
```

Testez ensuite l'application du manifest **init.pp** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/examples$ sudo
/opt/puppetlabs/bin/puppet apply --noop init.pp
Notice: Compiled catalog for master.i2tch.loc in environment production in 0.24 seconds
Notice: /Stage[main]/Accounts::Ssh/File[/etc/ssh/sshd_config]/content: current_value
'{md5}739d6887c8f3dd71a9168c614c07175c', should be '{md5}0876b6b0db0707db221a5c736d8a896a' (noop)
Notice: /Stage[main]/Accounts::Ssh/Service[ssh]: Would have triggered 'refresh' from 1 event
Notice: Class[Accounts::Ssh]: Would have triggered 'refresh' from 2 events
Notice: Stage[main]: Would have triggered 'refresh' from 1 event
Notice: Applied catalog in 0.03 seconds
```

Et finalement appliquez le manifest :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/examples$ sudo
/opt/puppetlabs/bin/puppet apply init.pp
Notice: Compiled catalog for master.i2tch.loc in environment production in 0.25 seconds
Notice: /Stage[main]/Accounts::Ssh/File[/etc/ssh/sshd_config]/content: content changed
'{md5}0876b6b0db0707db221a5c736d8a896a' to '{md5}1ddc2551e3c3766390706609083581b2'
Notice: /Stage[main]/Accounts::Ssh/Service[ssh]: Triggered 'refresh' from 1 event
Notice: Applied catalog in 0.09 seconds
```

LAB #5 - Configuration d'IP Tables

Installez le paquet **iptables-persistent** ou **iptables-services** dans chaque machine virtuelle en fonction de la distribution :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/accounts/examples$ cd ~
toto@master:~$ sudo apt install iptables-persistent
```

```
root@slave01:~# apt install iptables-persistent
```

```
[root@slave02 ~]# systemctl stop firewalld && systemctl disable firewalld
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
Removed symlink /etc/systemd/system/basic.target.wants/firewalld.service.
[root@slave02 ~]# yum install iptables-services
```

Installez le module Puppet **puppetlabs-firewall** dans la machine virtuelle **puppetmaster** :

```
toto@master:~$ sudo /opt/puppetlabs/bin/puppet module install puppetlabs-firewall
Notice: Preparing to install into /etc/puppetlabs/code/environments/production/modules ...
Notice: Downloading from https://forgeapi.puppet.com ...
Notice: Installing -- do not interrupt ...
/etc/puppetlabs/code/environments/production/modules
└─┬ puppetlabs-firewall (v2.2.0)
  └── puppetlabs-stdlib (v6.2.0)
```

Naviguez maintenant vers le répertoire **/etc/puppetlabs/code/environments/production/modules/firewall/manifests/** :

```
toto@master:~$ cd /etc/puppetlabs/code/environments/production/modules/firewall/manifests/
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$
```

Créez le fichier **pre.pp** qui contiendra toutes les règles de base du pare-feu pour le réseau :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$ sudo vi pre.pp
```

```
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$ cat pre.pp
class firewall::pre {

  Firewall {
    require => undef,
  }

   # Accept all loopback traffic
  firewall { '000 lo traffic':
    proto       => 'all',
    iniface     => 'lo',
    action      => 'accept',
  }->

   #Drop non-loopback traffic
  firewall { '001 reject non-lo':
    proto       => 'all',
    iniface     => '! lo',
    destination => '127.0.0.0/8',
    action      => 'reject',
  }->

   #Accept established inbound connections
  firewall { '002 accept established':
    proto       => 'all',
    state       => ['RELATED', 'ESTABLISHED'],
    action      => 'accept',
  }->

   #Allow all outbound traffic
  firewall { '003 allow outbound':
    chain       => 'OUTPUT',
    action      => 'accept',
  }->
```

```
   #Allow ICMP/ping
  firewall { '004 allow icmp':
    proto       => 'icmp',
    action      => 'accept',
  }

   #Allow SSH connections
  firewall { '005 Allow SSH':
    dport    => '22',
    proto   => 'tcp',
    action  => 'accept',
  }->

   #Allow HTTP/HTTPS connections
  firewall { '006 HTTP/HTTPS connections':
    dport    => ['80', '443'],
    proto   => 'tcp',
    action  => 'accept',
  }

}
```

Créez le fichier **post.pp** pour interdire tout trafic qui n'est pas spécifiquement autorisé par le fichier **pre.pp** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$ sudo vi post.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$ cat post.pp
class firewall::post {

  firewall { '999 drop all':
    proto  => 'all',
    action => 'drop',
    before => undef,
  }
```

```
}
```

Vérifiez la syntaxe des deux fichiers :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$ sudo
/opt/puppetlabs/bin/puppet parser validate pre.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$ sudo
/opt/puppetlabs/bin/puppet parser validate post.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$
```

Vérifiez l'arborescence du répertoire **manifests** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$ tree
.
├── init.pp
├── linux
│   ├── archlinux.pp
│   ├── debian.pp
│   ├── gentoo.pp
│   └── redhat.pp
├── linux.pp
├── params.pp
├── post.pp
└── pre.pp

1 directory, 9 files
```

