Puppet Enterprise Report (S3540510)

Introduction

This document serves to show how I coded my puppet manifests files, and the reasoning for the decisions made.

The proof of it working is included as a .txt file containing the debug output of the command 'puppet agent -t -d'

Puppet Enterprise Report (S3540510)

Task	Code	Description

Create users (User management)	`# Creates all users`
	`class user {`
	`  require groups`
	`  require packages`
	
	`  user { 'becca' :`
	`    ensure     => present,`
	`    uid        => '10010510',`
	`    home       => '/home/becca',`
	`    managehome => true,`
	`    password   => '$1$j.cuPw6w$xEqjdNmz.qgs2lgZgwTqQ.',`
	`    groups     => ['sysadmin', 'cars'],`
	`    shell      => '/bin/bash',`
	`  }`
	
	`  user { 'fred' :`
	`    ensure     => present,`
	`    uid        => '10020510',`
	`    home       => '/home/fred',`
	`    managehome => true,`
	`    password   => '$1$j.cuPw6w$xEqjdNmz.qgs2lgZgwTqQ.',`
	`    groups     => ['trucks', 'cars', 'wheel'],`
	`    shell      => '/bin/csh',`
	`  }`
	
	`  user { 'wilma' :`
	`    ensure         => present,`
	`    uid            => '10030510',`
	`    home           => '/home/wilma',`
	`    managehome     => true,`
	`    password       => '$1$j.cuPw6w$xEqjdNmz.qgs2lgZgwTqQ.',`
	`    groups         => ['trucks', 'cars', 'ambulances'],`
	`    shell          => '/bin/bash',`
	`    purge_ssh_keys => true,`
	`  }`
	`  # SSH Key for wilma's account`
	`  -> ssh_authorized_key { 'wilmaKey' :`
	`    ensure => present,`
	`    user   => 'wilma',`
	`    type   => 'ssh-rsa',`
	`    key    => 'ThisIsWilmasKeyAbCdE',`
	`  }`
	`}`
		Every user is created in the same class, with the requirement for groups and packages to be instated prior to creating the users.
		This is required due to the assignment of groups to users which would fail without it, as well as the requirement for /bin/csh needing to exist before setting Fred's shell.
		
		Passwords are encrypted with SHA-1 on the linux machine then pasted into the file. I could not use the sha-1('!!') function as it would not allow me to log in with the password when attempting to 'su – becca'.
		
		Fred has been assigned the group 'wheel' as it is the redhat group to enable sudo privileges without manually changing the sudoers file.
		
		Wilma's ssh key is created straight after her user is created.

Puppet Enterprise Report (S3540510)

Task	Code	Description

Ensure packages are installed (Package Management)	`# Ensures that all packages specified are installed and installed`
	`class packages{`
	`  package { 'wget' :`
	`    ensure   => 'installed',`
	`    provider => 'yum',`
	`  }`	

	`  # This is needed for yum or things will be missing`
	`  exec { 'getDeps' :`
	`    onlyif  => '/usr/bin/yum-config-manager',`
	`    command => '/usr/bin/yum-config-manager --enable rhui-REGION-rhel-server-optional',`
	`  }`
		As yum doesn't have as many records of packages as apt-get or other package providers, I have required a dependency download before other packages are installed.

	`  # Install cgdb`
	`  package { 'cgdb' :`
	`    ensure   => 'installed',`
	`    source   => 'ftp://mirror.switch.ch/pool/4/mirror/epel/7/x86_64/c/cgdb-0.6.8-1.el7.x86_64.rpm',`
	`    provider => 'rpm',`
	`  }`
		Gcdb does not exist on the Yum repository, so I found an RPM and installed using the rpm provider instead.

	`  # Install fuse-sshfs`
	`  package { 'glib*' :`
	`    ensure   => 'installed',`
	`    provider => 'yum',`
	`  }`
	`  -> package { 'fuse*' :`
	`    ensure   => 'installed',`
	`    provider => 'yum',`
	`  }`
	`  -> package { 'fuse-sshfs' :`
	`    ensure   => 'installed',`
	`    source   => 'ftp://195.220.108.108/linux/dag/redhat/el7/en/x86_64/dag/RPMS/fuse-sshfs-2.5-1.el7.rf.x86_64.rpm',`
	`    provider => 'rpm',`
	`  }`
		Fuse-sshfs also does not exist on Yum, so I had to install using the same method as gcdb. As fuse-sshfs requires dependencies, I also ensured that they were downloaded and installed prior to fuse-sshfs.

	`  # Install dia2code`
	`  package { 'libxml2.so.2' :`
	`    ensure => installed,`
	`  }`
	`  -> package { 'dia2code' :`
		Dia2Code is similar to Fuse-sshfs in that it has dependencies required, so the same method was used.

Puppet Enterprise Report (S3540510)

Task	Code	Description
	ensure => 'installed', source => 'https://downloads.sourceforge.net/project/dia2code/dia2code/0.8.3/dia2code-0.8.3-3.1.i586.rpm?r=http%3A%2F%2Fdia2code.sourceforge.net%2Fdownload.html&ts=1507292790&use_mirror=ncu', provider => 'rpm', } # Install mysql-server package { 'mysql57-community-release' : ensure => 'installed', source => 'https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm', provider => 'rpm', } # Install additional packages that exist on Yum $packages = ['openssh-server', 'httpd', 'mysql', 'tigervnc-server', 'tmux', 'lynx', 'gcc', 'gdb', 'vim', 'emacs', 'csh',] package { $packages : ensure => 'installed', provider => 'yum', } }	Mysql-server was also installed using RPM as Yum did not have a reference to it. The remainder of packages existed on Yum after the dependency update at the top of the file, so I created an array of them, Then installed them all using a reference to the variable and specifying the Yum package provider.

