## Contents

1 SIMP Installation Guide
   1.1 Introduction ........................................... 3
   1.2 SIMP Server Installation ................................ 4
   1.3 Client Management ...................................... 7
   1.4 Hiera Overview ......................................... 9
   1.5 Apply Certificates ...................................... 11
   1.6 Configure the PXE Boot .................................. 14
   1.7 SIMP 4.2.0-Beta ....................................... 15
   1.8 Glossary of Terms ...................................... 22
   1.9 Appendix A - Sample Puppet Files ....................... 24

2 SIMP User Guide
   2.1 Introduction ........................................... 31
   2.2 User Management ....................................... 32
   2.3 Client Management ..................................... 37
   2.4 Configure the PXE Boot .................................. 38
   2.5 Maximum Number of Nodes ............................... 40
   2.6 Apply Certificates ...................................... 41
   2.7 SIMP Administration .................................... 42
   2.8 Backing up the Puppet Master ........................... 45
   2.9 Managing Workstation Infrastructures ................. 45
   2.10 VNC ..................................................... 49
   2.11 Upgrading SIMP ........................................ 51
   2.12 OpenStack Integration .................................. 54
   2.13 Logstash ............................................... 54
   2.14 Using Kerberos 5 in SIMP ............................... 58
   2.15 Jenkins Scripts ........................................ 59
   2.16 Troubleshooting SIMP ................................... 67
   2.17 SIMP FAQs ............................................ 68
   2.18 SIMP RPMs ............................................ 80
   2.19 SIMP 4.2.0-Beta ....................................... 81
   2.20 Glossary of Terms ...................................... 87
   2.21 Appendix A - Sample Puppet Files ....................... 90

3 SIMP Security Concepts .................................. 97
   3.1 Introduction ........................................... 97
   3.2 Technical Security ..................................... 97
   3.3 Operational Security ................................... 104
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.4 Information System Management</td>
<td>106</td>
</tr>
<tr>
<td>3.5 Security Concepts Appendices</td>
<td>106</td>
</tr>
<tr>
<td>4 License</td>
<td>179</td>
</tr>
<tr>
<td>5 Contact</td>
<td>181</td>
</tr>
<tr>
<td>6 Help</td>
<td>183</td>
</tr>
<tr>
<td>7 Indices and tables</td>
<td>185</td>
</tr>
</tbody>
</table>
This is the 4.2.0 release of the SIMP product based on the 6.6 release of CentOS and Red Hat Enterprise Linux (RHEL).

SIMP is a minimized, security-focused, continually managed operating system overlay compatible with RHEL and CentOS. Each system is tailored to the hosted project’s needs using Puppet.

Fundamentally, SIMP is a framework that is designed to be secure from a practical point of view out of the box. As a framework, SIMP is designed to be flexed to meet the needs of the end user. If, for some reason, this is not possible, then this should be reported to the developers as a bug.

Contents:
SIMP Installation Guide

Contents:

1.1 Introduction

This guide will walk a user through the process of managing a SIMP system. This system includes, at a minimum, a SIMP server with properly configured networking information and a working Puppet server. Additionally, this document outlines the process of managing clients and users associated with the SIMP system.

1.1.1 Level of Knowledge

SIMP is designed for use by system administrators or users with a strong background using Linux operating systems. The core applications that make up SIMP and require prerequisite knowledge are:

- *Puppet* - 3.7 or later
- *Domain Name System (DNS)* - BIND 9
- *Dynamic Host Configuration Protocol (DHCP)* - Internet Systems Consortium (ISC) DHCP
- *Lightweight Directory Access Protocol (LDAP)* - OpenLDAP
- RedHat Kickstart (including all tools behind it) - *Trivial File Transfer Protocol (TFTP)*, PXELinux, etc.
- Apache
- *Yellowdog Updater, Modified (YUM)*
- Rsyslog Version 3+
- *Internet Protocol Tables (IPtables)* (Basic knowledge of the rules)
- *Auditd* (Basic knowledge of the rules)
- *Advanced Intrusion Detection Environment (AIDE)* (Basic knowledge of the rules)
- Basic X.509 Key Management

By itself, SIMP does as much initial setup and configuration of these tools as possible. However, without at least some understanding, a user will be unable to tailor a SIMP system to fit the desired environment. A general understanding of how to control and manipulate these tools from the command line interface (CLI) will be necessary, as SIMP does not come stock with a graphical user interface (GUI). Additionally, knowledge of scripting and *Ruby* programming will also help to further customize a SIMP install. For example, in order to use the advanced features of Puppet, some Ruby programming is required.
1.1.2 SIMP Defined

SIMP is a continually managed minimal Linux framework compatible with both Red Hat Enterprise Linux (RHEL) and Community Enterprise Operating System (CentOS). By maintaining and managing file-level and network configuration consistency, SIMP addresses process degradation on an operating system level. SIMP uses Puppet to provide multi-system consistency over time while augmenting the software with tools like Capistrano for controlled application of one-time mass operations.

1.2 SIMP Server Installation

This chapter provides guidance on installing and configuring SIMP using the simp config utility.

1.2.1 System Requirements

SIMP scales well, but how much depends on a number of factors, including the number of nodes, the processor speed, the total memory, and the complexity of the manifests. The following minimal system requirements are recommended:

- Central Processing Unit (CPU): 2 Cores
- Random Access Memory (RAM): 2.2 GB
- Hard Disk Drive (HDD): 50 GB

1.2.2 Using the SIMP Utility

The SIMP Utility does not assist users through the entire configuration process; however, it does make the initial configuration easier and more repeatable.

**Important**

Correct time across all systems is important to the proper functioning of SIMP and Puppet in general.

If a user has trouble connecting to the Puppet server and errors regarding certificate validation appear, check the Puppet server and client times to ensure they are synchronized.

Using the configuration script, the following items are configured:

**NOTE** this needs updated for the new puppetserver settings which includes Puppet Environments

- Grub password in /boot/grub/grub.conf
- Basic network setup
- Autosigning in /etc/puppet/autosign.conf
- Fileserver in /etc/puppet/fileserver.conf
- Puppet server and Certificate Authority (CA) information in /etc/puppet/puppet.conf
- /etc/puppet/hieradata/simp_def.yaml
- Server certificates for the puppet server itself (Fake CA)
- Base YUM repositories

**Warning**

Keep in mind as the installation process begins that Puppet does not work well with capital letters in host names. Therefore, they should not be used.
1.2.3 SIMP Default Passwords and Settings

Below is a table containing the default passwords found on a basic SIMP server.

Important

All default passwords should be changed during the initial configuration process.

Table: SIMP Default Passwords

Below is a table containing sample variables and their corresponding values that apply to this SIMP deployment. These variables and values will be helpful in illustrating how configuration files are set up. Your values will obviously differ, depending on your installation environment.

<table>
<thead>
<tr>
<th>Utility</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grub</td>
<td>InitialGruubCredential$</td>
</tr>
<tr>
<td>Root User</td>
<td>Please Change This Immediately!</td>
</tr>
<tr>
<td>Simp User</td>
<td>CorrectHorseBatteryStaple</td>
</tr>
<tr>
<td>Variable name</td>
<td>Value</td>
</tr>
<tr>
<td>Domain name</td>
<td>simp.net</td>
</tr>
<tr>
<td>Fully qualified name</td>
<td>puppet.simp.net</td>
</tr>
<tr>
<td>IP address</td>
<td>192.168.1.10</td>
</tr>
<tr>
<td>Gateway</td>
<td>192.168.1.1</td>
</tr>
<tr>
<td>DNS server</td>
<td>192.168.1.10</td>
</tr>
<tr>
<td>DNS search entry</td>
<td>simp.net</td>
</tr>
<tr>
<td>Kickstart server</td>
<td>192.168.1.10</td>
</tr>
<tr>
<td>Yum server</td>
<td>192.168.1.10</td>
</tr>
<tr>
<td>LDAP URI</td>
<td>ldap://puppet.simp.net</td>
</tr>
<tr>
<td>LDAP Base DN</td>
<td>[dc=simp,dc=net]</td>
</tr>
<tr>
<td>LDAP Root DN</td>
<td>[cn=LDAPAdmin,ou=People,dc=simp,dc=net]</td>
</tr>
<tr>
<td>LDAP Bind DN</td>
<td>[cn=hostAuth,ou=Hosts,dc=simp,dc=net]</td>
</tr>
<tr>
<td>LDAP Sync DN</td>
<td>[cn=LDAPSync,ou=People,dc=simp,dc=net]</td>
</tr>
</tbody>
</table>

Table: Sample values for SIMP install

1.2.4 Preparing the SIMP Server Environment

The following table outlines the steps to prepare a system for SIMP installation.

Table: SIMP Pre-Install Procedures

1.2.5 Installing the SIMP Server

The following table outlines the steps to install a SIMP server.
### Step Process/Action

1. Boot the system and ensure the SIMP ISO is selected.

2. Press Enter to run the standard SIMP install, or choose from the customized options list.

3. When the installation is complete, the system will restart automatically.

4. Log on as root and type the default password shown in Table 2.1.

5. Type the default password again when prompted for the (current) UNIX password.

Type a new password when prompted for the New Password. Retype the password when prompted.

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Log on as simp and su - to gain root access.</td>
</tr>
<tr>
<td>2.</td>
<td>Type simp config</td>
</tr>
<tr>
<td></td>
<td>Type simp config -a <em><strong>&lt;Config File&gt;</strong></em> to load a previously generated configuration instead of generating the configuration from the script. This is the option to run for systems that will be rebuilt often.</td>
</tr>
<tr>
<td></td>
<td>For a list of additional commands, type simp help. Type simp help <em><strong>&lt;Command&gt;</strong></em> for more information on a specific command.</td>
</tr>
<tr>
<td>3.</td>
<td>Configure the system as prompted.</td>
</tr>
<tr>
<td>4.</td>
<td>Type simp bootstrap</td>
</tr>
<tr>
<td></td>
<td>NOTE: If progress bars are of equal length and the bootstrap finishes quickly, a problem has occurred. This is most likely due to an error in SIMP configuration. Refer to the previous step and make sure that all configuration options are correct.</td>
</tr>
<tr>
<td>5.</td>
<td>Type reboot</td>
</tr>
</tbody>
</table>

Table: SIMP Install Procedure

### 1.2.6 Performing Post-installation Setup on the SIMP Server

The following table outlines the SIMP post-installation procedures.
<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Log on as root</td>
</tr>
<tr>
<td>2.</td>
<td>Run puppet for the first time. Errors will appear for DHCP. These can be safely ignored at this stage. Type: puppet agent -t</td>
</tr>
<tr>
<td>3.</td>
<td>Copy CentOS RHEL_MAJOR_MINOR_VERSION ISO(s) to the server and unpack using the unpack_dvd utility. This creates a new tree under /var/www/yum/CentOS. Execute: unpack_dvd CentOS-RHEL_MAJOR_MINOR_VERSION-*####*-x86_64-Everything.iso</td>
</tr>
<tr>
<td>4.</td>
<td>Update your system using yum. The updates applied will be dependent on what ISO you initially used. Execute: yum clean all; yum makecache</td>
</tr>
<tr>
<td>5.</td>
<td>Run puppet. Ignore the same DHCP errors.</td>
</tr>
<tr>
<td>6.</td>
<td>Type reboot</td>
</tr>
</tbody>
</table>

Table: SIMP Post-Installation Procedure

### 1.3 Client Management

This chapter provides guidance to install and configure SIMP clients based on the standard SIMP system installed using the SIMP DVD.

#### 1.3.1 System Requirements

Before installing clients, the system should consist of the following minimum requirements:

- **Hardware/Virtual Machine (VM)**: Capable of running RHEL 6 or 7; 64-bit compatible
- **RAM**: 512 MB
- **HDD**: 5 GB

#### 1.3.2 Configuring the Puppet Master

Perform the following actions as root on the Puppet Master system prior to attempting to install a client.

#### 1.3.3 Configure DHCP

The table below lists the steps to configure DHCP.
Step | Process/Action
--- | ---
1. | Log on as root.
2. | Open the /var/simp/rsync/CentOS/RHEL\_MAJOR\_VERSION/dhcpd/dhcpd.conf file and edit it to suit the necessary environment.

**NOTE:** Enter the hardware ethernet and fixed-address for each client that will be kickstarted. An example dhcpd.conf is listed in the Appendix, where 192.168.1.100 is the client targeted for PXE/provisioning. Be sure to substitute in the actual values appropriate to your environment, including the ethernet address of the target client.

3. | Save and close the file.
4. | Type puppet agent -t --tags dhcpd on the Puppet Master to apply the changes.

Table: DHCP Configuration Procedure

### 1.3.4 Configure DNS

Most static files are pulled over rsync by Puppet in this implementation for network efficiency. Specific directories of interest are noted in this section.

It is possible to use an existing DNS setup; however, the following table lists the steps for a local setup.

Table: DNS Configuration Procedure

### 1.3.5 Configure the Kickstart

The table below lists the steps to configure the kickstart.

Table: Kickstart Configuration Procedure

The diskdetect.sh* script is responsible for detecting the first active disk and applying a disk configuration. Edit this file to meet any necessary requirements or use this file as a starting point for further work.

### 1.3.6 Setting Up the Client

The table below lists the steps to PXE boot the system and set up the client.

Table: PXE Boot Procedure

Upon successful deployment of a new CentOS or RHEL client, it is highly recommended that LDAP administrative accounts be created. See Chapter 2 of the SIMP Users Guide for user management.

### 1.3.7 Troubleshooting Issues

If the client has been kickstarted, but is not communicating with the Puppet server, try the following options:

- Check the forward and reverse DNS entries on the client and server; both must be correct.
• Check the time on the systems. More than an hour’s difference will cause serious issues with certificates.
• Remove /var/lib/puppet/ssl on the client system; run puppet cert --clean ***<Client Host Name>*** on the Puppet server; and try again.

1.3.8 Troubleshoot Certificate Issues

If host certificates do not appear to be working and the banner is not getting rsync’d to the clients, ensure that all certificates verify against the installed CA certificates.

The table below lists the steps to determine which certificates are working and which are not.

Table: Certificate Verification Procedure

If the TXT_DB error number 2 appears, revoke the certificate that is being regenerated. The table below lists the steps to revoke the certificate.

Table: Revoke Certificate Procedure

1.4 Hiera Overview

SIMP now uses Hiera natively instead of Extdata. From Puppet Labs website: Hiera is a key/value lookup tool for configuration data, built to set node-specific data without repeating yourself. It is an attempt to make SIMP more configurable to you, the end user. It configures Puppet in two ways: automatic parameter lookup/hiera lookup functions, and assigning classes to nodes. The former allows you to generate reusable code and concentrates parameter assignment to one directory. The latter is a supplement to the failed inheritance model.

1.4.1 Setting Parameters

Automatic Lookup

You can now safely declare any class on any node with ‘include’, even if the class is parametized. Before Hiera, this was not possible. Puppet will automatically retrieve class parameters from Hiera using keys. Add a key with a value pair to an appropriate yaml file, say default.yaml, as such:

Adding a Key/Value Pair to Hiera Examples

```yaml
---
classfoo::parameter_bar: "Woo"
classfoo::parameter_baz: "Hoo"
```

You can then ‘include classfoo’ on any node, with parameter_bar and parameter_baz defaulting to Woo and Hoo, respectively.

Lookup Functions

You are not required to set up your hierarchy for automatic variable lookup. Using three functions, you can query Hiera for any key.

The first is ‘hiera’. This uses standard priority lookup and can retrieve values of any data type from Hiera. If no key is found, a default should be included. $myvar = hiera('parameter_bar', 'Woo')

The second is ‘hiera_array’. This uses an array merge lookup. It retrieves all array values for a given key throughout the entire hierarchy and flattens them into a single array.

The third is ‘hiera_hash’. This uses a hash merge lookup. It retrieves all hash values for a given key throughout the entire hierarchy and merges them into a single hash.
1.4.2 Assigning Classes to Nodes

Assigning classes to nodes is done with the ‘hiera_include’ function. Hiera does an array merge lookup on ‘tags’ to retrieve classes which should be included on a node. In SIMP, we place hiera_include('classes') in /etc/puppet/manifests/site.pp. Since site.pp is outside of any node definition and below all top scope variables, every node controlled by puppet will get every class tagged with ‘classes’ in its hierarchy. Additionally, simp_def.yaml in is the hierarchy of every node, so every node will receive those classes (by default).

1.4.3 Assigning Defined Types to Nodes

Defined types do not have the ability to receive parameters via Hiera in the traditional sense. To include a defined type on a node, one could use create_resources, but this is messy and discouraged. Instead, make a site class, /etc/puppet/modules/site/manifests/my_site.pp. For example, to include tftpboot linux_model and assign_host on your puppet server, puppet.your.domain:

Adding a Site Manifest Examples

```
# in /etc/puppet/modules/site/manifests/tftpboot.pp
# Set KSSERVER statically or use Hiera for lookup

class site::tftpboot {
  include 'tftpboot'

  tftpboot::linux_model { 'CentOS_RHEL_MAJOR_VERSION':
    kernel => 'centosRHEL_MAJOR_VERSION_x86_64/vmlinuz',
    initrd => 'centosRHEL_MAJOR_VERSION_x86_64/initrd.img',
    ks => "http://KSSERVER/ks/pupclient_x86_64.cfg",
    extra => 'ipappend 2'
  }

  tftpboot::assign_host { 'default': model => 'CentOS_RHEL_MAJOR_VERSION' }
}
```

Then, in /etc/puppet/hieradata/hosts/puppet.your.domain.yaml

Adding TFTP Site to Hiera Examples

```
---
classes:
  - 'site::tftpboot'
```

1.4.4 SIMP Hiera File Structure

/etc/puppet/hiera.yaml Hiera’s config file, used to control the hierarchy of your backends.

/etc/puppet/hieradata/ Default location of the yaml files which contain your node data

/etc/puppet/hieradata/simp_classes.yaml The list of default classes to include on any SIMP system.

/etc/puppet/hieradata/simp_def.yaml Contains the variables needed to configure a working SIMP system. Modified by simp-config.

/etc/puppet/hieradata/hosts/ By populating this directory with some.host.name.yaml file, you can assign parameters to host some.host.name

/etc/puppet/hieradata/domains/ Same principal as hosts, but domain names.

/etc/puppet/manifests/ Contains site.pp and all other node manifests. BE CAREFUL when modifying this directory, site.pp contains your globals. This directory can be used to supplement or even REPLACE Hiera, with nodes. Note
that Hiera cannot regex hostnames to apply manifests, so a node manifest will have to be created here if you wish to have that ability.

## 1.5 Apply Certificates

This section provides guidance on obtaining official certificates and generating a Fake CA.

### 1.5.1 Obtaining Official Certificates

All SIMP systems must have Public Key Infrastructure (PKI) keypairs generated for the server. These keys reside in the `/etc/puppet/keydist` directory and are served to the clients over the Puppet protocol.

**Note**

These keypairs are not the keys that the Puppet server uses for its operation. Do not get the two confused.

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Type <code>mkdir /etc/puppet/environments/simp/modules/pki/files/keydist/&lt;Client System FQDN&gt;***</code></td>
</tr>
<tr>
<td>3.</td>
<td>Type <code>chown -R root.puppet /etc/puppet/environments/simp/modules/pki/files/keydist</code></td>
</tr>
<tr>
<td>4.</td>
<td>Type <code>chmod -R u=rwX,g=rX,o-rwx /etc/puppet/environments/simp/modules/pki/files/keydist</code></td>
</tr>
</tbody>
</table>

The table below lists the steps to add any keys for the server that were received from a proper CA to `/etc/puppet/keydist`.

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Type <code>mkdir /etc/puppet/keydist/&lt;Client System FQDN&gt;***</code></td>
</tr>
<tr>
<td>2.</td>
<td>Type `mv **<em>&lt;Certificate Directory&gt;</em>/&lt;FQDN&gt;*.[pem</td>
</tr>
<tr>
<td>3.</td>
<td>Type <code>chown -R root.puppet /etc/puppet/keydist</code></td>
</tr>
<tr>
<td>4.</td>
<td>Type <code>chmod -R u=rwX,g=rX,o-rwx /etc/puppet/keydist</code></td>
</tr>
</tbody>
</table>

Table: Official Certificates Procedure

The table below lists the steps to create and populate the `/etc/puppet/keydist/cacerts` directory.
### Directory Creation Procedure

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Type <code>cd /etc/puppet/environments/simp/modules/pki/files/keydist</code></td>
</tr>
<tr>
<td>2.</td>
<td>Type <code>mkdir cacerts</code> and copy the root CA public certificates into <code>cacerts</code> in <em>Privacy Enhanced Mail (PEM)</em> format (one per file).</td>
</tr>
<tr>
<td>3.</td>
<td>Type <code>cd cacerts</code></td>
</tr>
<tr>
<td>4.</td>
<td>Type <code>for file in *.pem; do ln -s $file </code>openssl x509 -in $file -hash -noout<code>.0; done</code></td>
</tr>
</tbody>
</table>

1. Type `cd /etc/puppet/keydist` |
2. Type `mkdir cacerts` and copy the root CA public certificates into `cacerts` in *Privacy Enhanced Mail (PEM)* format (one per file). |
3. Type `cd cacerts` |
4. Type `for file in *.pem; do ln -s $file `openssl x509 -in $file -hash -noout`.0; done` |

Table: `/etc/puppet/keydist/cacerts` Directory Creation Procedure

### 1.5.2 Generating Fake CAs

If server certificates have not or could not be obtained at the time of client installation, the SIMP team provides a way to create them for the system so that it will work until proper certificates are provided.