Naviguez vers le répertoire principal des manifests :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/firewall/manifests$ cd
/etc/puppetlabs/code/environments/production/manifests
toto@master:/etc/puppetlabs/code/environments/production/manifests$
```

Créez le fichier **site.pp** qui pilote l'ensemble des autres classes définies dans les fichiers précédents :

```
toto@master:/etc/puppetlabs/code/environments/production/manifests$ sudo vi site.pp
toto@master:/etc/puppetlabs/code/environments/production/manifests$ cat site.pp
node default {

}

node 'master.i2tch.loc' {

  include accounts

  resources { 'firewall':
    purge => true,
  }

  Firewall {
    before        => Class['firewall::post'],
    require       => Class['firewall::pre'],
  }

  class { ['firewall::pre', 'firewall::post']: }

  firewall { '200 Allow Puppet Master':
    dport         => '8140',
    proto         => 'tcp',
    action        => 'accept',
  }

}
```

Vérifiez la syntaxe du fichier site.pp :

```
toto@master:/etc/puppetlabs/code/environments/production/manifests$ sudo /opt/puppetlabs/bin/puppet parser
```

```
validate site.pp
toto@master:/etc/puppetlabs/code/environments/production/manifests$
```

Testez l'exécution de ce manifest :

```
toto@master:/etc/puppetlabs/code/environments/production/manifests$ sudo /opt/puppetlabs/bin/puppet apply --noop
site.pp
Notice: Compiled catalog for master.i2tch.loc in environment production in 0.14 seconds
Notice: /Stage[main]/Firewall::Pre/Firewall[000 lo traffic]/ensure: current_value 'absent', should be 'present'
(noop)
Notice: /Stage[main]/Firewall::Pre/Firewall[001 reject non-lo]/ensure: current_value 'absent', should be
'present' (noop)
Notice: /Stage[main]/Firewall::Pre/Firewall[002 accept established]/ensure: current_value 'absent', should be
'present' (noop)
Notice: /Stage[main]/Firewall::Pre/Firewall[003 allow outbound]/ensure: current_value 'absent', should be
'present' (noop)
Notice: /Stage[main]/Firewall::Pre/Firewall[004 allow icmp]/ensure: current_value 'absent', should be 'present'
(noop)
Notice: /Stage[main]/Firewall::Pre/Firewall[005 Allow SSH]/ensure: current_value 'absent', should be 'present'
(noop)
Notice: /Stage[main]/Firewall::Pre/Firewall[006 HTTP/HTTPS connections]/ensure: current_value 'absent', should be
'present' (noop)
Notice: Class[Firewall::Pre]: Would have triggered 'refresh' from 7 events
Notice: /Stage[main]/Main/Node[master.i2tch.loc]/Firewall[200 Allow Puppet Master]/ensure: current_value
'absent', should be 'present' (noop)
Notice: Node[master.i2tch.loc]: Would have triggered 'refresh' from 1 event
Notice: Class[Main]: Would have triggered 'refresh' from 1 event
Notice: /Stage[main]/Firewall::Post/Firewall[999 drop all]/ensure: current_value 'absent', should be 'present'
(noop)
Notice: Class[Firewall::Post]: Would have triggered 'refresh' from 1 event
Notice: Stage[main]: Would have triggered 'refresh' from 3 events
Notice: Applied catalog in 0.17 seconds
```

Appliquez maintenant le manifest **site.pp** :

```
toto@master:/etc/puppetlabs/code/environments/production/manifests$ sudo /opt/puppetlabs/bin/puppet apply site.pp
Notice: Compiled catalog for master.i2tch.loc in environment production in 0.12 seconds
Notice: /Stage[main]/Firewall::Pre/Firewall[000 lo traffic]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[001 reject non-lo]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[002 accept established]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[003 allow outbound]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[004 allow icmp]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[005 Allow SSH]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[006 HTTP/HTTPS connections]/ensure: created
Notice: /Stage[main]/Main/Node[master.i2tch.loc]/Firewall[200 Allow Puppet Master]/ensure: created
Notice: /Stage[main]/Firewall::Post/Firewall[999 drop all]/ensure: created
Notice: Applied catalog in 0.98 seconds
```

Visualisez les règles du pare-feu :

```
toto@master:/etc/puppetlabs/code/environments/production/manifests$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere             /* 000 lo traffic */
REJECT     all  --  anywhere             localhost/8          /* 001 reject non-lo */ reject-with icmp-port-
unreachable
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED /* 002 accept established
*/
ACCEPT     icmp --  anywhere             anywhere             /* 004 allow icmp */
ACCEPT     tcp  --  anywhere             anywhere             multiport dports ssh /* 005 Allow SSH */
ACCEPT     tcp  --  anywhere             anywhere             multiport dports http,https /* 006 HTTP/HTTPS
connections */
ACCEPT     tcp  --  anywhere             anywhere             multiport dports puppet /* 200 Allow Puppet Master
*/
DROP       all  --  anywhere             anywhere             /* 999 drop all */

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
```

```
Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             /* 003 allow outbound */
```

Modifiez le manifest **site.pp** pour inclure les sections pour déclarer les classes, les modules et les ressources à appliquer à **slave01.i2tch.loc** et à **slave02.i2tch.loc** :

```
toto@master:/etc/puppetlabs/code/environments/production/manifests$ sudo vi site.pp
toto@master:/etc/puppetlabs/code/environments/production/manifests$ cat site.pp
node default {