Puppet Enterprise Report (S3540510)

Task	Code	Description

Create groups (User management)	`# Creates all groups`
	`class groups {`
	`  group{'sysadmin':`
	`  ensure => present,`
	`  gid  => '500',`
	`  }`
	
	`  group{'cars':`
	`  ensure => present,`
	`  gid  => '501',`
	`  }`
	
	`  group{'trucks':`
	`  ensure => present,`
	`  gid  => '502',`
	`  }`
	
	`  group{'ambulances':`
	`  ensure => present,`
	`  gid  => '503',`
	`  }`
	`}`
		Groups are required for each user.
		Groups id starts at 500 as ID's below 500 are used by services and could be taken.
		
		I do not think further explanation is required.

Ensure that httpd is run on boot and subscribes to conf file. (Package Configuration)	`# Handles management of services, and initial setup`
	`class services {`
	`  service { 'httpd' :`
	`  ensure => running,`
	`  enable => true,`
	`  }`
	
	`  # Overwrite httpd.conf with custom configuration`
	`  ~> file { '/etc/httpd/conf/httpd.conf' :`
	`  ensure  => present,`
	`  notify  => Service['httpd'],`
	`  mode  => '0777',`
	`  owner  => 'root',`
	`  group  => 'root',`
	`  source  =>`
	`'/etc/puppetlabs/code/environments/production/manifests/configfiles/httpd.conf',`
	`  require => Package['httpd'],`
	`  }`
	`}`
		Httpd is the only service that can be enabled for boot as it is apache.
		The other packages aren't an ongoing service.
		
		Httpd is subscribed to the httpd.conf file, so it will restart whenever the file is changed.

Puppet Enterprise Report (S3540510)

Task	Code	Description

Enable SSH and subscribe to file (Package Configuration)	`# Ensures ssh is running and that root access is disabled`
	`class openssh {`
	`  service { 'sshd' :`
	`    ensure  => running,`
	`    enable  => true,`
	`    require => Package['openssh-server'],`
	`  }`
	
	`  # Overwrite sshd_config with custom one.`
	`  # Augeas cannot handle commenting.`
	`  ~> file { '/etc/ssh/sshd_config':`
	`    ensure  => present,`
	`    notify  => Service['sshd'],`
	`    mode    => '0777',`
	`    owner   => 'root',`
	`    group   => 'root',`
	`    source  =>`
	`'/etc/puppetlabs/code/environments/production/manifests/configfiles/sshd_config',`
	`    require => Package['openssh-server'],`
	`  }`
	`}`
		Openssh will start on system boot,
		
		and is subscribed to the config file (so it will restart when the file changes).
		The config file is a custom file that we were told we could use on blackboard, and is copied across to the server from a local location.
		This is due to Augeas being unable to uncomment lines in a file.

	`# Sets run interval, agent timestamp display, inclusion of /usr/local/bin,`	

Puppet Enterprise Report (S3540510)

Task	Code	Description

Agent Configuration	`# Becca sudoing, mounting Titan onto becca's drive.`
	`class iniconfig {`
	`  augeas { 'agent_runinterval' :`
	`  context => '/etc/puppetlabs/puppet/puppet.conf/agent',`
	`  changes => 'set runinterval 20m',`
	`  }`
	
	`  exec { 'runlevel' :`
	`  command => '/usr/bin/systemctl set-default multi-user.target',`
	`  }`
	
	`  # Set the timestamp to client`
	`  file { '/etc/profile.d/agent_login.sh' :`
	`  ensure  => present,`
	`  mode  => '0777',`
	`  owner  => 'root',`
	`  group  => 'root',`
	`  content => 'timeStamp=`/bin/date +"%d-%m-%Y_%H.%M.%S"`; echo "Agent started running at $timeStamp"',`
	`  }`
	
	`  # Include /usr/local/bin to user`
	`  file {'/etc/profile.d/set-user-bin.sh' :`
	`  owner  => 'root',`
	`  mode  => '0644',`
	`  content => 'PATH=$PATH:/usr/local/bin',`
	`  }`
	
	`  # Give Becca Sudoers privilege`
	`  exec { 'give_sudo_becca' :`
	`  command => '/usr/sbin/usermod -aG wheel becca',`
	`  }`
	
	`  # Mount titan over becca (/home/becca/titan)`
	`  # Make dir to use, connect using sshfs; only if it's not mounted already`
		Agent checkin 3 times per hour, achieved through augeas.
		
		Setting the default run-level to 3.
		Manual changes to the file that used to handle it no longer have any effect, so it must be achieved through a command instead.
		
		A timestamp is displayed whenever any user logs into an agent.
		
		/usr/local/bin is added to the default system path
		whenever a user logs in.
		Achieved the same way as the timestamp.
		
		Becca is given sudo privilege by editing the sudo file.
		It is bad practice to edit it directly, so a command is run instead.

Puppet Enterprise Report (S3540510)

Task	Code	Description
	exec { 'mount_titan_becca' : command => '/usr/bin/mkdir /home/becca/titan; echo \ "$(cat /etc/puppetlabs/code/environments/production/manifests/configfiles/ssh_pass)" | \ /usr/bin/sshfs -o StrictHostKeyChecking=no -o password_stdin s3540510@titan.csit.rmit.edu.au:/home/sl0/S3540510/ /home/becca/titan/', unless => '/usr/bin/find /home/becca/titan -mindepth 1 | /usr/bin/read', } }	SSHFS is used to mount my personal RMIT files from Titan into becca's home directory. A file containing my password is stored locally and referenced, maintaining security. The password is then supplied via piping to stdin, utilising the 'pasword_stdin' option in sshfs mounting command. Line breaks are required to pass puppet-lint.