**Note**  
This option should not be used for any operational system that can use proper enterprise PKI certificates.

The table below lists the steps to generate the Fake CAs.
<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Type <code>cd /etc/puppet/environments/simp/FakeCA</code></td>
</tr>
<tr>
<td>2.</td>
<td>Type <code>vi togen</code></td>
</tr>
<tr>
<td>3.</td>
<td>Remove old entries from the file and add the <em>Fully Qualified Domain Name (FQDN)</em> of the systems (one per line) for which certificates will be created.</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> To use alternate DNS names for the same system, separate the names with commas and without spaces. For example, <code>.name,alt.name1,alt.name2</code>.</td>
</tr>
<tr>
<td>4.</td>
<td>Type <code>wc cacertkey</code></td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> Ensure that the <em>cacertkey</em> file is not empty. If it is, enter text into the file; then save and close the file.</td>
</tr>
<tr>
<td>5.</td>
<td>Type <code>.gencerts_nopass.sh auto</code></td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> To avoid using the default Fake CA values, remove the Auto statement from the <code>.gencerts_nopass.sh</code> command.</td>
</tr>
</tbody>
</table>

1.5. **Apply Certificates**

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Type <code>cd /etc/puppet/Config/FakeCA</code></td>
</tr>
<tr>
<td>2.</td>
<td>Type <code>vi togen</code></td>
</tr>
<tr>
<td>3.</td>
<td>Remove old entries from the file and add the <em>Fully Qualified Domain Name (FQDN)</em> of the systems (one per line) for which certificates will be created.</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> To use alternate DNS names for the same system, separate the names with commas and without spaces. For example, <code>.name,alt.name1,alt.name2</code>.</td>
</tr>
<tr>
<td>4.</td>
<td>Type <code>wc cacertkey</code></td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> Ensure that the <em>cacertkey</em> file is not empty. If it is, enter text into the file; then save and close the file.</td>
</tr>
<tr>
<td>5.</td>
<td>Type <code>.gencerts_nopass.sh auto</code></td>
</tr>
</tbody>
</table>
Table: Generating Fake CAs Procedure

**Warning**

If the `clean.sh` command is run after the certificates have been generated, the running system will break. To troubleshoot certificate problems, see the section at the end of this chapter.

If issues arise while generating keys, type `cd /etc/puppet/Config/FakeCA` to navigate to the `/etc/puppet/Config/FakeCA` directory, then type `/clean.sh` to start over.

After running the `clean.sh` script, type `/gencerts_nopass.sh` to run the script again using the previous procedure table.

### 1.6 Configure the PXE Boot

In order to **Preboot Execution Environment (PXE)** boot clients, a copy of the ISOs for all versions of RHEL being kickstarted is required.

#### 1.6.1 Setting Up the Kickstart

The system follows the standard kickstart model. Kickstart files are placed in the `/var/www/ks` directory. Custom packages are placed in an appropriate repository created under the `/var/www/yum` directory.

Once the model is ready, the default SIMP settings provide access to the user’s trusted subnets as defined in the `../only:: not simp_4` directory.

```
/etc/puppet/environments/simp/hieradata/simp_def.yaml directory.
/etc/puppet/hieradata/simp_def.yaml directory.
```

The `pupclient_x86_64.cfg` file in the `/var/www/ks` directory is used as an example in the following sections.

**Note**

Sample kickstart templates have been provided in the `ks` directory on the SIMP DVD at the `root` level.

#### 1.6.2 Setting up TFTP

This section describes the process of setting up static files and manifests for TFTP.

**Static Files**

Type `cd /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot` and then type `ls` to check for the existence of the `/var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/rhel<Version>-<Architecture>` directory.

If the directory does not exist, create one in that location and add the `vmlinuz` and `initrd.img` files from the `images/pxeboot` directory of the SIMP DVD. An example is provided below for setting up the CentOS `RHEL_MAJOR_MINOR_VERSION` distribution.

```
cd /var/simp/rsync/CentOS/RHEL_MAJOR_MINOR_VERSION/tftpboot/linux-install
mkdir centosRHEL_MAJOR_MINOR_VERSION_x86_64; cd centosRHEL_MAJOR_MINOR_VERSION_x86_64
cp -p /var/www/yum/CentOS/RHEL_MAJOR_MINOR_VERSION_x86_64/images/pxeboot/*
```
cd ..

chmod 640 centosRHEL\_MAJOR\_MINOR\_VERSION\_x86\_64; chown root:nobody centosRHEL\_MAJOR\_MINOR\_VERSION\_x86\_64

unlink centosRHEL\_MAJOR\_VERSION\_x86\_64

ln -s centosRHEL\_MAJOR\_MINOR\_VERSION\_x86\_64 centosRHEL\_MAJOR\_VERSION\_x86\_64

**Manifest**

Assuming that the Puppet server is being used, create and add the following example code to a site manifest, `/etc/puppet/environment/simp/modules/site/manifests/tftpboot.pp`. Keep in mind that the code varies based on the model being kickstarted.

**Source Code for Setting Up TFTP on Puppet Server TFTP Examples**

```plaintext
# Set KSSERVER statically or use Hiera for lookup
class site::tftpboot {
    include 'tftpboot'

tftpboot::linux_model { 'CentOS_RHEL_MAJOR_VERSION':
    kernel => 'centosRHEL_MAJOR_VERSION_x86_64/vmlinuz',
    initrd => 'centosRHEL_MAJOR_VERSION_x86_64/initrd.img',
    ks   => "http://KSSERVER/ks/pupclient_x86_64.cfg",
    extra => "ksdevice=bootif\nipappend 2"
}

tftpboot::assign_host { 'default': model => 'CentOS_RHEL_MAJOR_VERSION' }
}
```

Next, add the tftpboot site manifest to your puppet server node via Hiera. If it does not already exist, create `/etc/puppet/hieradata/hosts/your.server.fqdn.yaml`. Add the following example code to that yaml file.

**Source Adding TFTP Site Manifest to Hiera TFTP Examples**

```plaintext
---
classes:
    - 'site::tftpboot'
```

After updating the above file, type `puppet agent -t --tags tftpboot` on the Puppet server.

### 1.7 SIMP 4.2.0-Beta

#### 1.7.1 Changelog
SIMP 4.2.0-Beta

Package: 4.2.0-Beta

This release is known to work with:

- RHEL 6.6 x86_64
- CentOS 6.6 x86_64

Significant Updates

- The rsyslog module has been completely rewritten to support rsyslog 7.4. This is a breaking change from previous releases and will require active updates to existing systems. All modules with rsyslog integration are been updated to accommodate this change:
  - aide
  - apache
  - auditd
  - dhcp
  - logstash
  - openldap
  - rsync
  - simp
  - sudosh

- In RHEL6, we updated the OpenLDAP password policy overlay to not conflict with the upcoming 6.7 update. However, this will require you to update your LDAP server schema manually with the attached LDIF. Additionally, there was a bug in previous versions of SIMP that can be fixed by running this LDIF as is in RHEL7 and replacing simp_check_password.so with check_password.so in RHEL6.

- The Electrical and SIMP modules for elasticsearch have been combined.
Upgrade Guidance

Fully detailed upgrade guidance can be found in the Upgrading SIMP portion of the User's Guide.

**Warning:** You must have at least **2GB** of **free** RAM on your system to upgrade to this release.

**Note:** Upgrading from releases older than 4.0 is not supported.

**Expectations**  Before you begin, please be aware that the following actions will take place as a result of the migration script:

- The `puppet-server` RPM will be removed
- The `puppetserver` RPM will be installed (no, that’s not a typo)
- **ALL** SIMP Puppet code will be migrated into a new `simp` environment
  - This will be located at `/etc/puppet/environments/simp`
- A backup of your running environment will be made available at `/etc/puppet/environments/pre_migration.simp`
  - You will find timestamped directories under the `pre_migration.simp` directory that correspond to runs of the migration script
  - Your old files will be in a `backup_data` directory and will be linked to a local bare Git repository in the same space

The upgrade steps will also have you install PuppetDB. PuppetDB is installed by default if you kick from the DVD.

**Security Announcements**

**CVEs Addressed**

**RPM Updates**

Numerous RPMs were updated in the creation of this release. Several were included due to our use of *repoclosure* to ensure that RPM dependencies are met when releasing a DVD.

- This version upgrades Facter to 2.4.

**Fixed Bugs**

- pupmod-aide
  - Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.
- pupmod-apache
  - Remove the apache_version fact and simply use the version controls built into the Apache configuration language.
  - Update all custom functions to properly scope definitions.
  - Ensure that mod_ldap is installed in SIMP >= 5.0.
• pupmod-simp-apache
  – Prevent apache from restarting after downloading a CRL.

• pupmod-clamav
  – Change the call to the rsyslog init script to the service command to seamlessly support both RHEL6 and RHEL7.

• pupmod-common
  – We no longer supply crontab or anacrontab in global_etcd.
  – Remove dynamic_swappiness cron job if a static value is set.
  – Ensure that the passgen() function fails on invalid scenarios. This prevents the accidental creation of empty passwords.
  – Allow the value 2 to be used for rp_filter in common::sysctl.
  – Added ability to return remote ip addrs.

• pupmod-dhcp
  – Change the call to the rsyslog init script to the service command to seamlessly support both RHEL6 and RHEL7.

• pupmod-iptables
  – Fixed a bug that would cause issues with Ruby 1.8.7.
  – Prevent IPv6 ::1 spoofed addresses by default.

• pupmod-simp-elasticsearch
  – Ensured that Elasticsearch works properly with the new version of Apache.
  – Removed our default ES tuning since the default works better for LogStash.
  – Ensure that Puppet manages the Elasticsearch logging file.

• pupmod-functions
  – Fixed sysv.rb to explicitly require puppet/util/selinux, which caused puppet describe to have errors.

• pupmod-simp-logstash
  – Fix issues with both TCPWrappers and IPTables when used with LogStash.

• pupmod-nfs
  – Updated the mountd port to be 20048 by default for SELinux issues in RHEL7.

• pupmod-ntp
  – Updated against NTP Security Vulnerabilities (Red Hat Article #1305723).
  – Ensure that restrict entries use DDQ format.

• pupmod-openldap
  – The Password Policy overlay was getting loaded into the default.ldif even if you didn’t want to use it. This has been fixed.
  – Made the password policy overlay align with the latest SIMP build of the plugin.
    * This means that you must have version simp-ppolicy-check-password-2.4.39-0 or later available to the system being configured.
- Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.
- Fixed reported bugs in `syncrepl.pp`.

**pupmod-openscap**
- Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.
- Changed default ssg base path to `/usr/share/xml/scap/ssg/content`

**pupmod-rsync**
- Fixed provider to run with `--dry-run` when puppet is run with a `--noop`.

**pupmod-ssh**
- Modernized the Ciphers, MACs, and Kex.
- Added explicit cases for FIPS and non-FIPS mode (as well as reasonable default cases for RHEL7 and below).
- Updated to use the new `augeasproviders` module dependencies.
- Added a function `ssh_format_host_entry_for_sorting()` that will properly sort SSH Host entries for inclusion with `concat`.

**pupmod-stunnel**
- Had a variable `options` in `stunnel.erb` that should have been scoped as `@options`.

**pupmod-sudosh**
- Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.

**pupmod-sysctl**
- Removed support for the old parsed-file provider and moved to using the new Augeas-based provider.

**pupmod-tftpboot**
- Purging of non-Puppet-managed items in `pxelinux.cfg` is now optional.

**pupmod-simp-tpm**
- IMA is disabled by default.

**simp-utils**
- Fixed the targets of `unpack_dvd`.

**pupmod-xinetd**
- Fixed: The default log_type should be ‘SYSLOG authpriv’ instead of ‘SYSLOG daemon info’.

**pupmod-vnc**
- Removed banners that broke some vnc clients.

**DVD**
- A default IP is no longer provided when booting from the ISO; simp config will set the network properly.
New Features

- **pupmod-augeasproviders**
  - This was updated to 2.1.3.
  - The update to 2.1.3 caused the addition of all of the pupmod-augeasproviders modules below.

- **augeasproviders_apache**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_base**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_core**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_grub**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_mounttab**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_nagios**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_pam**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_postgresql**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_puppet**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_shellvar**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_ssh**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **augeasproviders_sysctl**
  - Imported 2.1.3 to support the Augeasproviders stack.

- **pupmod-common**
  - Created parse_hosts function.

- **pupmod-richardc-datacat**
  - Incorporated the richarde/datacat module into the core for user convenience.

- **pupmod-freeradius**
  - Split the Freeradius module based on version so that it can be properly selected against the installed version of Freeradius. This may take two runs to coalesce.

- **pupmod-puppetlabs-inifile**
Updated to version 1.2.0.

- **pupmod-pki**
  - Now generate a system RSA public key against the passed private key.

- **pupmod-puppetlabs-postgresql**
  - Initial import of the Puppet Labs PostgreSQL module.
  - Modifications were made to support the SIMP concat.

- **pupmod-puppetlabs-puppetdb**
  - New import of the Puppet Labs PuppetDB module.

- **pupmod-puppetlabs-stdlib**
  - Updated to version 4.5.1.

- **pupmod-tftpboot**
  - Updated to use native packages and pull as much as possible.

- **Mcollective**
  - Mcollective is now available to be installed and used with SIMP. It uses SSL/TLS along with user certificates for proper encryption and authentication.

- **PuppetDB**
  - PuppetDB is now supported by SIMP and installed by default.

- **Puppetserver**
  - The puppet master service has been replaced by the puppetserver service. This is a major rewrite by Puppetlabs. Puppetserver scales better for larger agent deployments with a single puppet master.
  - Uses Environments by default, this allows for tools such as r10K. Production environment is a link to simp by default.

- **simp config**
  - simp config was rewritten to allow for new features and flexibilty.
  - Now provided as a Ruby gem “simp-cli”.

- **pupmod-simp-logstash**
  - Integrated SIMP and Electrical Logstash modules.
  - Changes the existing Logstash module to allow users to apply default SIMP filters.

- **simp-rsync**
  - Content has been restructured to eliminate licensing conflicts.
  - ClamAV has been refactored into a separate (GPL) package.

- **pupmod-simp-rsyslog**
  - Module has been rewritten to support rsyslog 7.4.

- **pupmod-simp-kibana**
  - Add Kibana dashboards to the Kibana module.
  - Allows users to apply default SIMP kibana Dashboards.

- **Facter 2.4**
– Facter now returns the following facts as their actual boolean or integer values, instead of converting them into strings:

activeprocessorcount  is_virtual  mtu_<INTERFACE>  physicalprocessorcount  processorcount  selinux_enforced  selinux  sp_number_processors  sp_packages

**Known Bugs**

- Setting pwdReset to ‘true’ in LDAP does not force a user to reset their password like it is supposed to. This works with FreeIPA and we are looking to move to support that system in the future.
- SSSD is currently broken and will allow logins via SSH even if your password has expired. This has been noted by Red Hat and is in the pipeline. Their suggestion it to move to FreeIPA from OpenLDAP. We are looking to do this in the future.
- If you are running libvirtd, when svckill runs it will always attempt to kill dnsmasq unless you are deliberately trying to run the dnsmasq service. This does not actually kill the service but is, instead, an error of the startup script and causes no damage to your system.

**1.8 Glossary of Terms**

- **Advanced Intrusion Detection Environment (AIDE)**  An intrusion detection system for checking the integrity of files under Linux. AIDE can be used to help track file integrity by comparing a snapshot of the system’s files prior to and after a suspected incident. It is maintained by Rami Lehti and Pablo Virolainen.
- **Central Processing Unit (CPU)**  The hardware within a computer system that carries out the instructions of a computer program by performing the basic arithmetical, logical, and input/output operations of the system.
- **Community Enterprise Operating System (CentOS)**  An Enterprise-grade Operating System that is directly compatible with a prominent US Linux distribution.
- **Domain Name System (DNS)**  A database system that translates a computer’s fully qualified domain name into an IP address and the reverse.
- **Dynamic Host Configuration Protocol (DHCP)**  A network protocol that enables a server to assign an IP address to a computer automatically.
- **Fully Qualified Domain Name (FQDN)**  A domain name that specifies its exact location in the tree hierarchy of the Domain Name System (DNS). It specifies all domain levels, including the top-level domain and the root zone. An FQDN is distinguished by its unambiguity; it can only be interpreted one way.
- **Hard Disk Drive (HDD)**  A device for storing and retrieving digital information, primarily computer data.
- **Internet Protocol Tables (IPtables)**  A user space application that provides an interface to the IPv4 firewall rules on modern Linux systems.
- **Internet Protocol 6 Tables (IP6tables)**  A user space application that provides an interface to the IPv6 firewall rules on modern Linux systems.
- **Lightweight Directory Access Protocol (LDAP)**  A protocol for querying and modifying LDAP directory services including information such as names, addresses, email, phone numbers, and other information from an online directory.
- **Pluggable Authentication Modules (PAM)**  A mechanism to integrate multiple low-level authentication schemes into a high-level application programming interface (API). It allows programs that rely on authentication to be written independent of the underlying authentication scheme.
Preboot Execution Environment (PXE) An environment to boot computers using a network interface independently of data storage devices (like hard disks) or installed operating systems.

Privacy Enhanced Mail (PEM) An early standard for securing electronic mail. This is the public-key of a specific certificate. This is also the format used for Certificate Authority certificates.

Public Key Infrastructure (PKI) A security architecture that has been introduced to provide an increased level of confidence for exchanging information over an increasingly insecure Internet. PKI enables users of a basically insecure public networks, such as the Internet, to securely authenticate to systems and exchange data. The exchange of data is done by using a combination of cryptographically bound public and private keys.

Puppet An Open Source configuration management tool written and maintained by Puppet Labs. Written as a Ruby DSL, Puppet provides a declarative language that allows system administrators to provide a consistently applied management infrastructure. Users describes system resource and resource state in the Puppet language. Puppet discovers system specific information via facter and compiles Puppet manifests into a system specific catalog containing resources and resource dependencies, which are applied to each client system.

Random Access Memory (RAM) A form of computer data storage. A random access device allows stored data to be accessed in nearly the same amount of time for any storage location, so data can be accessed quickly in any random order.

Red Hat A collection of many different software programs, developed by Red Hat®, Inc. and other members of the Open Source community. All software programs included in Red Hat®Linux® are GPG signed by Red Hat to indicate that Red Hat supplied them.

See also RHEL.

Red Hat Enterprise Linux (RHEL) A commercial Linux operating system produced by Red Hat®, Inc. RHEL is designed to provide an Enterprise-ready Linux distribution suitable to multiple target applications.

See also Red Hat Linux.

Ruby A dynamic, reflective, general-purpose object-oriented programming language that combines syntax inspired by Perl with Smalltalk-like features. Ruby originated in Japan during the mid-1990s and was first developed and designed by Yukihiro “Matz” Matsumoto. It was influenced primarily by Perl, Smalltalk, Eiffel, and Lisp. Ruby supports multiple programming paradigms, including functional, object oriented, imperative and reflective. It also has a dynamic type system and automatic memory management; it is therefore similar in varying respects to Smalltalk, Python, Perl, Lisp, Dylan, Pike, and CLU.

Secure Shell (SSH) An application for secure data communication, remote shell services, or command execution between networked computers. SSH utilizes a server/client model for point-to-point secure communication.

Secure Sockets Layer (SSL) The standard security technology for using PKI keys to provide a secure channel between two servers.

See also TLS.

SIMP A security framework that sits on top of RHEL or CentOS.

Transport Layer Security (TLS) A cryptographic protocol that provides network communications security. TLS and SSL encrypt the segments of network connections above the Transport Layer, using asymmetric cryptography for privacy and a keyed message authentication codes for message reliability.

See also SSL.

Trivial File Transfer Protocol (TFTP) A file transfer protocol generally used for automated transfer of configuration or boot files between machines in a local environment.

Virtual Machine (VM) A completely isolated guest operating system installation within a normal host operating system.

Yellowdog Updater, Modified (YUM) A software installation tool for Linux. It is a complete software management system that works with RPM files. YUM is designed to be used over a network or the Internet.