}

node 'master.i2tch.loc' {

  include accounts

  resources { 'firewall':
    purge => true,
  }

  Firewall {
    before        => Class['firewall::post'],
    require       => Class['firewall::pre'],
  }

  class { ['firewall::pre', 'firewall::post']: }

  firewall { '200 Allow Puppet Master':
    dport         => '8140',
    proto         => 'tcp',
    action        => 'accept',
  }
```

```
}

node 'slave01.i2tch.loc' {

  include accounts

  resources { 'firewall':
    purge => true,
  }

  Firewall {
    before        => Class['firewall::post'],
    require       => Class['firewall::pre'],
  }

  class { ['firewall::pre', 'firewall::post']: }

}

node 'slave02.i2tch.loc' {

  include accounts

  resources { 'firewall':
    purge => true,
  }

  Firewall {
    before        => Class['firewall::post'],
    require       => Class['firewall::pre'],
  }

  class { ['firewall::pre', 'firewall::post']: }
```

```
}
```

Exécuter l'Agent Puppet sur slave01.i2tch.loc

Connectez-vous à la machine virtuelle **slave01.i2tch.loc** en ssh et devenez **root** :

```
trainee@slave01:~$ su -
Password: fenestros
```

Exécutez l'agent Puppet :

```
root@slave01:~# /opt/puppetlabs/bin/puppet agent -t
...
Notice: /Stage[main]/Accounts::Ssh/Service[ssh]: Triggered 'refresh' from 1 event
Notice: /Stage[main]/Accounts/User[toto]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[000 lo traffic]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[001 reject non-lo]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[002 accept established]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[003 allow outbound]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[004 allow icmp]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[005 Allow SSH]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[006 HTTP/HTTPS connections]/ensure: created
Notice: /Stage[main]/Firewall::Post/Firewall[999 drop all]/ensure: created
Notice: Applied catalog in 1.78 seconds
```

Déconnectez-vous :

```
root@slave01:~# exit
logout
trainee@slave01:~$ exit
logout
Connection to localhost closed.
```

Re-connectez-vous à la machine virtuelle **slave01.i2tch.loc** en ssh et en tant que l'utilisateur **toto** :

```
...
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-29-generic x86_64)

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


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

330 packages can be updated.
8 updates are security updates.

*** System restart required ***

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.

toto@slave01:~$
```

Vérifiez que les règles du pare-feu ont été appliquées :

```
toto@slave01:~$ sudo iptables -L
```

```
[sudo] password for toto: toto
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere             /* 000 lo traffic */
REJECT     all  --  anywhere             localhost/8          /* 001 reject non-lo */ reject-with icmp-port-
unreachable
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED /* 002 accept established
*/
ACCEPT     icmp --  anywhere             anywhere             /* 004 allow icmp */
ACCEPT     tcp  --  anywhere             anywhere             multiport dports ssh /* 005 Allow SSH */
ACCEPT     tcp  --  anywhere             anywhere             multiport dports http,https /* 006 HTTP/HTTPS
connections */
DROP       all  --  anywhere             anywhere             /* 999 drop all */

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             /* 003 allow outbound */
```

Exécuter l'Agent Puppet sur slave02.i2tch.loc

Connectez-vous à la machine virtuelle **slave02.i2tch.loc** en ssh et devenez **root** :

```
[trainee@slave02 ~]$ su -
Mot de passe : fenestros
Dernière connexion : mercredi 13 mars 2019 à 22:01:35 CET sur pts/0
[root@slave02 ~]#
```

Exécutez l'agent Puppet :

```
[root@slave02 ~]# /opt/puppetlabs/bin/puppet agent -t
```

```
...
Notice: /Stage[main]/Firewall::Pre/Firewall[000 lo traffic]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[001 reject non-lo]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[002 accept established]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[003 allow outbound]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[004 allow icmp]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[005 Allow SSH]/ensure: created
Notice: /Stage[main]/Firewall::Pre/Firewall[006 HTTP/HTTPS connections]/ensure: created
Notice: /Stage[main]/Firewall::Post/Firewall[999 drop all]/ensure: created
Notice: Applied catalog in 0.97 seconds
```

Déconnectez-vous :

```
[root@slave02 ~]# exit
logout
[trainee@slave02 ~]$ exit
déconnexion
Connection to localhost closed.
```

Re-connectez-vous à la machine virtuelle **slave02.i2tch.loc** en ssh et en tant que l'utilisateur **toto**. Vérifiez que les règles du pare-feu ont été appliquées :

```
-bash-4.2$ sudo iptables -L

Nous espérons que vous avez reçu de votre administrateur système local les consignes traditionnelles.
Généralement, elles se concentrent sur ces trois éléments :

    #1) Respectez la vie privée des autres.
    #2) Réfléchissez avant d'utiliser le clavier.
    #3) De grands pouvoirs confèrent de grandes responsabilités.

[sudo] Mot de passe de toto : toto
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
```

```
ACCEPT     all  --  anywhere             anywhere             /* 000 lo traffic */
REJECT     all  --  anywhere             loopback/8           /* 001 reject non-lo */ reject-with icmp-port-
unreachable
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED /* 002 accept established
*/
ACCEPT     icmp --  anywhere             anywhere             /* 004 allow icmp */
ACCEPT     tcp  --  anywhere             anywhere             multiport dports ssh /* 005 Allow SSH */
ACCEPT     tcp  --  anywhere             anywhere             multiport dports http,https /* 006 HTTP/HTTPS
connections */
DROP       all  --  anywhere             anywhere             /* 999 drop all */

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             /* 003 allow outbound */
```

LAB #6 - Déployer Apache avec Puppet en mode Agent/Maître

Le but ici est de créer un rôle contenant les fichiers suivants :

```
.
├── examples
│   └── init.pp
├── files
│   ├── apache2.conf
│   └── httpd.conf
├── manifests
│   ├── init.pp
│   ├── params.pp
│   └── vhosts.pp
```

```
└── templates
    ├── vhosts-deb.conf.erb
    └── vhosts-rh.conf.erb
```

Création du Rôle

Naviguez vers le répertoire **/etc/puppetlabs/code/environments/production/modules/** et créez le sous-répertoire **apache** :

```
toto@master:/etc/puppetlabs/code/environments/production/manifests$ cd ../modules/
toto@master:/etc/puppetlabs/code/environments/production/modules$
toto@master:/etc/puppetlabs/code/environments/production/modules$ sudo mkdir apache
```

Placez-vous dans le répertoire **/etc/puppetlabs/code/environments/production/modules/apache** et créez les répertoires **manifests**, **templates**, **files** et **examples** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules$ cd apache
toto@master:/etc/puppetlabs/code/environments/production/modules/apache$ sudo mkdir
{manifests,templates,files,examples}
```

Création des Manifests

Placez-vous dans le répertoire **/etc/puppetlabs/code/environments/production/modules/apache/manifests** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache$ cd manifests/
```

Créez le manifest **init.pp** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ sudo vi init.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ cat init.pp
class apache (
  $apachename   = $::apache::params::apachename,
```

```
  $conffile   = $::apache::params::conffile,
  $confsource = $::apache::params::confsource,
) inherits ::apache::params {

  package { 'apache':
    name    => $apachename,
    ensure  => present,
  }

  file { 'configuration-file':
    path    => $conffile,
    ensure  => file,
    source  => $confsource,
    notify  => Service['apache-service'],
  }

  service { 'apache-service':
    name      => $apachename,
    hasrestart    => true,
  }

}
```

Dans ce fichier on note :

- le paquet à installer est référencé par une variable **$apachename**,
- la variable $apachename est fixée par la classe **apache::params** et injecté dans le fichier grâce à **$apachename = $::apache::params::apachename**,
- la ressource **file** utilise deux variables, **$conffile** et **$confsource**, également fixées par la classe apache::params.

Créez ensuite le manifest **params.pp** pour définir la classe **apache::params** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ sudo vi params.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ cat params.pp
```

```
class apache::params {

  if $::osfamily == 'RedHat' {
    $apachename   = 'httpd'
    $conffile     = '/etc/httpd/conf/httpd.conf'
    $confsource   = 'puppet:///modules/apache/httpd.conf'
  }
  elsif $::osfamily == 'Debian' {
    $apachename   = 'apache2'
    $conffile     = '/etc/apache2/apache2.conf'
    $confsource   = 'puppet:///modules/apache/apache2.conf'
  }
  else {
    fail ( 'this is not a supported distro.')
  }

}
```

Les hôtes virtuels d'Apache sont gérés différemment selon que **$::osfamily** soit RedHat ou Debian. Créez donc le manifest **vhosts.pp** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ sudo vi vhosts.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ cat vhosts.pp
class apache::vhosts {

  if $::osfamily == 'RedHat' {
    file { '/etc/httpd/conf.d/vhost.conf':
      ensure    => file,
      content   => template('apache/vhosts-rh.conf.erb'),
    }
    file { [ "/var/www/$servername",
             "/var/www/$servername/public_html",
             "/var/www/$servername/logs", ]:
      ensure    => directory,
    }
```

```
  } elsif $::osfamily == 'Debian' {
    file { "/etc/apache2/sites-available/$servername.conf":
      ensure  => file,
      content  => template('apache/vhosts-deb.conf.erb'),
    }
    file { [ "/var/www/$servername",
             "/var/www/$servername/public_html",
             "/var/www/$servername/logs", ]:
      ensure    => directory,
    }
  } else {
    fail ( 'This is not a supported distro.')
  }

}
```

Création des Fichiers de Configuration

Le fichier ci-dessus fait référence à deux **$confsource** différents selon que **$osfamily** soit RedHat ou Debian. Ces fichiers doivent être créés et sont des fichiers standards de configuration d'Apache.