1.8. Glossary of Terms

23
1.9 Appendix A - Sample Puppet Files

This is a sample reverse (e.g. 0.0.10.db) file

```
$TTL 86400
@ IN SOA simp.net. sample.simp.net. (2015061101
   1h ; refresh
   15m ; retry
   1w ; expiry
   1d ) ; minimum
IN NS sample.simp.net.
15 IN PTR sample.simp.net.
```

This is a sample forward (e.g. your.domain) file

```
$TTL 86400
simp.net. IN SOA sample.simp.net. hostmaster.simp.net. (2015061201
   3H ; refresh
   15M ; retry
   1W ; expiry
   1D ) ; minimum
IN NS sample.simp.net.
sample IN A 192.168.122.15
```

This is a sample (e.g. dhcpd.conf) dhcpd file

```
allow booting;
allow bootp;
ddns-update-style interim;

class "pxeclients" {
   match if substring(option vendor-class-identifier, 0, 9) = "PXEClient";
   next-server 192.168.122.15;
   filename "linux-install/pxelinux.0";
}

subnet 192.168.122.0 netmask 255.255.255.0 {
    option routers 192.168.122.1;
    option subnet-mask 255.255.255.0;

    option domain-name "simp.net";
    option domain-name-servers 192.168.122.15;

    option time-offset -0;
    default-lease-time 21600;
    max-lease-time 43200;

    # We explicitly list our hosts to restrict the hosts that can access our
    # network.
    # host sample {
    #    hardware ethernet DE:AD:BE:EF:00:00;
    #    fixed-address 192.168.122.16;
    # }
}
```
This is a sample named file

```plaintext
#FIX_YOUR_NAMED_CONFIG_PRIOR_TO_RUNNING

acl trusted_hosts {
  127.0.0.1;
};

options {
  version " ";
  query-source port 53;
  query-source-v6 port 53;
  directory "/var/named";
  dump-file "data/cache_dump.db";
  statistics-file "data/named_stats.txt";
  memstatistics-file "data/named_mem_stats.txt";
  allow-query { 192.168.122.0/24; };
  allow-recursion { 192.168.122.0/24; };
  allow-transfer { "none"; };
};

logging {
  channel default_syslog {
    syslog daemon;
    print-category yes;
    severity info;
  };
  channel secure_syslog {
    syslog local6;
    print-category yes;
    severity notice;
  };
  category "default" {
    "default_syslog";
  };
  category "xfer-out" {
    "secure_syslog";
  };
  category "xfer-in" {
    "secure_syslog";
  };
  category "dnssec" {
    "secure_syslog";
  };
  category "update" {
    "secure_syslog";
  };
  category "notify" {
    "secure_syslog";
  }
```
This is a sample zones (e.g. your.domain) file
This is a sample kickstart (e.g. pupclient_x86_64.cfg) file

```sh
# Replace the following strings in this file
# #BOOTPASS# - Your MD5 hashed bootloader password = (your encrypted PW)
# #ROOTPASS# - Your MD5 hashed root password = (your encrypted PW i.e.$6$sk6ngonx...YFLEkw816lLfyMM4eh5M2mVtDMfMbOX/
# #KSSERVER# - The IP address of your YUM server = 192.168.122.15
# #YUMSERVER# - The IP address of your YUM server = 192.168.122.15
# #LINUXDIST# - The LINUX Distribution you are kickstarting
#  - Current CASE SENSITIVE options: RedHat CentOS

authconfig --enableshadow --passalgo=sha512 --enablemd5
bootloader --location=mbr --append="console=ttyS1,57600 console=tty1" --iscrypted --password=$6$sk6ngonx$7c0eXVPVg0E7TqB1l.U0L8dugFIH89fWwppITz3Yu2QBkzK2ak69seYFLEkw816lLfyMM4eh5M2mVtDMfMbOX/
rootpw --iscrypted $6$sk6ngonx$7c0eXVPVg0E7TqB1l.U0L8dugFIH89fWwppITz3Yu2QBkzK2ak69seYFLEkw816lLfyMM4eh5M2mVtDMfMbOX/
zerombr
key --skip
firewall --enabled --ssh
firstboot --disable
logging --level=info
network --bootproto=dhcp
reboot
selinux --permissive
timezone --utc GMT

install
skipx

%include /tmp/repo-include

text
keyboard us
lang en_US
url --url http://192.168.122.15/yum/CentOS/6/x86_64

%include /tmp/part-include

%packages --nobase
redhat-lsb
xorg-x11-xauth
acl
rpm
yum
bzip2
crontabs
libutempter
pciutils
psacct
quota
tmpwatch
tmpwatch
vixie-cron
```
amtu
anacron
coolkey
cpuspeed
cryptsetup-luks
dhclient
gnupg
irqbalance
krb5-workstation
libaio
logwatch
logrotate
mdadm
microcode_ctl
nss_db
openssh-clients
openssh-server
pam_krb5
pam_passwdqc
pam_pkcs11
ntp
readahead
smartmontools
stunnel
sudo
tcp_wrappers
unzip
usbutils
vconfig
wget
which
zip
aide
iptables
iptables-ipv6
netlabel_tools
audit
lsscsi
net-snmp
sysstat
vlock
mutt
subversion
-sysklogd
rsyslog
lsof
vim-enhanced
-sendmail
dracut-fips
dracut-kernel
# Puppet stuff
puppet
facter
rsync
ruby-rdoc
# in case of broken repo, these should be installed.
hdparm
kbd
%pre
ksserver="192.168.122.15"
wget -O /tmp/diskdetect.sh http://$ksserver/ks/diskdetect.sh;
chmod 750 /tmp/diskdetect.sh;
/tmp/diskdetect.sh;
wget -O /tmp/repodetect.sh http://$ksserver/ks/repodetect.sh;
chmod 750 /tmp/repodetect.sh;
/tmp/repodetect.sh '6' $ksserver;
%end

%post
ostype="CentOS"
if [ $ostype == "CentOS" ]; then
  sed -i '/enabled=/d' /etc/yum.repos.d/CentOS-Base.repo;
  sed -i '^[\[.*\]]/ a\'
  enabled=0' /etc/yum.repos.d/CentOS-Base.repo;
fi

ksserver="192.168.122.15"

# Add boot target to GRUB kernel command line
grubby --update-kernel=`grubby --default-kernel` \
--args="boot=`blkid \"/boot\" /etc/mtab | cut -f1 -d' ' | cut -f2 -d' ' | tr -d '\"'"

# Turn off prelinking and remove all previous
sed -i '/PRELINKING=yes/ c\PRELINKING=no' /etc/sysconfig/prelink
prelink -u -a

# Enable the firstboot bootstrapping script.
wget --no-check-certificate -O /etc/init.d/runpuppet http://$ksserver/ks/runpuppet;
chmod 700 /etc/rc.d/init.d/runpuppet;
chkconfig --add runpuppet;
chkconfig --level 35 runpuppet on;
%end
Chapter 2

SIMP User Guide

Contents:

2.1 Introduction

This guide will walk a user through the process of managing a SIMP system. This system includes, at a minimum, a SIMP server with properly configured networking information and a working Puppet server. Additionally, this document outlines the process of managing clients and users associated with the SIMP system.

2.1.1 Level of Knowledge

SIMP is designed for use by system administrators or users with a strong background using Linux operating systems. The core applications that make up SIMP and require prerequisite knowledge are:

- **Puppet** - 3.7 or later
- **Domain Name System (DNS)** - BIND 9
- **Dynamic Host Configuration Protocol (DHCP)** - Internet Systems Consortium (ISC) DHCP
- **Lightweight Directory Access Protocol (LDAP)** - OpenLDAP
- **RedHat Kickstart (including all tools behind it)** - *Trivial File Transfer Protocol (TFTP)*, PXELinux, etc.
- **Apache**
- **Yellowdog Updater, Modified (YUM)**
- **Rsyslog** Version 3+
- **Internet Protocol Tables (IPtables)** (Basic knowledge of the rules)
- **Auditd** (Basic knowledge of the rules)
- **Advanced Intrusion Detection Environment (AIDE)** (Basic knowledge of the rules)
- **Basic X.509 Key Management**

By itself, SIMP does as much initial setup and configuration of these tools as possible. However, without at least some understanding, a user will be unable to tailor a SIMP system to fit the desired environment. A general understanding of how to control and manipulate these tools from the command line interface (CLI) will be necessary, as SIMP does not come stock with a graphical user interface (GUI). Additionally, knowledge of scripting and Ruby programming will also help to further customize a SIMP install. For example, in order to use the advanced features of Puppet, some Ruby programming is required.
2.1.2 SIMP Defined

SIMP is a continually managed minimal Linux framework compatible with both Red Hat Enterprise Linux (RHEL) and Community Enterprise Operating System (CentOS). By maintaining and managing file-level and network configuration consistency, SIMP addresses process degradation on an operating system level. SIMP uses Puppet to provide multi-system consistency over time while augmenting the software with tools like Capistrano for controlled application of one-time mass operations.

2.2 User Management

The chapter explains how to manage users in the default SIMP environment.

2.2.1 Managing Users with Lightweight Directory Access Protocol (LDAP)

SIMP natively uses OpenLDAP for user and group management. Actionable copies of the LDAP Data Interchange Format (.ldif) files can be found on the system in the /usr/share/doc/simp-<Version>/ldifs directory.

Users cannot have any extraneous spaces in .ldif files.

Type `:set list` in vim to see hidden spaces at the end of lines.

Type `sed -i 's/\(^[[:graph:]]\)*:/:/\1\2/' file.ldif` to strip out inappropriate characters.

Note

Use the [ and ] characters to scroll right when using ELInks.

2.2.2 Add Users

Users can be added with or without a password. Follow the instructions in the following sections.

Warning

This process should not be used to create users or groups for daemon processes unless the user has experience.

Adding Users With a Password

To add a user to the system, Secure Shell (SSH) to the LDAP server and use the slappasswd command to generate a password hash for a user.

Create a /root/ldifs directory and add the following information to the /root/ldifs/adduser.ldif file. Replace the information within < > with the installed system’s information.

Example ldif to add a user

dn: uid=<User UID>,ou=People,dc=your,dc=domain
uid: <User UID>
cn: <User UID>
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
objectClass: ldapPublicKey
shadowMax: 90
shadowMin: 1
shadowWarning: 7
shadowLastChange: 10167
pwdReset: TRUE
sshPublicKey: <User SSH Public Key>
loginShell: /bin/bash
uidNumber: <User UID Number>
gidNumber: <User Primary GID>
homeDirectory: /home/<User UID>
userPassword: <Password Hash from slappasswd>

Type `ldapadd -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain" -f /root/ldifs/adduser.ldif` .

Ensure that an administrative account is created as soon as the SIMP system has been properly configured. Administrative accounts should belong to the administrators LDAP group (gidNumber 700). Members of this LDAP group can utilize sudo sudo for privilege escalation.

Note

The \texttt{pwdReset: TRUE} command causes the user to change the assigned password at the next login. This command is useful to pre-generate the password first and change it at a later time.

This command appears to be broken in some versions of \texttt{nss_ldap}. Therefore, to avoid future issues set \texttt{shadowLastChange} to a value around 10000.

Adding Users Without a Password

Create a \texttt{/root/ldifs} directory and add the following information to the \texttt{/root/ldifs/adduser.ldif} file. Replace the information within \texttt{<>} with the installed system’s information.

Example ldif example to add a user

```
dn: uid=<User UID>,ou=People,dc=your,dc=domain
uid: <User UID>
cn: <User UID>
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
objectClass: ldapPublicKey
sshPublicKey: <User SSH Public Key>
loginShell: /bin/bash
uidNumber: <User UID Number>
gidNumber: <User Primary GID>
homeDirectory: /home/<User UID>
```

Type `\texttt{ldapadd -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain" -f /root/ldifs/adduser.ldif}` .

Wait for the \texttt{akeys.pl} command to run (hourly) prior to the user being able to log in with the assigned keys.
### 2.2.3 Remove Users

To remove a user, create a `/root/ldifs/removeuser.ldif` file. Add the information below to the file and replace the text within `< >` with the installed system’s information.

**Example ldif to remove a user**

```plaintext
dn: cn=<User UID>,ou=Group,dc=example,dc=domain
changeType: delete

dn: uid=<User UID>,ou=People,dc=example,dc=domain
changeType: delete
```

Type `**ldapmodify -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain" -f /root/ldifs/removeuser.ldif**`.

### 2.2.4 Additional .ldif File Commands

Other useful commands for .ldif files can be found below. Before using these commands, ensure that the `/root/ldifs` directory has been created.

**Changing a Password**

To change a password, add the following information to the `/root/ldifs/<.ldif File>` file. Replace the information below within `< >` with the installed system’s information.

**Example ldif to change password**

```plaintext
dn: uid=<User UID>,ou=People,dc=your,dc=domain
changetype: modify
replace: userPassword
userPassword: <Hash from slappasswd>
```

Type `**ldapmodify -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain" -f <.ldif_file>**`.

**Adding a Group**

To add a group, add the following information to the `/root/ldifs/<.ldif File>` file. Replace the information below within `< >` with the installed system’s information.

**Example ldif to add a group**

```plaintext
dn: cn=<Group Name>,ou=Group,dc=your,dc=domain
objectClass: posixGroup
objectClass: top
cn: <Group Name>
gidNumber: <GID>
description: "Some Descriptive Text"
```

Type `**ldapadd -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain" -f <.ldif_file>**`.
Removing a Group

To remove a group, add the following information to the `/root/ldifs/<.ldif File>` file. Replace the information below within <> with the installed system’s information.

Example ldif to remove a group

```
dn: cn=<Group Name>,ou=Group,dc=your,dc=domain
changeType: delete
```

Type `ldapmodify -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain" -f <.ldif_file>`.

Adding Users to a Group

To add users to a group, add the following information to the `/root/ldifs/<.ldif File>` file. Replace the information below within <> with the installed system’s information.

Example ldif to add to a group

```
dn: cn=<Group Name>,ou=Group,dc=your,dc=domain
changeType: modify
add: memberUid
memberUid: <UID1>
memberUid: <UID2>
...
memberUid: <UIDX>
```

Type `ldapmodify -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain" -f <.ldif_file>`.

Removing Users from a Group

To remove users from a group, add the following information to the `/root/ldifs/<.ldif File>` file. Replace the information below within <> with the installed system’s information.

Example ldif to remove a user from a group

```
dn: cn=<Group Name>,ou=Group,dc=your,dc=domain
changeType: modify
delete: memberUid
memberUid: <UID1>
memberUid: <UID2>
...
memberUid: <UIDX>
```

Type `ldapmodify -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain" -f <.ldif_file>`.

Updating an SSH Public Key

To update an SSH public key, add the following information to the `/root/ldifs/<.ldif File>` file. Replace the information below within <> with the installed system’s information.

Example ldif to update SSH public key

```
```
Forcing a Password Reset

To force a password reset, add the following information to the /root/ldifs/<.ldif File> file. Replace the information below within <> with the installed system’s information.

Example ldif to reset user’s shadowLastChange

```
  dn: uid=<User UID>,ou=People,dc=your,dc=domain
  changetype: modify
  replace: pwdReset
  pwdReset: TRUE
  replace: shadowLastChange
  shadowLastChange: 10000
```

```
  Type `ldapmodify -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain"
  -f <.ldif File>` .
```

Note

The ldapmodify command is only effective when using the ppolicy overlay. In addition, the user’s shadowLastChange must be changed to a value prior to the expiration date to force a Pluggable Authentication Modules (PAM) reset.

Unlocking an LDAP Account

To unlock an LDAP account, add the following information to the /root/ldifs/<.ldif File> file. Replace the information below within <> with the installed system’s information.

Example ldif to Unlock LDAP Account

```
  dn: uid=<User UID>,ou=People,dc=your,dc=domain
  changetype: modify
  delete: pwdAccountLockedTime
```

```
  Type `ldapmodify -Z -x -W -D "cn=LDAPAdmin,ou=People,dc=your,dc=domain"
  -f <.ldif File>` .
```

Note

The `ldapmodify` command is only effective when using the `ppolicy` overlay.

2.2.5 Troubleshooting Issues

If a user’s password is changed in LDAP or the user changes it shortly after its initial setup, the “Password too young to change” error may appear. In this situation, apply the `pwdReset:TRUE` command to the user’s account as described in the Add Users with a Password section.
2.3 Client Management

This chapter provides guidance to install and configure SIMP clients based on the standard SIMP system installed using the SIMP DVD.

2.3.1 System Requirements

Before installing clients, the system should consist of the following minimum requirements:

- **Hardware/Virtual Machine (VM)**: Capable of running RHEL 6 or 7; 64-bit compatible
- **RAM**: 512 MB
- **HDD**: 5 GB

2.3.2 Configuring the Puppet Master

Perform the following actions as *root* on the Puppet Master system prior to attempting to install a client.

2.3.3 Configure DHCP

The table below lists the steps to configure DHCP.

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Log on as <em>root</em>.</td>
</tr>
<tr>
<td>2.</td>
<td>Open the <code>/var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/dhcpd/dhcpd.conf</code> file and edit it to suit the necessary environment. <strong>NOTE</strong>: Enter the hardware ethernet and fixed-address for each client that will be kickstarted. An example dhcpd.conf is listed in the Appendix, where 192.168.1.100 is the client targeted for PXE/provisioning. Be sure to substitute in the actual values appropriate to your environment, including the ethernet address of the target client.</td>
</tr>
<tr>
<td>3.</td>
<td>Save and close the file.</td>
</tr>
<tr>
<td>4.</td>
<td>Type <code>puppet agent -t --tags dhcpd</code> on the Puppet Master to apply the changes.</td>
</tr>
</tbody>
</table>

Table: DHCP Configuration Procedure

2.3.4 Configure DNS

Most static files are pulled over `rsync` by Puppet in this implementation for network efficiency. Specific directories of interest are noted in this section.

It is possible to use an existing DNS setup; however, the following table lists the steps for a local setup.

Table: DNS Configuration Procedure
2.3.5 Configure the Kickstart

The table below lists the steps to configure the kickstart.

Table: Kickstart Configuration Procedure

The diskdetect.sh* script is responsible for detecting the first active disk and applying a disk configuration. Edit this file to meet any necessary requirements or use this file as a starting point for further work.

2.3.6 Setting Up the Client

The table below lists the steps to PXE boot the system and set up the client.

Table: PXE Boot Procedure

Upon successful deployment of a new CentOS or RHEL client, it is highly recommended that LDAP administrative accounts be created. See Chapter 2 of the SIMP Users Guide for user management.

2.3.7 Troubleshooting Issues

If the client has been kickstarted, but is not communicating with the Puppet server, try the following options:

- Check the forward and reverse DNS entries on the client and server; both must be correct.
- Check the time on the systems. More than an hour’s difference will cause serious issues with certificates.
- Remove /var/lib/puppet/ssl on the client system; run puppet cert --clean <Client Host Name> on the Puppet server; and try again.

2.3.8 Troubleshoot Certificate Issues

If host certificates do not appear to be working and the banner is not getting rsync’d to the clients, ensure that all certificates verify against the installed CA certificates.

The table below lists the steps to determine which certificates are working and which are not.

Table: Certificate Verification Procedure

If the TXT_DB error number 2 appears, revoke the certificate that is being regenerated. The table below lists the steps to revoke the certificate.

Table: Revoke Certificate Procedure

2.4 Configure the PXE Boot

In order to Preboot Execution Environment (PXE) boot clients, a copy of the ISOs for all versions of RHEL being kickstarted is required.

2.4.1 Setting Up the Kickstart

The system follows the standard kickstart model. Kickstart files are placed in the /var/www/ks; directory. Custom packages are placed in an appropriate repository created under the /var/www/yum directory.

Once the model is ready, the default SIMP settings provide access to the user’s trusted subnets as defined in the only:: not simp_4
Note

Sample kickstart templates have been provided in the ks directory on the SIMP DVD at the root level.

2.4.2 Setting up TFTP

This section describes the process of setting up static files and manifests for TFTP.

Static Files

Type cd /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot and then type ls to check for the existence of the /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/rhel<Version>-<Architecture> directory.

If the directory does not exist, create one in that location and add the vmlinuz and initrd.img files from the images/pxeboot directory of the SIMP DVD. An example is provided below for setting up the CentOS RHEL_MAJOR_MINOR_VERSION distribution.

```bash
cd /var/simp/rsync/CentOS/RHEL\_MAJOR\_MINOR\_VERSION/tftpboot/linux-install

mkdir centosRHEL\_MAJOR\_MINOR\_VERSION\_x86\_64; cd centosRHEL\_MAJOR\_MINOR\_VERSION\_x86\_64

cp -p /var/www/yum/CentOS/RHEL\_MAJOR\_MINOR\_VERSION/x86\_64/images/pxeboot/* .

cd ..

chmod 640 centosRHEL\_MAJOR\_MINOR\_VERSION\_x86\_64; chown root:nobody centosRHEL\_MAJOR\_MINOR\_VERSION\_x86\_64

unlink centosRHEL\_MAJOR\_VERSION\_x86\_64

ln -s centosRHEL\_MAJOR\_MINOR\_VERSION\_x86\_64 centosRHEL\_MAJOR\_VERSION\_x86\_64
```

Manifest

Assuming that the Puppet server is being used, create and add the following example code to a site manifest, /etc/puppet/environment/simp/modules/site/manifests/tftpboot.pp. Keep in mind that the code varies based on the model being kickstarted.