Naviguez vers le répertoire **cd /etc/puppetlabs/code/environments/production/modules/apache/files** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ cd ../files/
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/files$
```

Créez le fichier **httpd.conf** suivant :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/files$ sudo vi httpd.conf
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/files$ cat httpd.conf
# This file is managed by Puppet
#
# This is the main Apache HTTP server configuration file.  It contains the
```

```
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/content/2.4/> for detailed information.
# In particular, see
# <URL:http://httpd.apache.org/content/2.4/mod/directives.html>
# for a discussion of each configuration directive.
#
# Do NOT simply read the instructions in here without understanding
# what they do.  They're here only as hints or reminders.  If you are unsure
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path.  If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so 'log/access_log'
# with ServerRoot set to '/www' will be interpreted by the
# server as '/www/log/access_log', where as '/log/access_log' will be
# interpreted as '/log/access_log'.

#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# Do not add a slash at the end of the directory path.  If you point
# ServerRoot at a non-local disk, be sure to specify a local disk on the
# Mutex directive, if file-based mutexes are used.  If you wish to share the
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/etc/httpd"

#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
```

```
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 80

#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
Include conf.modules.d/*.conf

#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
# It is usually good practice to create a dedicated user and group for
# running httpd, as with most system services.
#
User apache
Group apache

# 'Main' server configuration
#
```

```
# The directives in this section set up the values used by the 'main'
# server, which responds to any requests that aren't handled by a
# <VirtualHost> definition.  These values also provide defaults for
# any <VirtualHost> containers you may define later in the file.
#
# All of these directives may appear inside <VirtualHost> containers,
# in which case these default settings will be overridden for the
# virtual host being defined.
#

#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed.  This address appears on some server-generated pages, such
# as error documents.  e.g. admin@your-domain.com
#
ServerAdmin root@localhost

#
# ServerName gives the name and port that the server uses to identify itself.
# This can often be determined automatically, but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If your host doesn't have a registered DNS name, enter its IP address here.
#
#ServerName www.example.com:80

#
# Deny access to the entirety of your server's filesystem. You must
# explicitly permit access to web content directories in other
# <Directory> blocks below.
#
<Directory />
    AllowOverride none
    Require all denied
```

```
</Directory>

#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#

#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/var/www/html"

#
# Relax access to content within /var/www.
#
<Directory "/var/www">
    AllowOverride None
    # Allow open access:
    Require all granted
</Directory>

# Further relax access to the default document root:
<Directory "/var/www/html">
    #
    # Possible values for the Options directive are "None", "All",
    # or any combination of:
    #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
    #
    # Note that "MultiViews" must be named *explicitly* --- "Options All"
    # doesn't give it to you.
```

```
    #
    # The Options directive is both complicated and important.  Please see
    # http://httpd.apache.org/content/2.4/mod/core.html#options
    # for more information.
    #
    Options Indexes FollowSymLinks

    #
    # AllowOverride controls what directives may be placed in .htaccess files.
    # It can be "All", "None", or any combination of the keywords:
    #   Options FileInfo AuthConfig Limit
    #
    AllowOverride None

    #
    # Controls who can get stuff from this server.
    #
    Require all granted
</Directory>

#
# DirectoryIndex: sets the file that Apache will serve if a directory
# is requested.
#
<IfModule dir_module>
    DirectoryIndex index.html
</IfModule>

#
# The following lines prevent .htaccess and .htpasswd files from being
# viewed by Web clients.
#
<Files ".ht*">
    Require all denied
```

```
</Files>

#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog "logs/error_log"

#
# LogLevel: Control the number of messages logged to the error_log.
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn

<IfModule log_config_module>
    #
    # The following directives define some format nicknames for use with
    # a CustomLog directive (see below).
    #
    LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
    LogFormat "%h %l %u %t \"%r\" %>s %b" common

    <IfModule logio_module>
      # You need to enable mod_logio.c to use %I and %O
      LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O" combinedio
    </IfModule>

    #
    # The location and format of the access logfile (Common Logfile Format).
    # If you do not define any access logfiles within a <VirtualHost>
```

```
    # container, they will be logged here.  Contrariwise, if you *do*
    # define per-<VirtualHost> access logfiles, transactions will be
    # logged therein and *not* in this file.
    #
    #CustomLog "logs/access_log" common

    #
    # If you prefer a logfile with access, agent, and referer information
    # (Combined Logfile Format) you can use the following directive.
    #
    CustomLog "logs/access_log" combined
</IfModule>

<IfModule alias_module>
    #
    # Redirect: Allows you to tell clients about documents that used to
    # exist in your server's namespace, but do not anymore. The client
    # will make a new request for the document at its new location.
    # Example:
    # Redirect permanent /foo http://www.example.com/bar

    #
    # Alias: Maps web paths into filesystem paths and is used to
    # access content that does not live under the DocumentRoot.
    # Example:
    # Alias /webpath /full/filesystem/path
    #
    # If you include a trailing / on /webpath then the server will
    # require it to be present in the URL.  You will also likely
    # need to provide a <Directory> section to allow access to
    # the filesystem path.

    #
    # ScriptAlias: This controls which directories contain server scripts.
```

```
    # ScriptAliases are essentially the same as Aliases, except that
    # documents in the target directory are treated as applications and
    # run by the server when requested rather than as documents sent to the
    # client.  The same rules about trailing "/" apply to ScriptAlias
    # directives as to Alias.
    #
    ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"

</IfModule>

#
# "/var/www/cgi-bin" should be changed to whatever your ScriptAliased
# CGI directory exists, if you have that configured.
#
<Directory "/var/www/cgi-bin">
    AllowOverride None
    Options None
    Require all granted
</Directory>

<IfModule mime_module>
    #
    # TypesConfig points to the file containing the list of mappings from
    # filename extension to MIME-type.
    #
    TypesConfig /etc/mime.types

    #
    # AddType allows you to add to or override the MIME configuration
    # file specified in TypesConfig for specific file types.
    #
    #AddType application/x-gzip .tgz
    #
    # AddEncoding allows you to have certain browsers uncompress
```

```
    # information on the fly. Note: Not all browsers support this.
    #
    #AddEncoding x-compress .Z
    #AddEncoding x-gzip .gz .tgz
    #
    # If the AddEncoding directives above are commented-out, then you
    # probably should define those extensions to indicate media types:
    #
    AddType application/x-compress .Z
    AddType application/x-gzip .gz .tgz

    #
    # AddHandler allows you to map certain file extensions to "handlers":
    # actions unrelated to filetype. These can be either built into the server
    # or added with the Action directive (see below)
    #
    # To use CGI scripts outside of ScriptAliased directories:
    # (You will also need to add "ExecCGI" to the "Options" directive.)
    #
    #AddHandler cgi-script .cgi

    # For type maps (negotiated resources):
    #AddHandler type-map var

    #
    # Filters allow you to process content before it is sent to the client.
    #
    # To parse .shtml files for server-side includes (SSI):
    # (You will also need to add "Includes" to the "Options" directive.)
    #
    AddType text/html .shtml
    AddOutputFilter INCLUDES .shtml
</IfModule>
```

```
#
# Specify a default charset for all content served; this enables
# interpretation of all content as UTF-8 by default.  To use the
# default browser choice (ISO-8859-1), or to allow the META tags
# in HTML content to override this choice, comment out this
# directive:
#
AddDefaultCharset UTF-8

<IfModule mime_magic_module>
    #
    # The mod_mime_magic module allows the server to use various hints from the
    # contents of the file itself to determine its type.  The MIMEMagicFile
    # directive tells the module where the hint definitions are located.
    #
    MIMEMagicFile conf/magic
</IfModule>

#
# Customizable error responses come in three flavors:
# 1) plain text 2) local redirects 3) external redirects
#
# Some examples:
#ErrorDocument 500 "The server made a boo boo."
#ErrorDocument 404 /missing.html
#ErrorDocument 404 "/cgi-bin/missing_handler.pl"
#ErrorDocument 402 http://www.example.com/subscription_info.html
#

#
# EnableMMAP and EnableSendfile: On systems that support it,
# memory-mapping or the sendfile syscall may be used to deliver
# files.  This usually improves server performance, but must
# be turned off when serving from networked-mounted
```

```
# filesystems or if support for these functions is otherwise
# broken on your system.
# Defaults if commented: EnableMMAP On, EnableSendfile Off
#
#EnableMMAP off
EnableSendfile on

# Supplemental configuration
#
# Load config files in the "/etc/httpd/conf.d" directory, if any.
IncludeOptional conf.d/*.conf
```