Source Code for Setting Up TFTP on Puppet Server TFTP Examples

```bash
# Set KSSERVER statically or use Hiera for lookup
class site::tftpboot { 
  include 'tftpboot'

  tftpboot::linux_model { 'CentOS_RHEL_MAJOR_VERSION': 
    kernel => 'centosRHEL_MAJOR_VERSION_x86_64/vmlinuz',
  }
}
```

2.4. Configure the PXE Boot
Next, add the tftpboot site manifest to your puppet server node via Hiera. If it does not already exist, create
/etc/puppet/hieradata/hosts/your.server.fqdn.yaml. Add the following example code to that yaml file.

Source Adding TFTP Site Manifest to Hiera TFTP Examples

```yaml
---
classes:
  - 'site::tftpboot'
```

After updating the above file, type `puppet agent -t --tags tftpboot` on the Puppet server.

### 2.5 Maximum Number of Nodes

The maximum number of clients reasonable per each system is dependent on many variables, including number of processors and size of memory. Although it is impossible to predict exactly how many clients a specific server may be able to handle, a simple algorithm can give the user an estimate.

Servers with different hardware have been tested at worst case scenario. This means that all of the server’s clients will run Puppet at the exact same time. The most important information collected during these runs was the compile time, which shows the increase in seconds that it takes for each node to compile when another node is added. After a certain number of nodes, nodes begin to drop to compile times lower than 30 seconds. These nodes are not actually completing their Puppet runs. This data can be seen in the following graph:

#### 2.5.1 Number of Nodes vs. Compile Time

The queue size can be found by looking at the maximum number of clients running Puppet at once before any are dropped. According to the SIMP team’s data, a server with two cores has a queue size of four; a server with three cores has a queue size of six; however, a server with four cores has a queue size of six. Although it may appear that the queue size is plateauing as cores are increased, the SIMP team predicts that this is due to the limited memory. However, the team is confident that a system with four cores and 4GB of ram will indeed have a queue size of eight clients. From this, it can be concluded that, given enough memory, \(\text{Queue Size} = 2 \times \text{Cores}\).

Also using this data, the compile times for other systems can be predicted given the amount of processors, memory, and nodes. This is done using ordinary least squares in Octave.

In addition, the maximum number of clients can also be predicted with the use of the following equation:

\[
\text{Max_Num_Of_Total_Clients} = \left(\frac{\text{Run Time In Sec}}{\text{Comp Time}}\right) \times \text{Queue Size}
\]

Where \(\text{Run Time In Sec}\) is the number of seconds per half an hour (1800), \(\text{Queue Size}\) is the maximum number of clients in the worst case scenario (queue size), and \(\text{Comp Time}\) is the average compile time of the clients when there are \(\text{Max_Num_Worstcase}\) clients.
2.6 Apply Certificates

This section provides guidance on obtaining official certificates and generating a Fake CA.

2.6.1 Obtaining Official Certificates

All SIMP systems must have Public Key Infrastructure (PKI) keypairs generated for the server. These keys reside in the /etc/puppet/keydist directory and are served to the clients over the Puppet protocol.

Note

These keypairs are not the keys that the Puppet server uses for its operation. Do not get the two confused.

The table below lists the steps to add any keys for the server that were received from a proper CA to /etc/puppet/keydist.
### 2.6.2 Generating Fake CAs

If server certificates have not or could not be obtained at the time of client installation, the SIMP team provides a way to create them for the system so that it will work until proper certificates are provided.

**Note**

This option should not be used for any operational system that can use proper enterprise PKI certificates.

The table below lists the steps to generate the Fake CAs.

#### Table: Generating Fake CAs Procedure

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Type <code>cd /etc/puppet/keydist</code></td>
</tr>
<tr>
<td>2.</td>
<td>Type <code>mkdir cacerts</code> and copy the root CA public certificates into <code>cacerts</code> in <em>Privacy Enhanced Mail (PEM)</em> format (one per file).</td>
</tr>
<tr>
<td>3.</td>
<td>Type <code>cd cacerts</code></td>
</tr>
<tr>
<td>4.</td>
<td>Type <code>for file in *.pem; do ln -s $file </code>openssl x509 -in $file -hash -noout<code>.0; done</code></td>
</tr>
</tbody>
</table>

#### Warning

If the `clean.sh` command is run after the certificates have been generated, the running system will break. To troubleshoot certificate problems, see the section at the end of this chapter.

If issues arise while generating keys, type `cd /etc/puppet/Config/FakeCA` to navigate to the `/etc/puppet/Config/FakeCA` directory, then type `/clean.sh` to start over.

After running the `clean.sh` script, type `/gencerts_nopass.sh` to run the script again using the previous procedure table.

### 2.7 SIMP Administration

This chapter provides basic guidance on how to administer a SIMP environment.
Warning

While working with the system, keep in mind that Puppet does not work well with capital letters in host names. Therefore, they should not be used.

2.7.1 Nightly Updates

All SIMP systems are configured, by default, to do a YUM update of the entire system on a nightly basis.

The configuration pulls updates from all repositories that the system is aware of. To change this behavior, refer to the Excluding Repositories FAQ section. This configuration is also helpful because it is easier to manage symlinks in YUM repositories than it is to manage individual package minutia for every single package on every system.

The general technique is to put packages that all systems will receive into the Updates repository provided with SIMP. Any packages that will only go to specific system sets will then be placed into adjunct repositories under /var/www/yum and the user will point specific systems at those repositories using the yumrepo Puppet type. Any common packages can be symlinked or hard linked between repositories for maximum space utilization.

2.7.2 Sudosh

By default, a SIMP system uses Sudosh to enable logging of sudo sessions to Rsyslog. To open a sudo session as root (or any other user), type su - as simp, or sudo sudosh as anyone else, instead of sudo su.

The logs are stored in /var/log/sudosh.log. Sessions can be replayed by typing sudosh-syslog-replay.

2.7.3 User Accounts

By default, users can add local users to a system or use LDAP to administer users.

It is recommended that LDAP is used for adding all regular users so that there is no conflict with multiple system updates and synchronization. For more information on managing LDAP users, refer to the User Management chapter.

It is also possible that there will be users that are local to the system. To have these users follow the normal password expiration conventions set on the system, use the native Puppet user and group types.

To have a user that does not expire, look at the /etc/puppet/localusers file to enable these users across the systems. The comments in the file provide instructions on generating entries for the desired systems. It is hoped that future versions of Puppet will support the modification of password expiration values via the native types and that the localusers file will be retired.

2.7.4 Certificate Management

This section describes the two different types of certificates used in a SIMP system and how to manage them. For information on initial certificate setup, refer to the Apply Certificates section of the Client Management chapter.

2.7.5 Server Certificates

Server certificates are the standard PKI certificates assigned either by an official CA or generated using the FakeCA utility offered by SIMP. They can be found in the /etc/pki/ directory of both the client and server systems. These certificates are set to expire annually. To change this, edit the following files with the number of days for the desired lifespan of the certificates:
Note

This assumes that the user has generated Certificates with the FakeCA provided by SIMP. If official certificates are being used, these settings must be changed within the official CA, not on the SIMP system.

- `/etc/puppet/Config/FakeCA/CA`
- `/etc/puppet/Config/FakeCA/ca.cnf`
- `/etc/puppet/Config/FakeCA/default_altnames.cnf`
- `/etc/puppet/Config/FakeCA/default.cnf`
- `/etc/puppet/Config/FakeCA/user.cnf`

In addition, any certificates that have already been created and signed will have a config file containing all of its details in `/etc/puppet/Config/FakeCA/output/conf/`.

Important

Editing any entries in the above mentioned config files will not affect the existing certificates. To make changes to an existing certificate it must be re-created and signed.

Below is an example of how to change the expiration time from one year (the default) to five years for any newly created certificate.

```
for file in $(grep -rl 365 /etc/puppet/Config/FakeCA/)
do
  sed -i 's/365/1825/' $file
done
```

2.7.6 Puppet Certificates

Puppet certificates are issued and maintained strictly within Puppet. They are different from the server certificates and should be managed with the `puppet cert` tool. For the complete documentation on the `puppet cert` tool, visit the Puppet Labs cert manual detailing its capabilities. On a SIMP system, these certificates are located in the `/var/lib/puppet/ssl/` directory and are set to expire every five years.

2.7.7 Applications

This section describes how to add services to the servers. To perform this action, it is important to understand how to use IPtables and what the `svckill.rb` script does on the system.

2.7.8 IPTables

By default, the SIMP system locks down all incoming connections to the server save port 22. Port 22 is allowed from all external sources since it is expected that the user will want to be able to SSH into the systems from the outside at all times.

The default alteration for the IPtables start-up script is such that it will “fail safe”. This means that if the IPtables rules are incorrect, the system will not open up the IPtables rule set completely. Instead, the system will deny access to all ports except port 22 to allow for recovery via SSH.

There are many examples of how to use the IPtables module in the source code; the Apache module at `/etc/puppet/modules/apache` is a particularly good example. In addition, look at the definitions in the IPtables module to understand their purpose and choose the best option. Refer to the IPtables page of the Developers Guide for a good summary and example code (HTML version only).
2.7.9 svckill.rb

To ensure that the system does not run more services than are required, the svckill.rb script has been implemented to stop any service that is not properly defined in the Puppet catalogue.

To prevent services from stopping, refer to the instructions in the *My Services Are Dying!* FAQ section.

2.7.10 GUI

SIMP was designed as a minimized system, but it is likely that the user will want to have a GUI on some of the systems. Refer to the *Infrastructure Setup* section for information on setting up GUIs for the systems.

2.8 Backing up the Puppet Master

This section details all of the steps required for backing up the Puppet Master.

*Note*

SIMP, by default, provides two ways to back up data. They are BackupPC and Git. If there is a different preferred method, the user may install it and configure it first.

*Warning*

BackupPC may, or may not, work properly for you on RHEL7+ systems. The SIMP team is currently evaluating other options for an inbuilt backup system.

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Backup <code>/var/lib/puppet/ssl</code></td>
</tr>
<tr>
<td>1.</td>
<td>Backup <code>/etc/puppet</code></td>
</tr>
<tr>
<td>1.</td>
<td>Backup <code>/srv/rsync</code> and/or <code>/var/simp/rsync</code></td>
</tr>
<tr>
<td>1.</td>
<td>Optional: Backup <code>/var/www</code></td>
</tr>
</tbody>
</table>

Table: SIMP Upgrade Process

2.9 Managing Workstation Infrastructures

This chapter describes how to manage client workstations with a SIMP system including GUIs, repositories, virtualization, Network File System (NFS), printing, and Virtual Network Computing (VNC).

2.9.1 Infrastructure Setup

The following sections provide examples for setting up a SIMP workstation environment.
2.9.2 User Workstation Setup

Below is an example manifest called `/etc/puppet/modules/site/manifests/workstation.pp` for setting up a user workstation.

```puppet
class site::workstation {
  include 'site::gui'
  include 'site::repos'
  include 'site::virt'
  include 'site::automount'
  include 'site::print::client'

  # Make sure everyone can log into all nodes.
  # If you want to change this, simply remove this line and add
  # individual entries to your nodes as appropriate
  pam::access::manage { "Allow Users":
      comment => 'Allow all users in the "users" group to access the system from anywhere.',
      users => '(users)',
      origins => ['ALL']
  }

  # General Use Packages
  package { 
    'pidgin',
    'git',
    'control-center-extra',
    'gconf-editor',
    'evince',
    'libreoffice-writer',
    'libreoffice-xsltfilter',
    'libreoffice-calc',
    'libreoffice- impress',
    'libreoffice-mailmerge',
    'libreoffice-base',
    'libreoffice-math',
    'libreoffice-pdfimport',
    'bluefish',
    'gnome-media',
    'pulseaudio',
    'file-roller',
    'inkscape',
    'gedit-plugins',
    'planner'
    ensure => 'latest'
  }
}
```

2.9.3 Graphical Desktop Setup

Below is an example manifest called `/etc/puppet/modules/site/manifests/gui.pp` for setting up a graphical desktop on a user workstation.

```puppet
class site::gui {
  include 'xwindows::gdm'
  include 'windowmanager::gnome'
  include 'vnc::client'
```
# Compiz Stuff

```
package { 
    'fusion-icon',
    'emerald-themes',
    'compiz-fusion-extras',
    'compiz-fusion-extras-gnome',
    'vinagre'
};
ensure => 'latest'
}
```

## 2.9.4 Workstation Repositories

Below is an example manifest called `/etc/puppet/modules/site/manifests/repos.pp` for setting up workstation repositories.

```
class site::repos {
    # Whatever local yumrepo statements you need for installing
    # your packages and keeping your systems up to date
}
```

## 2.9.5 Virtualization on User Workstations

Below is an example manifest called `/etc/puppet/modules/site/manifests/virt.pp` for allowing virtualization on a user workstation.

```
# We allow users to run VMs on their workstations.
# If you don't want this, just don't include this class.
# If this is installed, VM creation and management is still limited by PolicyKit

class site::virt {
    include 'libvirt::kvm'
    include 'libvirt::ksm'
    include 'network::redhat'

    network::redhat::add_eth { "em1":
        bridge => 'br0',
        hwaddr => $::macaddress_em1
    }

    network::redhat::add_eth { "br0":
        net_type => 'Bridge',
        hwaddr => $::macaddress_em1,
        require => Network::Redhat::Add_eth["em1"]
    }

    common::swappiness::conf { 'default':
        high_swappiness => '80',
        max_swappiness => '100'
    }

    # If 80% of memory is used, flush caches.
    exec { 'flush_cache_himem':
```
command => '/bin/echo 1 > /proc/sys/vm/drop-caches',
onlyif => inline_template("/bin/<%= memoryfree.split(/\s/)[0].to_f/memorysize.split(/\s/)[0].to_f < 0.2 ? true : false %>")
}

package { 'virt-manager': ensure => 'latest' }

2.9.6 Network File System

Below is an example manifest called /etc/puppet/modules/site/automount.pp for Network File System setup.

#If you are not using NFS, you do not need to include this.

class site::automount {
   include 'autofs'

   file { '/net':
      ensure => 'directory',
      mode => '0755'
   }

   #A global share
   Autofs::map::master { 'share':
      mount_point => '/net',
      map_name => '/etc/autofs/share.map'
   }

   #Map the share
   autofs::map::entry { 'share':
      options => '-fstype=nfs4, port=2049.soft',
      location => "{::nfs_server}:/share'
      Target => 'share'
   }
}

2.9.7 Setting up a Printer Environment

Below are example manifests for setting up a printing environment.

Setting up a Print Client

Below is an example manifest called /etc/puppet/modules/site/manifests/print/client.pp for setting up a print client.

class site::print::client inherits site::print::server {
   polkit::local_authority { 'print_support':
      identity => [unix_group:*],
      action => 'org.opensuse.cupskhelper.mechanism.*',
      section_name => 'Allow all print management permissions',
      result_any => 'yes',
      result_interactive => 'yes',
      result_active => 'yes'
   }
}

package { 'cups-pdf': ensure => 'latest' }
Setting up a Print Server

Below is an example manifest called /etc/puppet/modules/site/manifests/print/server.pp for setting up a print server.

class site::print::server {
  # Note, this is *not* set up for being a central print server.
  # You’ll need to add the appropriate IPTables rules for that to work.
  package { 'cups': ensure => 'latest' }
  service { 'cups':
    enable  => 'true',
    ensure  => 'running',
    hasrestart => 'true',
    hasstatus => 'true',
    require => Package['cups']
  }
}

2.10 VNC

Virtual Network Computing (VNC) is a tool that is used to manage desktops and workstations remotely through the standard setup or a proxy.

2.10.1 VNC Standard Setup

Note

You must have the pupmod-vnc RPM installed to use VNC on your system!

To enable remote access via VNC on the system, include vnc::server in Hiera for the node.

The default VNC setup that comes with SIMP can only be used over SSH and includes three default settings:

Table: VNC Default Settings

To connect to any of these settings, SSH into the system running the VNC server and provide a tunnel to 127.0.0.1:<VNC Port>. Refer to the SSH client’s documentation for specific instructions.

To set up additional VNC port settings, refer to the code in `/etc/puppet/modules/vnc/manifests/server.pp` for examples.

Important

Multiple users can log on to the same system at the same time with no adverse effects; however, none of these sessions are persistent.

To maintain a persistent VNC session, use the vncserver application on the remote host. Type `man vncserver` to reference the manual for additional details.
2.10.2 VNC Through a Proxy

The section describes the process to VNC through a proxy. This setup provides the user with a persistent VNC session.

**Important**

In order for this setup to work, the system must have a VNC server (`vserver.your.domain`), a VNC client (`vclnt.your.domain`), and a proxy (`proxy.your.domain`). A `vuser` account must also be set up as the account being used for the VNC. The `vuser` is a common user that has access to the server, client, and proxy.

**Modify Puppet**

If definitions for the machines involved in the VNC do not already exist in Hiera, create an `/etc/puppet/hieradata/hosts/vserv.your.domain.yaml` file. In the client hosts file, modify or create the entries shown in the examples below. These additional modules will allow `vserv` to act as a VNC server and `vclnt` to act as a client.

**VNC Server node**

```yaml
# vserv.your.domain.yaml
classes:
  - 'windowmanager::gnome'
  - 'mozilla::firefox'
  - 'vnc::server'
```

**VNC client node**

```yaml
# vclnt.your.domain.yaml
classes:
  - 'windowmanager::gnome'
  - 'mozilla::firefox'
  - 'vnc::client'
```

**Run the Server**

As `vuser` on `vserv.your.domain`, type `vncserver`.

The output should mirror the following:

```
New ‘vserv.your.domain:<Port Number> (vuser)’ desktop is vserv.your.domain:<Port Number>
Starting applications specified in /home/vuser/.vnc/xstartup Log file is /home/vuser/.vnc/vserv.your.domain:<Port Number>.log
```

**Note**

Remember the port number; it will be needed to set up an SSH tunnel.

**Set up an SSH Tunnel**

Set up a tunnel from the client (`vclnt`), through the proxy server (`proxy`), to the server (`vserv`). The table below lists the steps to set up the tunnel.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Start the VNC server.</td>
</tr>
<tr>
<td>2</td>
<td>Forward the VNC server's port to the proxy.</td>
</tr>
<tr>
<td>3</td>
<td>Forward the proxy's port to the client.</td>
</tr>
</tbody>
</table>

**Note**

The port number in `590<Port Number>` is the same port number as previously described. For example, if the `<Port Number>` was 6, then all references below to `590<Port Number>` become 5906.
Set Up Clients

On vclnt.your.domain, type **vncviewer localhost:**590***<Port Number>*** to open the Remote Desktop viewer.

Troubleshooting VNC Issues

If nothing appears in the terminal window, X may have crashed. To determine if this is the case, type **ps -ef | grep XKeepsCrashing**

If any matches result, stop the process associated with the command and try to restart **vncviewer** on vclnt.your.domain.

### 2.11 Upgrading SIMP

This chapter provides information on how to upgrade a running instance to the latest codebase.

#### 2.11.1 Pre-Upgrade Recommendations

The following process should be followed before upgrade.

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Run <code>puppet agent --disable</code> to disable puppet.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If you think you will need more than 4 hours to complete this task, also disable puppet in root’s crontab.</td>
</tr>
<tr>
<td>2.</td>
<td>You may wish to block all communications with agents while updating the server. This is not required but could spare you some headaches if something doesn’t work properly.</td>
</tr>
<tr>
<td></td>
<td>The simplest way to do this is to set the catalog retrieval capability to 127.0.0.1 in /etc/puppet/auth.conf as shown below.</td>
</tr>
</tbody>
</table>

```
path ~ ^/catalog/(\[^/]+)/$  
method find  
    # Uncomment this when complete and delete the other entries  
    #allow $1  
    allow 127.0.0.1  
```

| Table: SIMP Pre Upgrade | |

Using the syntax above, you can add fully qualified domain names, one at a time, to the ‘allow’ list and only those hosts will be able to retrieve their catalog from the running server. 127.0.0.1 serves as a placeholder so that no host can actually retrieve their catalog.
2.11.2 Migrating To Environments

SIMP 4.1 and 5.0 used the traditional, Rack-based, Puppet Master. Starting with 4.2 and 5.1, SIMP now uses the Clojure-based Puppet Server. Unfortunately, there are some conflicts with directly upgrading from the Puppet Master to the Puppet Server since some of the RPM package prerequisites conflict. This new Puppet Server can properly utilize Puppet Environments. To provide our users with this capability, and to facilitate more dynamic workflows in the future, the SIMP team has migrated all existing material to a native simp environment. To help facilitate your migration, the SIMP team has created two migration scripts that both upgrade your Puppet Server and migrate your existing data into the new simp environment.

**Warning**

You must have at least **2.2G of free memory** to run the new Puppet Server.

2.11.3 Migration Script Features

The migration script will perform the following actions on your system:

- Remove the `puppet-server` package from your system
- Install the `puppetserver` package onto your system
- Update all packages from your repositories
- Create a backup folder at `/etc/puppet/environments/pre_migration.simp`
- Create a Git repository in the backup folder under a timestamped directory
- Commit all current materials from `/etc/puppet` into the backup Git repository
- Checkout the backup Git repository under the timestamped directory as `backup_data` for ease of use
- Migrate all existing data into the new simp environment under `/etc/puppet/environments/simp`

**Note**

All future upgrades will only affect the new simp environment. You may create new environments and/or modify the contents of `/etc/puppet/modules` without fear of the SIMP packages overwriting your work.

2.11.4 Migration Script Execution

Table: Executing the Migration Script

Your new Puppet Server should now be running and a run of `puppet agent -t` should complete as usual.

2.11.5 Converting from Extdata to Hiera

SIMP now uses Hiera natively instead of Extdata. Tools have been put into place by Puppet Labs and SIMP to make the conversion as easy as possible. Two scripts have been provided to automatically convert generic csv files and simp_def.csv to yaml. The first example shows how to convert an Extdata csv file called foo.csv into a Hiera yaml file called bar.yaml:

```
extdata2hiera -i foo.csv -o bar.yaml
```

The second example shows how to convert an Extdata csv simp_def file called simp_def.csv into a Hiera yaml file called simp_def.yaml.
Puppet will automatically retrieve class parameters from Hiera, using lookup keys like myclass::parameter_one. Puppet classes can optionally include parameters in their definition. This lets the class ask for data to be passed in at the time that it’s declared, and it can use that data as normal variables throughout its definition.

There are two main ways to reference Hiera data in puppet manifests. The first, and preferred way, is to use the automatic class variable lookup capability. For each class that you create, the variables will be automatically discovered in hiera should they exist. This is quite powerful in that you no longer need to provide class parameters in your manifests and can finally properly separate your data from your code.

**Note**

For more information on the lookup functions, see [http://docs.puppetlabs.com/hiera/1/puppet.html#hiera-lookup-functions](http://docs.puppetlabs.com/hiera/1/puppet.html#hiera-lookup-functions).

```erb
# Some class file in scope...
class foo (
  $param1 = 'default1'
  $param2 = 'default2'
) { .... }
```

```yaml
# /etc/puppet/hieradata/default.yaml
---
foo::param1: 'custom1'
```

The second is similar to the old Extdata way, and looks like the following:

```erb
$var = hiera("some_hiera_variable", "default_value")
```

The following is from the Puppet Labs documentation, and explains the reason for switching to Hiera.

Automatic parameter lookup is good for writing reusable code because it is regular and predictable. Anyone downloading your module can look at the first line of each manifest and easily see which keys they need to set in their own Hiera data. If you use the Hiera functions in the body of a class instead, you will need to clearly document which keys the user needs to set.

**Note**

For more information on hiera and puppet in general, see [http://docs.puppetlabs.com/hiera/1/complete_example.html](http://docs.puppetlabs.com/hiera/1/complete_example.html).

### 2.11.6 Scope Functions

All scope functions must take arguments in array form. For example in `/etc/puppet/modules/apache/templates/ssl.conf.erb`, `<%= scope.function_bracketize(l) %>` becomes `<% scope.function_bracketize([l]) %>`.

### 2.11.7 Commands

Deprecated commands mentioned in Puppet 2.7 upgrade are now completely removed.

### 2.11.8 Lock File

Puppet agent now uses the two lock files instead of one. These are the run-in-progress lockfile (agent_catalog_run_lockfile) and the disabled lockfile (agent_disabled_lockfile). The puppetagent_cron file (made by the pupmod module) must be edited to suit this change.
2.12 OpenStack Integration

This chapter explains how to integrate OpenStack IceHouse in the SIMP environment.

2.12.1 OpenStack In SIMP - IceHouse

OpenStack is a collection of IaaS cloud computing services aimed at creating a free and open source platform for cloud development and deployment. It is designed in a modular fashion, encompassing the following components:

- **Cinder** Block Storage Service
- **Glance** VM Image Service
- **Keystone** Identity Service
- **Nova** Compute Service
- **Horizon** Dashboard Service
- **Neutron** Networking Service
- **Swift** Object Storage Service
- **Ceilometer** Metrics
- **Heat** Templating

Integration of OpenStack into SIMP provides an easily scalable, secure cloud infrastructure for the end user. Currently, SIMP supports OpenStack IceHouse for CentOS 6.5, with the exception of Swift.

Each OpenStack module has been encapsulated into SIMP as a Puppet module, for rapid deployment. The following modules have been integrated into SIMP for OpenStack support: **Puppetlabs Apache, Cinder, Glance, INIFile, Keystone, MYSQL, Nova, Qpid, Horizon, Memcached, OpenVSwitch, Heat, Ceilometer, Neutron, and OpenStack**.

2.12.2 Deployment

The premise of OpenStack deployment is to create a matrix of control and compute nodes. The control nodes retain all identity, network, database, and communication services; compute nodes run virtualization services.

Each OpenStack SIMP puppet module provides, for itself, the necessary configuration and security required to run inside SIMP. All OpenStack modules are abstracted into a single module, called ‘openstack’`. This module contains pre-loaded configurations for OpenStack deployment, including the framework for basic control and compute nodes.

Please note that each OpenStack puppet module (keystone, glance, cinder, nova, etc.) can run independent of the supplied ‘openstack’ module and manifests. That means you, the end user, can opt to use the example manifests directly or not at all. Create your own manifests for specific site deployments.

To deploy, you must install the pupmod-puppetlabs-openstack module (which will chain-install all dependencies). `/etc/puppet/modules/openstack/simp` has example hieradata yaml files for compute and control nodes.

2.13 Logstash

This chapter gives instruction for getting a basic configuration of Logstash working in a SIMP environment.
2.13.1 Logstash

Logstash is an open source tool that provides a means for SIMP implementations to have logs and events collected, searched, and forwarded (filtered or unfiltered) to another host. SIMP comes with three separate but related modules. The modules are:

- **Logstash**: Installs the RPMs and configuration needed for log inputs, filters, and outputs.
- **Kibana**: Installs the RPMs and configuration needed for the Kibana 3 web interface.
- **Elasticsearch**: Installs the RPMs and configuration needed for Elasticsearch.

**Warning**

The Logstash class is incompatible with the SIMP rsyslog::stock::server class! You cannot enable both of them on the same server.

2.13.2 Logstash Architecture

The overall model for Logstash is very simple. It takes inputs from various sources, optionally applies filters, and outputs the results to a specified target. It’s likely that you can already forward logs to Logstash and output them in a useful format as part of your existing architecture.

Logstash filters can manipulate logs after ingest and before output. Examples of existing filters include fixing logs to split/combine lines, adding fields, normalizing time stamps, and adding GeoIP fields. Depending on the type of log manipulation that is desired, there is likely a filter and associated documentation that already exists.

2.13.3 Logstash SIMP Architecture

Applying the SIMP Logstash, Elasticsearch, and Kibana modules provides an implementation with a functioning log reduction and search capability. Unless scale dictates otherwise, these three modules can easily be applied to a single host.

The intent of providing Logstash in SIMP is to replace the default Rsyslog server with a capability that is easier to search and analyze over time. Once your Logstash server is set up, you simply need to direct your hosts to forward logs to your Logstash server. In a default SIMP configuration, this can be done by setting the $log_server variable in hiera.

**Note**

SIMP does NOT apply any filters to the logs by default.

It is up to each implementation to define and apply filters that meet their local requirements. While multiple output targets may be defined, SIMP only defines the Elasticsearch output by default. Please see the Elasticsearch Puppet module for details on how to define additional output targets.

2.13.4 SIMP Logstash Fow

**Logstash, SIMP, and Security**

The provided SIMP modules for Logstash, Elasticsearch, and Kibana have been built with connection security in mind. Overriding these settings could adversely affect the security of the logging infrastructure. The following list describes the security features in place with the default SIMP module settings:
Warning

The native (Java) Elasticsearch connections are not encrypted! This will be remedied in the future as sufficient methods are found.

• **User Name and Password Protection for Kibana:** The Kibana web can be exposed to a defined list of hosts. If you are connecting to Kibana from anything other than the localhost, a user name and password is required for authentication. Both LDAP and local database users are supported.

• **Syslog over Stunnel:** The default behavior in SIMP is to encrypt syslog traffic over Stunnel. This remains the case with Logstash. Unencrypted traffic is also supported for network devices.

• **Limiting Web Actions:** The Kibana module restricts what HTTP commands a user can perform on the Elasticsearch data store. Full POST action must be given to the Logstash nodes and some nodes may require DELETE capabilities. Logstash hosts should be tightly controlled so that administrative users cannot modify data inside of Elasticsearch with carefully crafted commands. This is one reason that we use syslog on the local hosts.

**Important**

The Puppet modules for Logstash, Kibana, and Elasticsearch contain dozens of variables that may be manipulated. You should read each product’s documentation and ensure you understand any setting that is changed from the default SIMP values. Changes can affect both security and functionality of the system.

### 2.13.5 Logstash Setup

### 2.13.6 Logstash System Requirements

The storage requirements for Logstash and Elasticsearch vary depending on how long you plan on keeping logs. If you use the settings in `?`, then your logs are not being filtered and are being sent to Elasticsearch. When using Elasticsearch, the logs are formatted for Elasticsearch and stored in `/var/elasticsearch`. You can also configure how many days of data you wish to keep in Elasticsearch (keep_days => ‘99’). Therefore, you should ensure you have enough space on `/var` to keep your defined number of days worth of logs.

As you grow your Elasticsearch cluster to handle increasing log loads, you will want to ensure that your keep_days is set to handle your entire cluster appropriately.

**Note**

You should have at least 4G of memory available on any Elasticsearch node.

**Important**

You should NOT install Logstash, Elasticsearch, nor Kibana on your Puppet master. There will likely be conflicts with Apache and resource limitations.

### 2.13.7 Logstash Module Recommended SIMP Setup

The following example manifest can be applied to a single host with a large `/var` volume and 4GB of memory.

```yaml
---
# Add these settings to only your Logstash node.
apache::ssl::sslverifyclient: %{hiera('kibana::ssl_verify_client')}

kibana::redirect_web_root: true
kibana::ssl_allowroot: %{hiera('client_nets')}
kibana::ssl_verify_client: 'none'
# You can add more groups under ldap_groups if you want others
```
In the case of the Elasticsearch node setup below, it may be better to use a group match to pull your Hiera settings. To do this, you should add the following to a file like `/etc/puppet/manifests/nodegroups.pp`

```puppet
if $trusted['certname'] =~ /es\d+.your\.domain/ {
  $hostgroup = 'elasticsearch'
}
```

Then, ensure that a file called `elasticsearch.yaml` is present in the `../only::not simp_4` directory and contains the following content.

```yaml
---
# All nodes running elasticsearch in your cluster should use these settings.
elasticsearch::simp::cluster_name: 'a_unique_hard_to_guess_name'
# This can be no more than the total number of ES nodes that you have in your cluster.
elasticsearch::simp::replicas: '2'
elasticsearch::simp::java_install: true

classes:
- 'elasticsearch::simp'
```

Make sure you point your clients to the Logstash server by setting the `log_server` variable to the fqdn of the Logstash server in hiera. This is further covered in `?`.

### Using LogStash and ElasticSearch

With the default settings applied, you should be able to connect to port 443 on your Kibana host. If connecting from localhost, you will not be prompted for a password. If you are connecting from an external host, a valid LDAP account with that user being defined in the Kibana Class is needed. The page is SSL protected so use `https://<hostname>/kibana`.

With the web interface up, you now have the ability to search logs.
There are several resources available to help with searching. The Kibana Overview Page and Elasticsearch Guide are a good place to start. You should also visit the main Logstash page to see demonstrations and read their tips for searching logs.

2.14 Using Kerberos 5 in SIMP

The Kerberos (Krb5) module helps an administrator obtain a working Key Distribution Center (KDC) setup and configure clients to use the KDC.

**Important**

Given the highly sensitive nature of Kerberos passwords and tokens, this module does not store or use any passwords related to the Kerberos KDC.

Remember the passwords chosen for the Kerberos KDC. Puppet does not have the ability to retrieve forgotten passwords.

As a result of the nature of Kerberos, an administrator must run `/usr/sbin/kdb5_util create -s` on the KDC to set the principal administrator password and initialize the database.

The following sections provide instruction on how to get started with Kerberos 5. For more detailed information, review the official Red Hat documentation at https://access.redhat.com/knowledge/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/Configuring_a_Kerberos_5_Server.html.
2.14.1 Creating Principals

Once all of the systems using Kerberos are properly configured, either via the `krb::stock` classes or otherwise, the administrator must register principals with the KDC.

Create the Admin Principal

The first principal to be registered is an admin principal that manages the environment, since it is in the admin group. This principal must be created on the KDC system.

Before creating the admin principal, the user must first create an Access Control List (ACL) appropriate. To accomplish this, add the following Puppet code to the site manifest for the KDC system. If a custom implementation of Kerberos is being used, changes may need to be made to the code.

Code for Creating an Admin Principal Kerberos

```ruby
krb5_acl{ "${::domain}_admin":
    principal   => "*/admin@${::domain}",
    operation_mask => '*'
}
```

The table below lists the steps to create an admin principal that is appropriate for common organizations. These steps should be accomplished after creating the ACL by using the code provided in the previous example.

Table: Creating the Admin Principal Procedure

Create the Host Principal(s)

Once the admin principal has been created, host principals for each host can be made. The table below lists the steps to complete this action.

Table: Creating Host Principal Procedure

Once the Puppet Agent runs on the clients, the keytabs are copied to the `/etc/krb5_keytabs` directory. The keytab matching the FQDN is set in place as the default keytab, `/etc/krb5.keytab`.

2.15 Jenkins Scripts

Jenkins is the tool that was chosen to perform integration tests on the SIMP system. Jenkins is an application that monitors execution of specific jobs. Below are the scripts that are used to generate VMs from a SIMP ISO and build a basic SIMP test system.

2.15.1 Preconfiguration

This chapter assumes that the following conditions are met before attempting to use the scripts below within Jenkins:

- Jenkins is installed and you can create new jobs through the dashboard
- Libvirt is installed
- The jenkins user is part of the kvm group
- The rubygem rubygem-net-scp is installed
- There is at least 30G of harddrive space available for each VM you wish to create
• A DHCP server exists and is available with an entry for any VMs you expect to create
• You will be able to ssh into the VMs you create as root
• The following preconfigured files exist in /srv/info:
  – simp_conf.csv.<vm_name> - the output of the “simp config” command specific for the VM, one should exist for every VM
  – <reverseip>.db.<vm_name> - set up for the reverse file, one should exist for each VM on the host machine
  – <domain name>.db.<vm_name> - for forward, one should exist for each VM on the host machine
  – named.conf.<vm_name> - configured for the VM, one should exist for each VM on the host machine
  – <domain name> - containing the VM’s zones information
  – dhcpd.conf.<vm_name> - containing all entries for any clients the VM will have, one should exist for each VM on the host machine
  – hosts - file that will replace /etc/hosts on the VM
  – jenkins.pub - the jenkins user public key
  – pupser.pp.<vm_name> - puppet_server node definition containing all necessary tftpboot information, one should exist for each VM on the host machine
  – ldifs - a folder containing a script that will run the ldapadd command correctly and an ldiff containing any users you wish to add
  – ksfiles.sh - a script that configures the config files in /var/www/ks;

          ip=<ipaddress>
          dist=<distribution>
          pushd .
          
          # Update the kickstart files
          sed -i "s/#KSSERVER#/\${ip}/g" /var/www/ks;/*.cfg
          sed -i "s/#YUMSERVER#/\${ip}/g" /var/www/ks;/*.cfg
          sed -i "s/#BOOTPASS#/MD5 Hashed grub password/g" /var/www/ks;/*.cfg
          sed -i "s/#ROOTPASS#/MD5 Hashed root password/g" /var/www/ks;/*.cfg
          sed -i "s/#LINUXDIST#/\${dist}/g" /var/www/ks;/*.cfg
          chown root.apache /var/www/ks;*
          chmod 640 /var/www/ks;*
          chcon --reference=/var/www/ks;/diskdetect_puppet_server.sh /var/www/ks;/*.cfg
          
          # Change server name to client name (fqdn) in togen file
          sed -i 's/<server_fqdn>/<client_fqdn>/g' /etc/puppet/Config/FakeCA/togen
          cd /etc/puppet/Config/FakeCA
          ./gencerts_nopass.sh auto
          popd
          
          # Check /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/rhel<version>-<arch>
          echo "Check that the boot image(s) are owned by root.nobody and perms are 644."
          echo ""
          chgrp -h nobody /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/*initrd.img
          chgrp nobody /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/*vmlinuz
          chmod 644 /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/*initrd.img
          chmod 644 /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/*vmlinuz
          ls -ld /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/*/initrd.img
          ls -ld /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/*vmlinuz
echo ""
ls -l /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/tftpboot/linux-install/*/vmlinuz
echo ""

# Compare pupser.pp with /etc/puppet/manifests/nodes/puppet_servers.pp
diff /srv/info/pupser.pp.u6s40cent /etc/puppet/manifests/nodes/puppet_servers.pp

# add users
/srv/info/ldifs/ladd.cmd /srv/info/ldifs/users.ldif

# add alias
echo "alias root='sudo sudosh'" >> /etc/bashrc

Note
The following modifications were made to ISO the before using it to create a VM:

**Change the bootprotocol to dhcp**

```bash
sed -i 's/network --bootproto=static --ip=192.168.0.111 --netmask=255.255.255.0 --gateway=192.168.1.1 --nodns --hostname=puppet.change.me/network --bootproto=dhcp/g' ks/dvd/include/common_ks_base
```

**Prevent the system from forcing a root password change**

```bash
sed -i 's/change -d 0 root;//g' ks/dvd/*.cfg
```

**Use simp-big by default instead of just simp**

```bash
sed -i 's/default simp$/default simp-big/g' isolinux/isolinux.cfg
```