Créez aussi le fichier **apache2.conf** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/files$ sudo vi apache2.conf
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/files$ cat apache2.conf
# This file is managed by Puppet
# This is the main Apache server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See http://httpd.apache.org/content/2.4/ for detailed information about
# the directives and /usr/share/doc/apache2/README.Debian about Debian specific
# hints.
#
#
# Summary of how the Apache 2 configuration works in Debian:
# The Apache 2 web server configuration in Debian is quite different to
# upstream's suggested way to configure the web server. This is because Debian's
# default Apache2 installation attempts to make adding and removing modules,
# virtual hosts, and extra configuration directives as flexible as possible, in
# order to make automating the changes and administering the server as easy as
# possible.

# It is split into several files forming the configuration hierarchy outlined
# below, all located in the /etc/apache2/ directory:
```

```
#
#   /etc/apache2/
#   |-- apache2.conf
#   |    `--  ports.conf
#   |-- mods-enabled
#   |    |-- *.load
#   |    `-- *.conf
#   |-- conf-enabled
#   |    `-- *.conf
#   `-- sites-enabled
#        `-- *.conf
#
#
# * apache2.conf is the main configuration file (this file). It puts the pieces
#   together by including all remaining configuration files when starting up the
#   web server.
#
# * ports.conf is always included from the main configuration file. It is
#   supposed to determine listening ports for incoming connections which can be
#   customized anytime.
#
# * Configuration files in the mods-enabled/, conf-enabled/ and sites-enabled/
#   directories contain particular configuration snippets which manage modules,
#   global configuration fragments, or virtual host configurations,
#   respectively.
#
#   They are activated by symlinking available configuration files from their
#   respective *-available/ counterparts. These should be managed by using our
#   helpers a2enmod/a2dismod, a2ensite/a2dissite and a2enconf/a2disconf. See
#   their respective man pages for detailed information.
#
# * The binary is called apache2. Due to the use of environment variables, in
#   the default configuration, apache2 needs to be started/stopped with
#   /etc/init.d/apache2 or apache2ctl. Calling /usr/bin/apache2 directly will not
```

```
#   work with the default configuration.