### 2.15.2 Create a VM

**String Parameters**
- simp_ver - The version of SIMP (2.0.X, 4.0.X, etc.)
- os_dist - The operating system distribution (CentOS or RedHat)
- vm_name - The name of the VM you will be creating
- vm_mac - The MAC address of the VM you will be creating
- build_dir - The directory where your SIMP ISO is stored

```bash
#!/bin/bash
cd ${build_dir}/SIMP-$(simp_ver)
if [ "$(simp_ver)" == "2.0.X" ]; then variant="rhel15.4"; else variant="rhel6"; fi
ISO="ls SIMP-<os_dist>*.iso"
EXIST=$(virsh --connect qemu:///system list --all | grep $vm_name)
EXISTOFF=$(virsh --connect qemu:///system list --all | grep $vm_name | grep "shut off")
ignore='false'
echo "Virsh list:"

virsh --connect qemu:///system list --all

if [ "$1" == '-i' ]; then ignore='true'; fi

if [ ! -d /var/lib/jenkins/VM ]; then mkdir /var/lib/jenkins/VM; fi

if [ "$ignore" == 'true' ] || [ ! -d "/var/lib/jenkins/VM/$vm_name" ]; then
mkdir /var/lib/jenkins/VM/$vm_name

elif [ -f "/var/lib/jenkins/VM/$vm_name/Disk1" ] && [ ! "$(EXISTSOFF)" == "" ]; then
  echo "VM $vm_name already exists, overwriting with the latest..."
  virsh --connect qemu:///system destroy $vm_name
fi

if [ -f "/var/lib/jenkins/VM/$vm_name/Disk1.base" ]; then
  echo "Removing old snapshots"
  rm -rf /var/lib/jenkins/VM/$vm_name/Disk1 /var/lib/jenkins/VM/$vm_name/Disk1/Test
  mv /var/lib/jenkins/VM/$vm_name/Disk1.base /var/lib/jenkins/VM/$vm_name/Disk1

  echo "Undefining $vm_name"
  virsh --connect qemu:///system undefine $vm_name
else
  echo "Creating VM..."
fi

echo "Starting installation of the $vm_name VM via the $ISO"
/usr/bin/virt-install --connect qemu:///system -n "$vm_name" -r 1024 --vcpus=1 --vnc --noautoconsole --os-variant=$variant --os-type=linux -w bridge:br0 --disk=path="/var/lib/jenkins/VM/$vm_name/Disk1",size=30,sparse='false' -v --accelerate --sound --cdrom=/build_dir/SIMP-$simp_ver/$ISO

wait
SUCCESS=`/usr/bin/virsh --connect qemu:///system autostart $vm_name`

if [ "$SUCCESS" == "Domain $vm_name marked as autostarted" ]; then
  still_running () { ps -f -C qemu-kvm | grep $vm_name | grep 'no-reboot' >$ /dev/null; return $?; }; while still_running; do echo -n '>'; sleep 5; done; echo;
sleep 5;

  echo "Starting $vm_name";
  /usr/bin/virsh --connect qemu:///system start $vm_name;
  echo "Waiting for VM to start...";
  still_rebooting () { test ""echo "" | telnet $vm_name 22 2> /dev/null | grep Connected" = ""; }; while still_rebooting; do echo -n '>'; sleep 5; done; echo;

  virsh --connect qemu:///system autostart --disable $vm_name;
  echo "Ready!";
  exit 0;
else
  exit 1;
fi

echo ""; echo "Virsh list:"

virsh --connect qemu:///system list --all

### 2.15.3 Setup VM

**String Parameters**

- **vm_name** - The name of the vm that was created using the previous script
- **vm_ip** - IP address of the VM that was just created

**Password Parameter**

- **vm_pass** - The password for root
#!/bin/bash
Clears the ip from known_hosts.
ssh-keygen -R ${vm_ip}

require 'rubygems'
require 'net/scp'

Net::SSH.start(ENV['vm_name'], 'root', :password => ENV['vm_pass'],
  :auth_methods => "password", :encryption => "aes256-cbc") do |ssh|
  ssh.exec!("mkdir /root/.ssh") do|ch, stream, data|
    puts data
  end
  ssh.exec!("chmod -R 700 /root/.ssh") do|ch, stream, data|
    puts data
  end
  ssh.exec!("mkdir /srv/info") do|ch, stream, data|
    puts data
  end
  puts "Copying over configuration files..."
end

Net::SCP.start(ENV['vm_name'], 'root', :password => ENV['vm_pass'],
  :auth_methods => "password", :encryption => "aes256-cbc") do |scp|
  scp.upload!("/srv/info/jenkins.pub", "/root/.ssh/authorized_keys") do |ch, name|
    puts "Copied jenkins public key to VM."
  end
  scp.upload!("/srv/info", "/srv/", :recursive => true) do |ch, name|
    %x(ls /srv/info).each do |x|
      puts "Copied #{x.chomp} to VM."
    end
  end
  scp.upload!("/srv/isos", "/srv/", :recursive => true) do |ch, name|
    %x(ls /srv/isos).each do |x|
      puts "Copied #{x.chomp} to VM."
    end
end

Net::SSH.start(ENV['vm_name'], 'root', :password => ENV['vm_pass'],
  :auth_methods => "password", :encryption => "aes256-cbc") do |ssh|
  puts "simp config -a /srv/info/simp_conf.csv.#{ENV['vm_name']}"
  ssh.exec!("chmod -R 750 /srv/info/") do|ch, stream, data|
    puts data
  end
  ssh.exec!("simp config -a /srv/info/simp_conf.csv.#{ENV['vm_name']}"
    do|ch, stream, data|
    puts data
  end
  puts "Bootstrapping..."
  ssh.exec!("simp bootstrap -v --no-track") do|ch, stream, data|
    puts data
  end
  puts "Installing java..."
  ssh.exec!("yum install -y java simp-mit rubygem-cucumber rubygem-rspec rubygem-net-ssh") do|ch, stream, data|
    puts data
  end
  puts "Rebooting..",
  ssh.exec!("shutdown -r +1") do|ch, stream, data|
    puts data
end
#!/bin/bash

virsh --connect qemu:///system destroy $vm_name
sleep 60
virsh --connect qemu:///system start $vm_name

echo "Waiting for VM to restart..."
still_rebooting () { test "\"%s\" 2>&1 | grep Connected\" = ""; }
while still_rebooting; do echo -n '>'; sleep 5; done
echo -e "\nVM restarted!"

# The key will have changed after setup; remove again.
ssh-keygen -R $vm_ip

# DHCP Setup
require 'rubygems'
require 'net/ssh'
Net::SSH.start(ENV['vm_name'], 'root', :password => ENV['vm_pass'],
:auth_methods => "password", :encryption => "aes256-cbc") do |ssh|
pus "Configuring DHCP..."
  ssh.exec!("cat /srv/info/dhcp.conf.$vm_name > /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/dhcpd/dhcpd.conf")
do |ch, stream, data|
pus data
end
pus "Configuring hostfile..."
ssh.exec!("cat /srv/info/hosts > /etc/hosts") do |ch, stream, data|
pus data
end

# DNS Setup
require 'rubygems'
require 'net/ssh'
Net::SSH.start(ENV['vm_name'], 'root', :password => ENV['vm_pass'],
:auth_methods => "password", :encryption => "aes256-cbc") do |ssh|
pus "Renaming your.domain to simp.dev..."
  ssh.exec!("mv /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/your.domain /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/simp.dev")
do |ch, stream, data|
pus data
end
pus "Configuring named.conf..."
ssh.exec!("cat /srv/info/named.conf.$vm_name > /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/simp.dev/named/etc/named.conf")
do |ch, stream, data|
pus data
end
pus "Configuring simp.dev zone..."
ssh.exec!("mv /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/simp.dev/named/etc/zones/your.domain /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/simp.dev/named/etc/zones/simp.dev")
do |ch, stream, data|
pus data
end
pus "Configuring reverse lookup..."
ssh.exec!("mv /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/simp.dev/named/var/named/reverse/0.0.10.db /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/simp.dev/named/var/named/reverse/<reverseip>.db")
do |ch, stream, data|
pus data
end
pus "Configuring forward lookup..."
ssh.exec!("mv /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/simp.dev/named/var/named/forward/your.domain.db /var/simp/rsync/CentOS/RHEL_MAJOR_VERSION/domains/simp.dev/named/var/named/forward/simp.dev.db")
do |ch, stream, data|
pus data
end

ssh.exec!("chown root.named /var/simp/rsync/\{OS\}/domains/simp.dev/named/var/named/forward/simp.dev.db")

ssh.exec!("chmod 640 /var/simp/rsync/\{OS\}/domains/simp.dev/named/var/named/reverse/<\{reverseip\}>.db ...")

Net::SSH.start(ENV['vm_name'], 'root', :password => ENV['vm_pass'], :auth_methods => "password", :encryption => "aes256-cbc") do |ssh|
puts "Setting up server to be able to kickstart clients..."
ssh.exec!("/srv/info/ksfiles.sh")
end

#!/bin/bash
if [ `ps -ef | grep puppet | grep -v grep | grep -v Rack | wc -l` -gt 0 ]; then
echo "Waiting for current puppet run to complete...";
fi
while [ `ps -ef | grep puppet | grep -v grep | grep -v Rack | wc -l` -gt 0 ]; do
sleep 5; done
echo "\nPuppet Agent Run - First Pass"
echo ""-

rtn1=${?}
echo "First Pass Return Code: ${rtn1}"
echo -e "Second Pass Return Code: ${rtn2}"
if [ ${rtn2} -eq 0 -o ${rtn2} -eq 2 ]; then
echo " - Successful Puppet Run";
else return 1;
fi

Test a Specific Module ===========
String Parameters
• test_mod - The name of the module you wish to test

yum install -y pupmod-$test_mod-test

if [ ! -d ${WORKSPACE}/junit ]; then mkdir ${WORKSPACE}/junit; fi

if [ ! -f /usr/share/simp/tests/modules/$test_mod/mit_tests/Rakefile ]; then
echo "Testing $test_mod..."
cd /usr/share/simp/tests/modules/$test_mod/mit_tests
if [ ! -f ./results.xml ]; then
rm -f results.xml
mkdir results.xml; chmod 755 results.xml; chgrp puppet results.xml
fi
rake testall:junit

cp /usr/share/simp/tests/modules/$test_mod/mit_tests/results.xml/* xml ${WORKSPACE}/junit/simp/
2.15.4 Create a Client VM

String Parameters

- simp_ver - The version of simp that the client will have loaded on it (2.0.X, 4.0.X, etc.)
- cli_name - The name of the client VM you want to create
- cli_mac - The MAC address of the client VM, this should match an entry that was placed in the dhcp.conf that was created on your server VM

```bash
#!/bin/bash
d1=`date`
if [ "$simp_ver" == "2.0.X" ]; then variant="rhel5.4"; else variant="rhel6"; fi
EXISTS=`virsh --connect qemu:///system list --all | grep $cli_name`
EXISTSOFF=`virsh --connect qemu:///system list --all | grep $cli_name | grep "shut off"

echo "Virsh list:"
virsh --connect qemu:///system list --all

if [ ! -d /var/lib/jenkins/VM ]; then mkdir /var/lib/jenkins/VM; fi
if [ ! -d "/var/lib/jenkins/VM/$cli_name" ]; then mdkir /var/lib/jenkins/VM/$cli_name
elif [ -f "/var/lib/jenkins/VM/$cli_name/Disk1" ] && [ ! "$EXISTS" == "" ]; then echo "VM $cli_name already exists, overwriting with the latest..."; fi
if [ "$EXISTSOFF" == "" ]; then echo "Destroying $cli_name"; virsh --connect qemu:///system destroy $cli_name
fi
if [ -f "/var/lib/jenkins/VM/$cli_name/Disk1.base" ]; then echo "Removing old snapshots"; rm -rf /var/lib/jenkins/VM/$cli_name/Disk1 /var/lib/jenkins/VM/$cli_name/Disk1_Test
mv /var/lib/jenkins/VM/$cli_name/Disk1.base /var/lib/jenkins/VM/$cli_name/Disk1
else echo "Creating VM..."
fi

/usr/bin/virt-install --connect qemu:///system -n "$cli_name" -r 512 --vcpus=1 --vnc --noautoconsole --os-variant=$variant --os-type=linux -w bridge:br0 -m $cli_mac --disk=path="/var/lib/jenkins/VM/$cli_name/Disk1",size=30,sparse='true',bus='virtio' -v --accelerate --pxe
wait
SUCCESS=`/usr/bin/virsh --connect qemu:///system autostart $cli_name`
if [ "$SUCCESS" =="Domain $cli_name marked as autostarted" ]; then
call_running () { ps -ef -C qemu-kvm | grep $cli_name | grep 'no-reboot' >| /dev/null; return $?
}
while call_running; do
  echo 
  sleep 5
done
```
2.16 Troubleshooting SIMP

This chapter provides guidance on how to troubleshoot common problems that occur when installing and using SIMP.

2.16.1 My Services Are Dying!

The following section describes how to mitigate issues relating to destructive reasoning and avoiding destruction of the SIMP system.

Destructive Reasoning with svckill.rb

Most security guides that have been published on the Internet strongly suggest disabling all services that are not necessary for system operation. However, to list every possible service that may be controlled by the chkconfig type on a given system in a manifest would not be useful and would bloat the memory space of the running Puppet process.

As an alternative solution, the SIMP Team implemented the svckill.rb script that runs with every Puppet run.

The svckill.rb script:

- Collects a list of all services on the system. These are the same services that the user sees after typing `chkconfig --list`
- Ignores certain critical services, including Puppet, IPtables, and the network.
- Collects a list of all services that are defined in the manifests and modules.
- Ensures that every service that is defined in the manifests and modules is excluded from the list of services to kill.
- Kills and disables everything else.

Avoiding Destruction

If certain services should not be killed, declare them in the node manifest space.
Note
The key is to declare the services and not set them to any other option. By adding them to the manifest, the `svckill.rb` script will ignore them.

The example below demonstrates this action, assuming that the `keepmealive` service is added to the `chkconfig` preventing a service from being killed by `svckill.rb`.

```ruby
service { "keepmealive": }
```

### 2.16.2 Puppet Server Certificate Issues

If Puppet has any certificate issues, regenerate the server CAs. To do this, remove the contents of the `ssl` folder and regenerate those `.pem` files.

The following table lists the steps to regenerate the server CAs:

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Type <code>service httpd stop</code></td>
</tr>
<tr>
<td>2.</td>
<td>Type <code>rm -rf /var/lib/puppet/ssl</code></td>
</tr>
<tr>
<td>3.</td>
<td>Type <code>puppet cert list --all</code></td>
</tr>
<tr>
<td>4.</td>
<td>Type <code>puppet cert --generate ***&lt;fqdn&gt;***</code></td>
</tr>
<tr>
<td>5.</td>
<td>Type <code>service httpd start</code></td>
</tr>
<tr>
<td>6.</td>
<td>Type <code>puppet agent --test</code></td>
</tr>
</tbody>
</table>

Table: Regenerate the Server CAs

Type `rm -rf /var/lib/puppet/ssl` on the client to delete old certificates.

### 2.17 SIMP FAQs

This chapter answers some of the frequently asked questions (FAQs) about SIMP.

#### 2.17.1 Centralized Logging

SIMP provides a pre-built set of classes within the `rsyslog` module for enabling centralized logging within the infrastructure.

After completing these steps, run Puppet on the server and clients, or wait until after the next run to see logs start to flow.
Enable the Server

To enable the pre-built log server, add the following example code to the designated logging node.

Code to Enable the Server Logging Examples

```yaml
classes:
  - 'simp::rsyslog::stock'
```

Enable the Clients

To have clients send data to the server, make the following changes to the `/etc/puppet/hieradata/simp_def.yaml` file.

Code to Enable the Client Logging Examples

```yaml
log_server="fqdn.of.your.log.server"
```

2.17.2 Changing Puppet Masters

It may be necessary to change the Puppet Master. To point a particular client to a new Puppet Master, follow the steps in the sections below.

On the Client

Enter the following changes into the `/etc/puppet/puppet.conf` file.

Code Changes on Client to Switch Puppet Masters

```yaml
server = new.puppet.master.fqdn
ca_server = new.puppet.master.fqdn
ca_port = 8141
```

To remove all files and sub-directories in the `/var/lib/puppet/ssl` directory, type `cd /var/lib/puppet/ssl`. Then type `rm -rf ./*`.

Assuming the new Puppet Master has been set up to properly accept the client, type `puppet agent --test` to run a full Puppet run while pointing to the new server.

If all goes well, the client will now be synchronized with the new Puppet Master. If not, refer to the SIMP Server Installation section of the SIMP Install Guide and ensure that the new Puppet Master was set up properly.

On the Old Puppet Master

Remove or comment out all items for the client node in the `/etc/puppet/hieradata/hosts` space.

To run `puppet agent` in `noop` mode to ensure that there are no inadvertent errors, type `puppet agent --test --noop`.

2.17.3 Building a Bootable DVD from the SIMP tarball

SIMP is an overlay on top of RHEL, not a complete distribution. As such, the user must build a bootable DVD if provided with the SIMP source code or `tar` file.

To build a bootable SIMP DVD, if provided a RHEL DVD and the SIMP `tar` file, follow the steps in the sections below.

2.17. SIMP FAQs 69
**Build the DVD**

The table below lists the steps to build a SIMP DVD, assuming that the user has copied the DVD to a location with enough space to house and unpack the ISO (around 10G).

Starting from the directory with the ISO, complete the steps outlined below. These steps are based on an example ISO of `rhel-server-5.8-x86_64-dvd.iso`.

Table: Build a SIMP DVD Procedure

The fully bootable SIMP DVD is ready to install on a new system. Replace the RHEL version and architecture to fit the user’s needs. See the Changelog for compatible RHEL versions.

**Use the Alternative Method**

If the Ruby `rake` utility is installed, use the `Rakefile` provided in the `Docs/examples` directory of the `tar` file.

### 2.17.4 Excluding Repositories

By default, SIMP applies updates from all available repositories on a nightly basis. This ensures that bug fixes and security updates are applied to all systems without minute management in Puppet manifests. This section provides guidance on how to include or exclude specific repositories from nightly YUM updates.

**Methodology**

The `common::yum_schedule::repos` and `common::yum_schedule::disable` variables in the `pupmod-common` module control which repositories are enabled for nightly updating. Both variables must be specified in array format.

- `common::yum_schedule::repos` is used to specify an array of repositories from which updates are provided; no other repositories will be used.
- `common::yum_schedule::disable` is used to specify an array of repositories from which updates are not provided; all other repositories will be used.

### 2.17.5 IPtables NAT Rules

See the IPtables Module Reference for notes on using the basic IPtables Module.

**Add NAT Rules**

The user may be required to add *Network Address Translation (NAT)* rules to the IPtables ruleset. To achieve this using the IPtables module, SIMP 1.1.3 or later is required and the `iptables::add_rules` input statement should be used to affect the appropriate changes.

The example below shows an IPtable NAT rule.

Example of an IPtable NAT Rule

```yaml
iptables::add_rules { "nat_global":
  table => "nat",
  first => "true",
  absolute => "true",
  header => "false",
  content => "",
```
:PREROUTING ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]

iptables::add_rules { "nat_test":
  table => "nat",
  header => "false",
  content => ":A PREROUTING --physdev-in
            eth1 -j DROP
            
" }

2.17.6 SSH Keys with Puppet

This section provides guidance on managing a local service account, and propagating keys and sudo permissions for the user via Puppet.

Resource Setup

In cases where there is a need to SSH to a target machine/VM as a local user, there are several items that need to be addressed prior to experiencing a successful connection. First, the group account must exist locally, as well as the local user (belonging to the respective account). PAM must allow the desired group access to the machine/VM, and keys must be put in the correct directories to allow the user access.

The below code is a template to make this happen, with explanations of the generic variables included, as well as 2048-bit RSA key generation.

Code to Set Up a Local User SSH Connection Examples

class <CLASS NAME> { 
  include "ssh"

  group { "<GROUP NAME>":
    gid => "<GROUP ID NUMBER>",
    allowdupe => false,
    ensure => present,
  }

  user { "<USER NAME>":
    uid => "<USER ID NUMBER>",
    allowdupe => false,
    ensure => present,
    gid => "<GROUP NAME>",
    home => "/srv/<USER NAME>",
    managehome => true,
    shell => "/bin/sh",
    require => Group["<GROUP NAME>"]
  }

  file { "/srv/<USER NAME>/.ssh":
    owner => "<USER NAME>",
    group => "<GROUP NAME>",
    mode => "700",
  }

2.17. SIMP FAQs
ensure => directory,
)

ssh_authorized_key { "<USER NAME>":
  type => "ssh-rsa",
  key => ssh_autokey("<USER NAME>", "2048" ),
  target =>"/srv/<USER NAME>/.ssh/authorized_keys",
  require => [
    File("/srv/<USER NAME>/.ssh"),
    User("<USER NAME>")
  ]
}

file { "/srv/<USER NAME>/.ssh/id_rsa":
  mode => "600",
  owner => "<USER NAME>",
  group => "<GROUP NAME>",
  source =>"puppet://$puppet_server/site/ssh_autokeys/<USER NAME>",
  require => Ssh_authorized_key("<USER NAME>")
}

file { "/etc/ssh/local_keys/<USER NAME>":
  ensure => present,
  owner => "root",
  group => "root",
  mode => "644"
  source =>"puppet://$puppet_server/site/ssh_autokeys/<USER NAME>.pub",
  require => Ssh_authorized_key("<USER NAME>")
}

sudo::user_specification { "<USER NAME>":
  user_list => ["<USERNAME>"],
  host_list => "<HOST>",
  runas => "<RUN AS>",
  cmd => ["<COMMAND LIST>"],
  passwd => "false",
  require => User("<USER NAME>")
}

pam::access::manage { "Allow <USER NAME>":
  users => <USER NAME>,
  origins => ['ALL']
}

Variables

The table below provides explanations of the variables included in the template code in the previous section.
### Testing

The table below lists the steps to test that the configuration was applied correctly.

<table>
<thead>
<tr>
<th>Step</th>
<th>Process/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Log on to a server that has the template code configuration applied.</td>
</tr>
<tr>
<td>2.</td>
<td>Type su - <em><strong>&lt;USER NAME&gt;</strong></em></td>
</tr>
<tr>
<td>3.</td>
<td>Type exec /usr/bin/ssh-agent /bin/bash to ensure that ssh-agent has a shell running.</td>
</tr>
<tr>
<td>4.</td>
<td>Type /usr/bin/ssh-add to attach the user’s certificates.</td>
</tr>
<tr>
<td>5.</td>
<td>Type /usr/bin/ssh-add -l to double check that the user’s certificates were added successfully.</td>
</tr>
<tr>
<td>6.</td>
<td>Type ssh <em><strong>&lt;HOST&gt;</strong></em> to SSH to a target machine that has the template code configuration applied.</td>
</tr>
</tbody>
</table>

**NOTE:** This step is optional.

If successful, the user should be authenticated and gain access to the target machine without entering a password. If the user is prompted for a password, check to see if the permissions are set up properly and that the certificate keys are in the correct locations. In addition, check the `/etc/security/access.conf` file to ensure that it contains the user or user’s group in an allow statement. See access.conf(5) for details.

#### 2.17.7 Network-based Initial Server Build

This section provides guidance to install the initial SIMP server via an existing kickstart infrastructure.

**Prepare the Kickstart**

To kickstart the initial server, copy the `netboot.cfg` file into the kickstart location from `ks/` at the root level of the extracted DVD.

Replace the `KS_SERVER` and `KS_BASE` variables in the `netboot.cfg` file to match the system settings.
Kickstart the System

Kickstart the system against the netboot.cfg file; this will build a functional SIMP server identical to the one that the user would have received from the DVD.

Post-Installation

This section describes the post installation procedures to use the server.

Setting up the new YUM repo

All of the SIMP systems must be able to reference two YUM locations after install. The first is the Local repo, which is spawned from the Local directory at the top of the DVD. This is expected to be referenced as http://yum_server/yum/SIMP/<Architecture> by the clients.

The second location is the Updates repo, which contains a repo with all of the base operating system RPMs. This is expected to be referenced as http://yum_server/yum/(RedHat|CentOS)/<Version>/<Architecture>/Updates by the clients.

The user is responsible for adjusting these locations in the pre-existing system; however, the table below lists the steps to adjust these locations on the newly built SIMP server.

Table: Set Up the New YUM Repo Procedure

Follow the instructions in the Client Management for additional assistance.

2.17.8 Overriding the Security Module

The section explains how to override the security module.

Introduction

SIMP includes a module that applies a large set of security-related enforcements to the systems to which it is installed. This module resides in /etc/puppet/modules/sec.

It is recommended that any changes the user needs to make to a base module are done via overrides instead of modifying the base module itself. There are two reasons for this approach: to avoid breaking other logic within the module hierarchies and to avoid erasing all changes by updating the base module via Resource Package Manager (RPM).

Split the Application of the Security Settings

First, break the security settings out of the default application to the nodes. To do this, enter information similar to the example structure below in the site.pp or imported file.

Break Out Security Settings from the Application

class base_config {

  import "common"
  include "common"
  <etc../...>

  < Actions normally done to a node
By entering this information, the user can ensure that all nodes that are built have the advanced security class applied to them if they are not otherwise defined.

Create the Override Class

After providing the logical separation needed to apply the security settings separately from the rest of the site configuration, create a class that overrides `sec::advanced`.

In the example below, creating an override class is done via a site module. The information can also be included in the `site.pp`; however, the site module is more flexible.

The table below lists the steps to create the module directories.

<table>
<thead>
<tr>
<th>Table: Create the Module Directories Procedure</th>
</tr>
</thead>
</table>

Create the contents of the `site/modules/init.pp` file using the example below to override those portions of `sec::advanced` that need to be changed.

Remove Security Module from Site Files

By using the example, the Network File System (NFS) service will be enabled if, and only if, the `$security_override_enable_nfs` variable is set to the value `true`.

**Important**

Variables in Puppet are dependent upon the parse order of the configuration files and cannot be overridden once defined.
Create the Node with the Override

Enter the information from the example below in site.pp or an included file.

```puppet
node clientfqdn {
    # We need to do this to get all of the default site settings
    include "base_config"

    $security_override_enable_nfs = "true"
    include "site::security_override"
}
```

These steps for overriding the security module can be applied as needed to any of the base modules.

2.17.9 Performing One-shot Operations

This section introduces the options provided for performing one-shot commands on all Puppet-managed systems without using Puppet. This is useful when the user needs to perform an action one time in every location, but does not want to enforce that action over time.

Use the PSSH Utility

Parallel Secure Shell (PSSH) has been included in SIMP for some time, but has not been installed by default.

The table below lists the steps to use PSSH.

<table>
<thead>
<tr>
<th>Table: Use PSSH Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Note</td>
</tr>
</tbody>
</table>

Other SSH Options

Using the `-f` option forces `TTY` for SSH, which allows the user to run sudo commands via PSSH.

Using the `-OStrictHostKeyChecking=no` option connects the user to the target servers via SSH even if there is an issue with `~/.ssh/known_hosts`.

2.17.10 Puppet Server Behind a NAT

This section provides guidance for when the Puppet server is behind a NAT but is managing hosts outside the NAT.

To resolve this issue, open the `/etc/puppet/manifests/vars.pp` file and rename the `$puppet_servers` variable to `$puppet_server_hosts_mod`. Then, create a new `$puppet_servers` variable and point it to `template('site/nat_ip_switch.erb')`.

The entries in `vars.pp` should look like the following example.

```
$security_override_enable_nfs = "true"
```
Create a `/etc/puppet/modules/site/templates/nat_ip_switch.erb` file with the content shown in the next example. Change the appropriate portions of the content to meet the needs of the user environment.

**Important**

Ensure that the `.erb` file is owned by `root.puppet` and mode 640.

Source

Create the `nat_ip_switch.erb`

```erb
<% # Edit this variable to provide the IP address mappings. # The left-hand side should contain the internal addresses. # The right-hand side should contain the external addresses.
  t_ipmap = {
    "1.2.3.4" => "10.10.10.10",
    "2.3.4.5" => '10.2.3.4'
  }

  # Edit this regex to match the hosts.
  # This is done with a Regexp; the user can use whichever is preferred.
  # Pure IP matching would be faster using the IPAddr class.
  t_inside_nets = Regexp.new("^5\.*")

  t_pupsrvs = puppet_server_hosts_mod.split(/\s|,|;/)

  # Change the ipaddress variable to the host that the regexp above is matching.
  if not t_inside_nets.match(ipaddress) then
    t_pupsrvs.each_index do |t_i|
      t_vals = t_pupsrvs[t_i].split(/\|/)
      if t_ipmap.include?(t_vals.last) then
        t_vals[-1] = t_ipmap[t_vals.last]
        t_pupsrvs[t_i] = t_vals.join('|')
      end
    end
    t_pupsrvs = t_pupsrvs.join(' ')  
  end
-%>
<%= t_pupsrvs -%>
```

Run `puppet agent -t` on the client to receive the appropriately mapped NAT address of the Puppet server.

If the user cannot connect to the NAT’d Puppet server, change the values in the `/etc/hosts` directory to the correct values and try running `puppet agent -t` again.

### 2.17.11 Redundant LDAP

This section describes how to set up redundant OpenLDAP servers in SIMP.

The version of OpenLDAP in RHEL5 only supports `syncrepl`. Multi-master replication has been added in a more recent version of OpenLDAP but is not currently supported in SIMP. `Syncrepl` is optimal for Wide Area Network (WAN) situations and is the SIMP default.

#### Set up the Master

If the standard `puppet_servers.pp` file in SIMP is being used, the user has a working master server. If not, the following example demonstrates how to use the SIMP `openldap` module to create a server using the `puppet_servers.pp` file.
Source Code for Using an OpenLDAP Server openldap

```bash
# These are some common variables.
# See /etc/puppet/manifests/vars.pp for the stock version.

$ldap_master = 'ldap://ldapmaster.your.domain'

class ldap_common {
    include 'openldap::slapd_pki'

    openldap::slapd::conf { 'default':
        suffix => 'dc=your,dc=domain',
        rootdn => 'dn=LDAPAdmin,ou=People,dc=your,dc=domain',
        rootpw => '(SSHA)$klskf$asoghaagsgagggawawg',
        tlsCertificateFile => "/etc/pki/public/${fqdn}.pub",
        tlsCertificateKeyFile => "/etc/pki/private/${fqdn}.pem",
        client_nets => [ '1.2.3.4/16' ]
    }
}

class ldap_master inherits ldap_common {
    include 'openldap::slapo::syncprov'

    openldap::slapo::syncprov::conf { "default": }
}

node ldapmaster {
    include 'ldap_master'
}
```

Set up the Replicated Servers

Once the master is ready, LDAP slave nodes must be configured to replicate data from the master. The example below shows an the code that should be added to the slave node in Puppet. The actual order of which gets done first is irrelevant; the replicated servers will attempt to contact the master until they are successful.

Source Code to Configure an LDAP Slave Node replication

```bash
class ldap_repl inherits ldap_common {
    include 'openldap::slapd::syncrepl'

    openldap::slapd::syncrepl::conf { "111":
        provider => $ldap_master,
        syncrepl_retry => '60 10 600 +',
        searchbase => 'dc=your,dc=domain',
        starttls => 'critical',
        bindmethod => 'simple',
        binddn => 'cn=LDAPSync,ou=People,dc=your,dc=domain',
        credentials => '<plain text password>',
        updateref => $ldap_master
    }
}

node ldaprepl1 {
    include "ldap_repl"
}

node ldaprepl2 {
```
Promote a Slave Node

Slave nodes can be promoted to act as the LDAP master node. To do this, change the node classifications of the relevant hosts. The following example shows the promotion of the `ldaprepl1` server to the master server.

Source Promoting a Slave Node LDAP

```plaintext
# Change the common ldap server variable to promote the slave node.
$ldap_master = 'ldap://ldaprepl1.your.domain'

node ldapmaster {
    # include 'ldap_master'
}

node ldaprepl1 {
    # include 'ldap_repl'
    include 'ldap_master'
}
```

After the next Puppet run on all hosts, `ldaprepl1` will be promoted to the master and all slave nodes will point to it.

Troubleshooting

If the system is not replicating, it is possible that another user has updated the `$ldap_sync_passwd` and `$ldap_sync_hash` entries in the `/etc/puppet/manifests/vars.pp` file without also updating the value in LDAP itself; this is the most common issue reported by users.

Currently, SIMP cannot self-modify the LDAP database directly; therefore, the LDAP Administrator needs to perform this action. Refer to the User Management chapter for more information on manipulating entries in OpenLDAP.

The example below shows the changes necessary to update the `$ldap_sync` information in LDAP.

Update $ldap_sync Information in LDAP Examples

```plaintext
dn: cn=LDAPSync,ou=People,dc=your,dc=domain
changetype: modify
replace: userPassword
userPassword: <Hash from $ldap_sync_hash>
```

Master Node Demotion

In the event that multiple master nodes have been set up, it may be necessary to demote one or more of them to slave instances. To do this, add the replication code shown in the previous section titled Set up the Replicated Servers to the manifest of the master node being demoted.

Once this is complete, manually remove the active database from the LDAP server being demoted and then run Puppet. The SIMP team is working to enable SIMP to handle this transition automatically in the future.

2.17.12 SFTP Restricted Account

This section describes the method for restricting an account to SSH File Transfer Protocol (SFTP) access only.
Add a User

Create a user account based on the following example.

Create a User Account Examples

```
user { "foo":
  uid => <UID>,
  gid => <GID>,
  shell => <Path to SFTP Server>
}
```

On a SIMP system, shell would be:
"/usr/libexec/openssh/sftp-server"

Modify `/etc/shells`

To modify `/etc/shells` to include the shell information provided in the previous user account example, add `common::shells` in Hiera, and add `/usr/libexec/openssh/sftp-server` to the list.

2.17.13 SSH Keys in LDAP

This section provides guidance on managing SSH keys within the SIMP environment.

LDAP Enabled

When enabled, ssh keys are both stored and retrieved directly from LDAP.

See the ? chapter for more information on user management in LDAP.

Without LDAP

If LDAP is not being used, use the `/etc/ssh/local_keys` directory for all user keys.

2.18 SIMP RPMs

This provides a comprehensive list of all SIMP RPMs and related metadata. Most importantly, it provides a list of which modules are installed by default and which are simply available in the repository.

The data will be updated over time as users request additional information.
pupmod-jenkins 4.1.0-6  true  pupmod-krb5 4.1.0-3  true  pupmod-libvirt 4.1.0-15  true  pupmod-logrotate 4.1.0-2  true  pupmod-mcafe 4.1.0-2  true  pupmod-mozilla 4.1.0-1  true  pupmod-mrepo 4.1.0-3  true  pupmod-multipathd 4.1.0-2  true  pupmod-named 4.2.0-6  true  pupmod-network 4.1.0-4  true  pupmod-nfs 4.1.0-12  true  pupmod-puppetlabs-nova 4.0.0-2  true  pupmod-nscl 5.0.0-4  true  pupmod-ntpd 4.1.0-7  true  pupmod-oddjob 1.0.0-1  true  pupmod-openldap 4.1.0-17  true  pupmod-openscap 4.2.0-2  true  pupmod-openshift_origin 0.1.0-3  true  pupmod-puppetlabs-openstack 4.2.0-2  true  pupmod-pam 4.1.0-9  true  pupmod-pki 4.1.0-4  true  pupmod-polkit 4.1.0-1  true  pupmod-postfix 4.1.0-4  true  pupmod-pupmod 6.0.0-16  true  pupmod-ciscosystems-quantum 0.2.3-alpha2  true  pupmod-rsync 4.1.0-6  true  pupmod-rsyslog 4.1.0-13  true  pupmod-selinux 1.0.0-4  true  pupmod-shrinken 4.1.0-RC3  true  pupmod-simp 1.0.0-6  true  pupmod-simp-qpid 2.0.0-2  true  pupmod-site 2.0.0-2  true  pupmod-smnqd 4.1.0-3  true  pupmod-ssh 4.1.0-8  true  pupmod-sssd 4.1.0-5  true  pupmod-stunnel 4.2.0-6  true  pupmod-sudo 4.1.0-1  true  pupmod-sudossh 4.1.0-2  true  pupmod-svckill 1.0.0-4  true  pupmod-systctl 4.1.0-3  true  pupmod-tcpwrappers 3.0.0-2  true  pupmod-tftppboot 4.1.0-6  true  pupmod-tpm 0.0.1-6  true  pupmod-upstart 4.1.0-3  true  pupmod-vnc 4.1.0-2  true  pupmod-vsftpd 4.1.0-8  true  pupmod-windowmanager 4.1.0-2  true  pupmod-xinetd 2.1.0-2  true  pupmod-xwindows 4.1.0-3  true

2.19 SIMP 4.2.0-Beta

2.19.1 Changelog

Contents

- SIMP 4.2.0-Beta
  - Changelog
    * SIMP 4.2.0-Beta
      - Significant Updates
      - Upgrade Guidance
      - Expectations
      - Security Announcements
      - CVEs Addressed
      - RPM Updates
      - Fixed Bugs
      - New Features
      - Known Bugs

SIMP 4.2.0-Beta

Package: 4.2.0-Beta

This release is known to work with:

- RHEL 6.6 x86_64
- CentOS 6.6 x86_64

Significant Updates

- The rsyslog module has been completely rewritten to support rsyslog 7.4. This is a breaking change from previous releases and will require active updates to existing systems. All modules with rsyslog integration have been updated to accommodate this change:
  - aide
• In RHEL6, we updated the OpenLDAP password policy overlay to not conflict with the upcoming 6.7 update. However, this will require you to update your LDAP server schema **manually** with the attached LDIF. Additionally, there was a bug in previous versions of SIMP that can be fixed by running this LDIF as is in RHEL7 and replacing `simp_check_password.so` with `check_password.so` in RHEL6.

• The Electrical and SIMP modules for elasticsearch have been combined.

**Upgrade Guidance**

Fully detailed upgrade guidance can be found in the **Upgrading SIMP** portion of the *User’s Guide*.

**Warning:** You must have at least **2GB of free RAM** on your system to upgrade to this release.

**Note:** Upgrading from releases older than 4.0 is not supported.

**Expectations**  Before you begin, please be aware that the following actions will take place as a result of the migration script:

• The `puppet-server` RPM will be removed

• The `puppetserver` RPM will be installed (no, that’s not a typo)

• **ALL** SIMP Puppet code will be migrated into a new `simp` environment
  
  – This will be located at `/etc/puppet/environments/simp`

• A backup of your running environment will be made available at `/etc/puppet/environments/pre_migration.simp`
  
  – You will find timestamped directories under the `pre_migration.simp` directory that correspond to runs of the migration script

  – Your old files will be in a `backup_data` directory and will be linked to a local bare Git repository in the same space

The upgrade steps will also have you install PuppetDB. PuppetDB is installed by default if you kick from the DVD.

**Security Announcements**

**CVEs Addressed**
RPM Updates

Numerous RPMs were updated in the creation of this release. Several were included due to our use of repoclosure to ensure that RPM dependencies are met when releasing a DVD.

- This version upgrades Facter to 2.4.

Fixed Bugs

- pupmod-aide
  - Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.

- pupmod-apache
  - Remove the apache_version fact and simply use the version controls built into the Apache configuration language.
  - Update all custom functions to properly scope definitions.
  - Ensure that mod_ldap is installed in SIMP >= 5.0.

- pupmod-simp-apache
  - Prevent apache from restarting after downloading a CRL.

- pupmod-clamav
  - Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.

- pupmod-common
  - We no longer supply crontab or anacrontab in global_etcd.
  - Remove dynamic_swappiness cron job if a static value is set.
  - Ensure that the passgen() function fails on invalid scenarios. This prevents the accidental creation of empty passwords.
  - Allow the value 2 to be used for `rp_filter` in `common::sysctl`.
  - Added ability to return remote ip addrs.

- pupmod-dhcp
  - Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.

- pupmod-iptables
  - Fixed a bug that would cause issues with Ruby 1.8.7.
  - Fixed DNS resolution in IPv6.
  - Prevent IPv6 ::1 spoofed addresses by default.

- pupmod-simp-elasticsearch
  - Ensured that Elasticsearch works properly with the new version of Apache.
  - Removed our default ES tuning since the default works better for LogStash.
  - Ensure that Puppet manages the Elasticsearch logging file.
• pupmod-functions
  – Fixed sysv.rb to explicitly require puppet/util/selinux, which caused puppet describe to have errors.

• pupmod-simp-logstash
  – Fix issues with both TCPWrappers and IPTables when used with LogStash.

• pupmod-nfs
  – Updated the `mountd` port to be 20048 by default for SELinux issues in RHEL7.

• pupmod-ntp
  – Updated against NTP Security Vulnerabilities (Red Hat Article #1305723).
  – Ensure that `restrict` entries use DDQ format.

• pupmod-openldap
  – The Password Policy overlay was getting loaded into the default.ldif even if you didn’t want to use it. This has been fixed.
  – Made the password policy overlay align with the latest SIMP build of the plugin.
    * This means that you must have version simp-ppolicy-check-password-2.4.39-0 or later available to the system being configured.
  – Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.
  – Fixed reported bugs in syncrepl.pp.

• pupmod-openscap
  – Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.
  – Changed default ssg base path to `/usr/share/xml/scap/ssg/content`

• pupmod-rsync
  – Fixed provider to run with --dry-run when puppet is run with a --noop.

• pupmod-ssh
  – Modernized the Ciphers, MACs, and Kex.
  – Added explicit cases for FIPS and non-FIPS mode (as well as reasonable default cases for RHEL7 and below).
  – Updated to use the new augeasproviders module dependencies.
  – Added a function `ssh_format_host_entry_for_sorting()` that will properly sort SSH Host entries for inclusion with concat.

• pupmod-stunnel
  – Had a variable options in `stunnel.erb` that should have been scoped as `@options`.

• pupmod-sudosh
  – Change the call to the `rsyslog` init script to the `service` command to seamlessly support both RHEL6 and RHEL7.

• pupmod-sysctl
  – Removed support for the old parsed-file provider and moved to using the new Augeas-based provider.
• pupmod-tftpboot
  – Purging of non-Puppet-managed items in pxelinux.cfg is now optional.
• pupmod-simp-tpm
  – IMA is disabled by default.
• simp-utils
  – Fixed the targets of unpack_dvd.
• pupmod-xinetd
  – Fixed: The default log_type should be ‘SYSLOG authpriv’ instead of ‘SYSLOG daemon info’.
• pupmod-vnc
  – Removed banners that broke some vnc clients.
• DVD
  – A default IP is no longer provided when booting from the ISO; simp config will set the network properly.

New Features

• pupmod-augeasproviders
  – This was updated to 2.1.3.
  – The update to 2.1.3 caused the addition of all of the pupmod-augeasproviders modules below.
• augeasproviders_apache
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_base
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_core
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_grub
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_mounttab
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_nagios
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_pam
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_postgresql
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_puppet
  – Imported 2.1.3 to support the Augeasproviders stack.
• augeasproviders_shellvar
– Imported 2.1.3 to support the Augeasproviders stack.

• augeasproviders_ssh
  – Imported 2.1.3 to support the Augeasproviders stack.

• augeasproviders_sysctl
  – Imported 2.1.3 to support the Augeasproviders stack.

• pupmod-common
  – Created parse_hosts function.

• pupmod-richardc-datacat
  – Incorporated the richardc/datacat module into the core for user convenience.

• pupmod-freeradius
  – Split the Freeradius module based on version so that it can be properly selected against the installed version of Freeradius. This may take two runs to coalesce.

• pupmod-puppetlabs-inifile
  – Updated to version 1.2.0.

• pupmod-pki
  – Now generate a system RSA public key against the passed private key.

• pupmod-puppetlabs-postgresql
  – Initial import of the Puppet Labs PostgreSQL module.
  – Modifications were made to support the SIMP concat.

• pupmod-puppetlabs-puppetdb
  – New import of the Puppet Labs PuppetDB module.

• pupmod-puppetlabs-stdlib
  – Updated to version 4.5.1.

• pupmod-tftpboot
  – Updated to use native packages and pull as much as possible.

• Mcollective
  – Mcollective is now available to be installed and used with SIMP. It uses SSL/TLS along with user certificates for proper encryption and authentication.

• PuppetDB
  – PuppetDB is now supported by SIMP and installed by default.

• Puppetserver
  – The puppet master service has been replaced by the puppetserver service. This is a major rewrite by Puppetlabs. Puppetserver scales better for larger agent deployments with a single puppet master.
  – Uses Environments by default, this allows for tools such as r10K. Production environment is a link to simp by default.

• simp config
  – simp config was rewritten to allow for new features and flexiblty.
Now provided as a Ruby gem “simp-cli”.

- pupmod-simp-logstash
  - Integrated SIMP and Electrical Logstash modules.
  - Changes the existing Logstash module to allow users to apply default SIMP filters.

- simp-rsync
  - Content has been restructured to eliminate licensing conflicts.
  - ClamAV has been refactored into a separate (GPL) package.

- pupmod-simp-rsyslog
  - Module has been rewritten to support rsyslog 7.4.

- pupmod-simp-kibana
  - Add Kibana dashboards to the Kibana module.
  - Allows users to apply default SIMP kibana Dashboards.

- Facter 2.4
  - Facter now returns the following facts as their actual boolean or integer values, instead of converting them into strings:
    - activeprocessorcount
    - is_virtual
    - mtu_<INTERFACE>
    - physicalprocessorcount
    - processorcount
    - selinux_enforced
    - selinux
    - sp_number_processors
    - sp_packages

Known Bugs

- Setting pwdReset to ‘true’ in LDAP does not force a user to reset their password like it is supposed to. This works with FreeIPA and we are looking to move to support that system in the future.

- SSSD is currently broken and will allow logins via SSH even if your password has expired. This has been noted by Red Hat and is in the pipeline. Their suggestion it to move to FreeIPA from OpenLDAP. We are looking to do this in the future.

- If you are running libvirtd, when svckill runs it will always attempt to kill dnsmasq unless you are deliberately trying to run the dnsmasq service. This does not actually kill the service but is, instead, an error of the startup script and causes no damage to your system.

2.20 Glossary of Terms

Access Control List (ACL)  A list of permissions attached to an object. An ACL specifies which users or system processes are granted access to objects, as well as what operations are allowed on given objects. Each entry in a typical ACL specifies a subject and an operation.

Advanced Intrusion Detection Environment (AIDE)  An intrusion detection system for checking the integrity of files under Linux. AIDE (Advanced Intrusion Detection Environment) can be used to help track file integrity by comparing a snapshot of the system’s files prior to and after a suspected incident. It is maintained by Rami Lehti and Pablo Virolainen.

Auditd  auditd is the userspace component to the Linux Auditing System. It’s responsible for writing audit records to the disk. Viewing the logs is done with the ausearch or aureport utilities. Configuring the audit rules is done with the auditctl utility. During startup, the rules in /etc/audit/audit.rules are read by auditctl. The audit daemon
itself has some configuration options that the admin may wish to customize. They are found in the `auditd.conf` file.

**Community Enterprise Operating System (CentOS)**  An Enterprise-grade Operating System that is directly compatible with a prominent US Linux distribution.

**Domain Name System (DNS)**  A database system that translates a computer’s fully qualified domain name into an IP address and the reverse.

**Dynamic Host Configuration Protocol (DHCP)**  A network protocol that enables a server to automatically assign an IP address to a computer.

**Fully Qualified Domain Name (FQDN)**  A domain name that specifies its exact location in the tree hierarchy of the Domain Name System (DNS). It specifies all domain levels, including the top-level domain and the root zone. An FQDN is distinguished by its unambiguity; it can only be interpreted one way.

**Hard Disk Drive (HDD)**  A device for storing and retrieving digital information, primarily computer data.

**Internet Protocol Tables (IPtables)**  A user space application that provides an interface to the IPv4 firewall rules on modern Linux systems.

**Internet Protocol 6 Tables (IP6tables)**  A user space application that provides an interface to the IPv6 firewall rules on modern Linux systems.

**Kerberos (Krb5)**  A computer network authentication protocol that works on the basis of “tickets” to allow nodes communicating over a non-secure network to prove their identity to one another in a secure manner.

**Key Distribution Center (KDC)**  Part of a cryptosystem intended to reduce the risks inherent in exchanging keys. KDCs often operate in systems within which some users may have permission to use certain services at some times and not at others.

**Lightweight Directory Access Protocol (LDAP)**  A protocol for querying and modifying LDAP directory services including information such as names, addresses, email, phone numbers, and other information from an online directory.

**Network Address Translation (NAT)**  The process of modifying IP address information in IP packet headers while in transit across a traffic routing device.

**Network File System (NFS)**  A distributed file system protocol that allows a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

**Parallel Secure Shell (PSSH)**  A tool that provides parallel versions of OpenSSH and other related tools.

**Pluggable Authentication Modules (PAM)**  A mechanism to integrate multiple low-level authentication schemes into a high-level application programming interface (API). It allows programs that rely on authentication to be written independent of the underlying authentication scheme.

**Practical Extraction and Report Language (PERL)**  A high-level, general-purpose, interpreted, dynamic programming language. Perl was originally developed by Larry Wall in 1987 as a general-purpose Unix scripting language to make report processing easier.

**Preboot Execution Environment (PXE)**  An environment to boot computers using a network interface independently of data storage devices (like hard disks) or installed operating systems.

**Privacy Enhanced Mail (PEM)**  An early standard for securing electronic mail. This is the public-key of a specific certificate. This is also the format used for Certificate Authority certificates.

**Public Key Infrastructure (PKI)**  A security architecture that has been introduced to provide an increased level of confidence for exchanging information over an increasingly insecure Internet. PKI enables users of a basically insecure public networks, such as the Internet, to securely authenticate to systems and exchange data. The exchange of data is done by using a combination of cryptographically bound public and private keys.
Puppet  A Open Source configuration management tool written and maintained by Puppet Labs. Written as a Ruby DSL, Puppet provides a declarative language that allows system administrators to provide a consistently applied management infrastructure. Users describes system resource and resource state in the Puppet language. Puppet discovers system specific information via facter and compiles Puppet manifests into a system specific catalog containing resources and resource dependencies, which are applied to each client system.

Random Access Memory (RAM)  A form of computer data storage. A random access device allows stored data to be accessed in nearly the same amount of time for any storage location, so data can be accessed quickly in any random order.

Red Hat  A collection of many different software programs, developed by Red Hat®, Inc. and other members of the Open Source community. All software programs included in Red Hat® Linux® are GPG signed by Red Hat to indicate that Red Hat supplied them.

See also RHEL.

Red Hat Enterprise Linux (RHEL)  A commercial Linux operating system produced by Red Hat®, Inc. RHEL is designed to provide an Enterprise-ready Linux distribution suitable to multiple target applications.

See also Red Hat Linux.

Resource Package Manager (RPM)  A package management system. The name RPM is associated with the .rpm file format, files in this format, software packaged in such files, and the package manager itself. RPM was developed primarily for GNU/Linux distributions; the file format is the baseline package format of the Linux Standard Base.

RSA  An algorithm for public-key cryptography that is based on the presumed difficulty of factoring large integers, the factoring problem. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described it in 1977.

Ruby  A dynamic, reflective, general-purpose object-oriented programming language that combines syntax inspired by Perl with Smalltalk-like features. Ruby originated in Japan during the mid-1990s and was first developed and designed by Yukihiro “Matz” Matsumoto. It was influenced primarily by Perl, Smalltalk, Eiffel, and Lisp. Ruby supports multiple programming paradigms, including functional, object oriented, imperative and reflective. It also has a dynamic type system and automatic memory management; it is therefore similar in varying respects to Smalltalk, Python, Perl, Lisp, Dylan, Pike, and CLU.

Secure Shell (SSH)  An application for secure data communication, remote shell services, or command execution between networked computers. SSH utilizes a server/client model for point-to-point secure communication.

Secure Sockets Layer (SSL)  The standard security technology for using PKI keys to provide a secure channel between two servers.

See also TLS.

SIMP  A security framework that sits on top of RHEL or CentOS.

SSH File Transfer Protocol (SFTP)  A network protocol that provides file access, file transfer, and file management functionalities over any reliable data stream. It was designed by the Internet Engineering Task Force (IETF) as an extension of the Secure Shell protocol (SSH) version 2.0 to provide secure file transfer capability, but is also intended to be usable with other protocols.

Sudosh  A filter that can be used as a login shell.

Transport Layer Security (TLS)  A cryptographic protocol that provides network communications security. TLS and SSL encrypt the segments of network connections above the Transport Layer, using asymmetric cryptography for privacy and a keyed message authentication codes for message reliability.

See also SSL.

Trivial File Transfer Protocol (TFTP)  A file transfer protocol generally used for automated transfer of configuration or boot files between machines in a local environment.

2.20. Glossary of Terms 89
TTY  A Unix command that prints to standard output the name of the terminal connected to standard input. The name of the program comes from teletypewriter, abbreviated “TTY”.

Virtual Machine (VM)  A completely isolated guest operating system installation within a normal host operating system.

Virtual Network Computing (VNC)  A graphical desktop sharing system that uses the remote framebuffer (RFB) protocol to control another computer remotely. It transmits the keyboard and mouse events from one computer to another, relaying the graphical screen updates back in the other direction, over a network.

Wide Area Network (WAN)  A computer networking technology used to transmit data over long distances, and between different Local Area Networks (LANs), Metropolitan Area Networks (MANs), and other localized computer networking architectures.

Yellowdog Updater, Modified (YUM)  A software installation tool for Linux. It is a complete software management system that works with RPM files. Yum is designed to be used over a network or the Internet.

See also RPM.

### 2.21 Appendix A - Sample Puppet Files

This is a sample reverse (e.g. 0.0.10.db) file