# Global configuration
#

#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# NOTE!  If you intend to place this on an NFS (or otherwise network)
# mounted filesystem then please read the Mutex documentation (available
# at <URL:http://httpd.apache.org/content/2.4/mod/core.html#mutex>);
# you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory path.
#
#ServerRoot "/etc/apache2"

#
# The accept serialization lock file MUST BE STORED ON A LOCAL DISK.
#
Mutex file:${APACHE_LOCK_DIR} default

#
# PidFile: The file in which the server should record its process
# identification number when it starts.
# This needs to be set in /etc/apache2/envvars
#
PidFile ${APACHE_PID_FILE}

#
# Timeout: The number of seconds before receives and sends time out.
#
```

```
Timeout 300

#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On

#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 100

#
# KeepAliveTimeout: Number of seconds to wait for the next request from the
# same client on the same connection.
#
KeepAliveTimeout 5


# These need to be set in /etc/apache2/envvars
User ${APACHE_RUN_USER}
Group ${APACHE_RUN_GROUP}

#
# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
```

```
HostnameLookups Off

# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog ${APACHE_LOG_DIR}/error.log

#
# LogLevel: Control the severity of messages logged to the error_log.
# Available values: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the log level for particular modules, e.g.
# "LogLevel info ssl:warn"
#
LogLevel warn

# Include module configuration:
IncludeOptional mods-enabled/*.load
IncludeOptional mods-enabled/*.conf

# Include list of ports to listen on
Include ports.conf



# Sets the default security model of the Apache2 HTTPD server. It does
# not allow access to the root filesystem outside of /usr/share and /var/www.
# The former is used by web applications packaged in Debian,
# the latter may be used for local directories served by the web server. If
# your system is serving content from a sub-directory in /srv you must allow
# access here, or in any related virtual host.
<Directory />
```

```
    Options FollowSymLinks
    AllowOverride None
    Require all denied
</Directory>

<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>

#<Directory /srv/>
#   Options Indexes FollowSymLinks
#   AllowOverride None
#   Require all granted
#</Directory>




# AccessFileName: The name of the file to look for in each directory
# for additional configuration directives.  See also the AllowOverride
# directive.
#
AccessFileName .htaccess

#
# The following lines prevent .htaccess and .htpasswd files from being
# viewed by Web clients.
```

```
#
<FilesMatch "^\.ht">
    Require all denied
</FilesMatch>



#
# The following directives define some format nicknames for use with
# a CustomLog directive.
#
# These deviate from the Common Log Format definitions in that they use %O
# (the actual bytes sent including headers) instead of %b (the size of the
# requested file), because the latter makes it impossible to detect partial
# requests.
#
# Note that the use of %{X-Forwarded-For}i instead of %h is not recommended.
# Use mod_remoteip instead.
#
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" vhost_combined
LogFormat "%h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %O" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Création des Templates

Le fichier **vhosts.pp** fait référence à deux fichiers de gabarit (templates). Naviguez donc au répertoire **/etc/puppetlabs/code/environments/production/modules/apache/templates** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/files$ cd ../templates/
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/templates$
```

Créez le fichier **/etc/puppetlabs/code/environments/production/modules/apache/templates/vhosts-rh.conf.erb** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/templates$ sudo vi vhosts-rh.conf.erb
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/templates$ cat vhosts-rh.conf.erb
<VirtualHost *:80>
    ServerAdmin <%= @adminemail %>
    ServerName <%= @servername %>
    ServerAlias www.<%= @servername %>
    DocumentRoot /var/www/<%= @servername -%>/public_html/
    ErrorLog /var/www/<%= @servername -%>/logs/error.log
    CustomLog /var/www/<%= @servername -%>/logs/access.log combined
</VirtualHost>
```

Ainsi que le fichier **/etc/puppetlabs/code/environments/production/modules/apache/templates/vhosts-deb.conf.erb** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/templates$ sudo vi vhosts-deb.conf.erb
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/templates$ cat vhosts-deb.conf.erb
<VirtualHost *:80>
    ServerAdmin <%= @adminemail %>
    ServerName <%= @servername %>
    ServerAlias www.<%= @servername %>
    DocumentRoot /var/www/html/<%= @servername -%>/public_html/
    ErrorLog /var/www/html/<%= @servername -%>/logs/error.log
    CustomLog /var/www/html/<%= @servername -%>/logs/access.log combined
    <Directory /var/www/html/<%= @servername -%>/public_html>
```

```
        Require all granted
    </Directory>
</VirtualHost>
```

Naviguez au répertoire **/etc/puppetlabs/code/environments/production/modules/apache** et vérifiez l'arborescence du module :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/templates$ cd ..
toto@master:/etc/puppetlabs/code/environments/production/modules/apache$ tree
.
├── examples
├── files
│   ├── apache2.conf
│   └── httpd.conf
├── manifests
│   ├── init.pp
│   ├── params.pp
│   └── vhosts.pp
└── templates
    ├── vhosts-deb.conf.erb
    └── vhosts-rh.conf.erb
```

Déployer Apache

Retournez au répertoire **/etc/puppetlabs/code/environments/production/modules/apache/manifests** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache$ cd manifests/
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$
```

Vérifiez la syntaxe des manifests **init.pp**, **params.pp** et **vhosts.pp** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ sudo
/opt/puppetlabs/bin/puppet parser validate init.pp params.pp vhosts.pp
```

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$
```

Naviguez vers le répertoire **/etc/puppetlabs/code/environments/production/modules/apache/examples** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/manifests$ cd ../examples/
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/examples$
```

Créez le fichier **init.pp** pour définir les valeurs des deux variables **$adminemail** et **$servername** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/examples$ sudo vi init.pp
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/examples$ cat init.pp
$adminemail = 'webmaster@i2tch.loc'
$servername = 'i2tch.loc'

include apache
include apache::vhosts
```

Naviguez au répertoire **/etc/puppetlabs/code/environments/production/modules/apache** et vérifiez l'arborescence du module :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/examples$ cd ..
toto@master:/etc/puppetlabs/code/environments/production/modules/apache$ tree
.
├── examples
│   └── init.pp
├── files
│   ├── apache2.conf
│   └── httpd.conf
├── manifests
│   ├── init.pp
│   ├── params.pp
│   └── vhosts.pp
└── templates
    ├── vhosts-deb.conf.erb
    └── vhosts-rh.conf.erb
```

```
4 directories, 8 files
```

Retournez au répertoire **/etc/puppetlabs/code/environments/production/modules/apache/examples** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache$ cd examples/
```

Testez l'application avec l'option **–noop** :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/examples$ sudo /opt/puppetlabs/bin/puppet
apply --noop init.pp
Notice: Compiled catalog for master.i2tch.loc in environment production in 0.03 seconds
Notice: /Stage[main]/Apache/Package[apache]/ensure: current_value 'purged', should be 'present' (noop)
Notice: /Stage[main]/Apache/File[configuration-file]/ensure: current_value 'absent', should be 'file' (noop)
Notice: /Stage[main]/Apache/Service[apache-service]: Would have triggered 'refresh' from 1 event
Notice: Class[Apache]: Would have triggered 'refresh' from 3 events
Notice: /Stage[main]/Apache::Vhosts/File[/etc/apache2/sites-available/i2tch.loc.conf]/ensure: current_value
'absent', should be 'file' (noop)
Notice: /Stage[main]/Apache::Vhosts/File[/var/www/i2tch.loc]/ensure: current_value 'absent', should be
'directory' (noop)
Notice: /Stage[main]/Apache::Vhosts/File[/var/www/i2tch.loc/public_html]/ensure: current_value 'absent', should
be 'directory' (noop)
Notice: /Stage[main]/Apache::Vhosts/File[/var/www/i2tch.loc/logs]/ensure: current_value 'absent', should be
'directory' (noop)
Notice: Class[Apache::Vhosts]: Would have triggered 'refresh' from 4 events
Notice: Stage[main]: Would have triggered 'refresh' from 2 events
Notice: Applied catalog in 0.14 seconds
```

Naviguez au répertoire **manifests** de l'environnement :

```
toto@master:/etc/puppetlabs/code/environments/production/modules/apache/examples$ cd
/etc/puppetlabs/code/environments/production/manifests
toto@master:/etc/puppetlabs/code/environments/production/manifests$
```

Modifiez le fichier **site.pp** :

```
toto@master:/etc/puppetlabs/code/environments/production/manifests$ sudo vi site.pp
toto@master:/etc/puppetlabs/code/environments/production/manifests$ cat site.pp
node default {

}

node 'master.i2tch.loc' {

  include accounts

  resources { 'firewall':
    purge => true,
  }

  Firewall {
    before        => Class['firewall::post'],
    require       => Class['firewall::pre'],
  }

  class { ['firewall::pre', 'firewall::post']: }

  firewall { '200 Allow Puppet Master':
    dport         => '8140',
    proto         => 'tcp',
    action        => 'accept',
  }

}

node 'slave01.i2tch.loc' {
  $adminemail = 'webmaster@i2tch.loc'
  $servername = 'slave01.i2tch.loc'

  include accounts
```

```
  include apache
  include apache::vhosts

  resources { 'firewall':
    purge => true,
  }

  Firewall {
    before        => Class['firewall::post'],
    require       => Class['firewall::pre'],
  }

  class { ['firewall::pre', 'firewall::post']: }

  }

node 'slave02.i2tch.loc' {
  $adminemail = 'webmaster@i2tch.loc'
  $servername = 'slave02.i2tch.loc'

  include accounts
  include apache
  include apache::vhosts

  resources { 'firewall':
    purge => true,
  }

  Firewall {
    before        => Class['firewall::post'],
    require       => Class['firewall::pre'],
  }

  class { ['firewall::pre', 'firewall::post']: }
```

```
  }
```

Exécutez l'agent de Puppet sur la machine slave01 :

```
toto@slave01:~$ sudo /opt/puppetlabs/bin/puppet agent -t
[sudo] password for toto:
Info: Using configured environment 'production'
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Retrieving locales
Info: Loading facts
Info: Caching catalog for slave01.i2tch.loc
Info: Applying configuration version '1582373993'
...
Notice: Applied catalog in 24.16 seconds
```

Ainsi que la machine slave02 :

```
-bash-4.2$ sudo /opt/puppetlabs/bin/puppet agent -t
[sudo] Mot de passe de toto :
Info: Using configured environment 'production'
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Retrieving locales
Info: Loading facts
Info: Caching catalog for slave02.i2tch.loc
Info: Applying configuration version '1582374068'
...
Notice: Applied catalog in 39.38 seconds
```