```plaintext
$TTL 86400
@ IN SOA simp.net. sample.simp.net. ( 2015061101
  1h ; refresh
  15m ; retry
  1w ; expiry
  1d ) ; minimum

IN NS sample.simp.net.

15 IN PTR sample.simp.net.
```

This is a sample forward (e.g.your.domain) file

```plaintext
$TTL 86400
simp.net. IN SOA sample.simp.net. hostmaster.simp.net. ( 2015061201
  3H ; refresh
  15M ; retry
  1W ; expiry
  1D ) ; minimum

IN NS sample.simp.net.

sample IN A 192.168.122.15
```

This is a sample (e.g. dhcpd.conf) dhcpd file

```plaintext
allow booting;
allow bootp;
ddns-update-style interim;

class "pxeclients" {
  match if substring(option vendor-class-identifier, 0, 9) = "PXEClient";
  next-server 192.168.122.15;
  filename "linux-install/pxelinux.0";
}
```
subnet 192.168.122.0 netmask 255.255.255.0 { 
  option routers 192.168.122.1; 
  option subnet-mask 255.255.255.0; 
  option domain-name "simp.net"; 
  option domain-name-servers 192.168.122.15; 
  option time-offset -0; 
  default-lease-time 21600; 
  max-lease-time 43200; 

  # We explicitly list our hosts to restrict the hosts that can access our 
  # network. 
  # host sample { 
  #   hardware ethernet DE:AD:BE:EF:00:00; 
  #   fixed-address 192.168.122.16; 
  # } 
}

This is a sample named file

#FIX_YOUR_NAMED_CONFIG_PRIOR_TO_RUNNING

acl trusted_hosts {
  127.0.0.1;
};

options {
  version " ";
  query-source port 53;
  query-source-v6 port 53;

  directory "/var/named";
  dump-file "data/cache_dump.db";
  statistics-file "data/named_stats.txt";
  memstatistics-file "data/named_mem_stats.txt";

  allow-query { 192.168.122.0/24; };
  allow-recursion { 192.168.122.0/24; };
  allow-transfer { "none"; };
};

logging {
  channel default_syslog {
    syslog daemon;
    print-category yes;
    severity info;
  };

  channel secure_syslog {
    syslog local6;
    print-category yes;
    severity notice;
  };

  category "default" {
  }}
"default_syslog"; }

category "xfer-out" {
    "secure_syslog";
};

category "xfer-in" {
    "secure_syslog";
};

category "dnssec" {
    "secure_syslog";
};

category "update" {
    "secure_syslog";
};

category "notify" {
    "secure_syslog";
};

category "update-security" {
    "secure_syslog";
};

category "lame-servers" {
null;
};

include "/etc/rndc.key";

controls {
    inet 127.0.0.1 allow { localhost; } keys { rndckey; }
};

zone "localdomain" IN {
    type master;
    file "localdomain.zone";
    allow-update { none; }
};

zone "localhost" IN {
    type master;
    file "localhost.zone";
    allow-update { none; }
};

zone "0.0.127.in-addr.arpa" IN {
    type master;
    file "named.local";
    allow-update { none; }
};

zone "255.in-addr.arpa" IN {
    type master;

This is a sample zones (e.g. your.domain) file

```nginx
zone "simp.net" IN {
    type master;
    file "forward/simp.net.db";
};

zone "122.168.192.in-addr.arpa" IN {
    type master;
    file "reverse/122.168.192.db";
};
```

This is a sample kickstart (e.g. pupclient_x86_64.cfg) file

```bash
# Replace the following strings in this file
# #BOOTPASS# - Your MD5 hashed bootloader password = (your encrypted PW)
# #ROOTPASS# - Your MD5 hashed root password = (your encrypted PW i.e.$6$sk6ngonx...YFLEkw8161L
# #KSSERVER# - The IP address of your YUM server = 192.168.122.15
# #YUMSERVER# - The IP address of your YUM server = 192.168.122.15
# #LINUXDIST# - The LINUX Distribution you are kickstarting
#       - Current CASE SENSITIVE options: RedHat CentOS

authconfig --enableshadow --passalgo=sha512 --enablemd5
bootloader --location=mbr --append="console=ttyS1,57600 console=tty1" --iscrypted --password=$6$sk6ngonx$7c0eXVPVg0E7TqB1l.U0L8dugFIH89fWwppITz3Yu2QBkzK2ak69seYFLEkw8161L
zerombr
key --skip
firewall --enabled --ssh
firstboot --disable
logging --level=info
network --bootproto= dhcp
reboot
selinux --permissive
timezone --utc GMT
install
skipx

%include /tmp/repo-include
```

2.21. Appendix A - Sample Puppet Files
keyboard us
lang en_US
url --url http://192.168.122.15/yum/CentOS/6/x86_64

%include /tmp/part-include

%packages --nobase
redhat-lsb
xorg-x11-xauth
acl
rpm
yum
bzip2
crontabs
libutempter
pciutils
psacct
quota
tmpwatch
vixie-cron
amtu
anacron
coolkey
cpuspeed
cryptsetup-luks
dhclient
gnupg
irqbalance
krb5-workstation
libaio
logwatch
logrotate
mdadm
microcode_ctl
nss_db
openssh-clients
openssh-server
pam_krb5
pam_passwdqc
pam_pkcs11
ntp
readahead
smartmontools
stunnel
sudo
tcp_wrappers
unzip
usbutils
vconfig
wget
which
zip
side
iptables
iptables-ipv6
netlabel_tools
audit
lsscsi
net-snmp
sysstat
vlock
mutt
subversion
-sysklogd
rsyslog
isof
vim-enhanced
-sendmail
dracut-fips
dracut-kernel
# Puppet stuff
puppet
facter
rsync
ruby-rdoc
# In case of broken repo, these should be installed.
hdparm
kbd
libhugetlbfs
policycoreutils
prelink
rootfiles
selinux-policy-targeted
setserial
sysfsutils
udftools
$end

$pre
ksserver="192.168.122.15"
wget -O /tmp/diskdetect.sh http://$ksserver/ks/diskdetect.sh;
chmod 750 /tmp/diskdetect.sh;
/tmp/diskdetect.sh;
wget -O /tmp/repodetect.sh http://$ksserver/ks/repodetect.sh;
chmod 750 /tmp/repodetect.sh;
/tmp/repodetect.sh '6' $ksserver;
$end

$post
ostype="CentOS"
if [ $ostype == "CentOS" ]; then
  sed -i '/enabled=/d' /etc/yum.repos.d/CentOS-Base.repo;
  sed -i '/\[.*\]/ a
  enabled=0' /etc/yum.repos.d/CentOS-Base.repo;
fi
ksserver="192.168.122.15"

# Add boot target to GRUB kernel command line
grubby --update-kernel='grubby --default-kernel' \
--args="boot=`blkid \"/boot\" | cut -f1 -d' ' | cut -f2 -d' ' | tr -d '"'"

# Turn off prelinking and remove all previous
sed -i '/PRELINKING=yes/ c\PRELINKING=no' /etc/sysconfig/prelink
prelink -u -a

# Enable the firstboot bootstrapping script.

2.21. Appendix A - Sample Puppet Files 95
wget --no-check-certificate -O /etc/init.d/runpuppet http://$ksserver/ks/runpuppet;
chmod 700 /etc/rc.d/init.d/runpuppet;
chkconfig --add runpuppet;
chkconfig --level 35 runpuppet on;
%end
3.1 Introduction

This manual describes the security concepts of the SIMP system. The system was originally designed to meet a specific set of technical security controls using industry best practices and has been modified recently to meet as many of the security controls provided by the National Institute of Standards and Technology’s (NIST) special publication 800-53 as possible.

This manual outlines three categories of security:

- **Technical Architecture**: discusses the technical approaches to securing the system
- **Operational Security**: discusses the security of SIMP in an operational setting
- **Information System Management**: discusses how SIMP helps achieve security in terms of system management

A brief discussion of how the SIMP system helps achieve categories of controls is provided; additional technical details regarding each control can be found in the SIMP Security Controls Traceability Matrix (SCTM).

When possible, the security control identifier will be found at the end of a concept to provide the reader with a reference to the specific control that is being discussed. The identifier is written as [AB-X(Y)], where A is the control family, X is the control section, and Y is the control enhancement.

3.2 Technical Security

This chapter contains SIMP security concepts that are related to the technical security controls described in *NIST 800-53*.

3.2.1 Identification and Authentication

This section addresses the identification and authentication of users and devices.
3.2.2 User Identification and Authentication

Identification and authentication of system and service users can occur at the system level or globally in the SIMP architecture. While local accounts and groups can be created manually, the SIMP team suggests adding users via the `/etc/puppet/localusers` file or by using the native Puppet user and group types. System users can authenticate their access using Secure Shell (SSH) keys or passwords. For more centralized control, identify and authenticate users by using the Lightweight Directory Access Protocol (LDAP). [IA-2]

The SIMP team recommends using LDAP as the primary source for user management and provides a functional default OpenLDAP configuration for this purpose. LDAP and Pluggable Authentication Modules (PAM) work together closely and, with the default SIMP configuration, the PAM settings are enforced on top of the LDAP settings for two layers of control. Due to this partnership, items such as account lockouts may need to be reset on both the local system and the LDAP server. If the suggested settings in the SIMP-provided default Lightweight Directory Interchange Formats (LDIF) are not used, implementations must ensure that security is maintained through manual procedures. Use of group accounts for users is strongly discouraged. System services may need to have accounts, but all of these should be managed by Puppet using the user and group native types. [IA-2(5)].

3.2.3 Device Identification and Authentication

Devices are identified by a Media Access Control (MAC) address prior to receiving an IP address via the Dynamic Host Configuration Protocol (DHCP). In the default SIMP architecture, IP addresses are fixed mappings to their associated MAC address (i.e., not assigned dynamically). There is no authentication for the binding of MAC addresses to IP addresses due to the nature of the DHCP protocol.

Device authentication occurs through the mapping of the MAC to the IP through the internally controlled DHCP and the mapping of the IP to the host name through the internally controlled Domain Name System (DNS) service for each individual Puppet client. After kickstart, each client system generates an internal cryptographic identifier and communicates that information with the Puppet server to be approved by an administrator at a later time. All further communication between the Puppet server and the clients over the Puppet protocol is encrypted subsequently and authenticated with this identifier. Automatic approval can be set up in tightly controlled environments; however, this option is not suggested for open environments. [IA-3, IA-3(3)]

3.2.4 Identifier Management

Managing user identifiers (also known as user names) involves administrative procedures that are unique for each implementation. Disabling unused local accounts is the only control that SIMP can enforce technologically. In this case, if an account has an expired password that has not been changed 35 days after expiration, the account will be disabled. If a user does not have a password (e.g., he or she only authenticates with SSH keys), then there is no inherent technological mechanism for enforcement due to the nature of the software. [IA-4(e)]

3.2.5 Authenticator Management

Authenticators for users are passwords and/or SSH keys; the management of each is implementation specific. SSH keys do not expire; therefore, implementations must provide a procedure for removing invalid keys. Removing public keys from LDAP is one practical solution.

When using passwords, local and LDAP passwords provided for users should be set to change at first log on. This is the default in the SIMP-provided LDIFs. Once a user attempts to change a password, the settings in PAM and LDAP enforce complexity requirements. By default, SIMP requires 14-character passwords with at least one character from three of the four designated categories (i.e., upper case letters, lower case letters, numbers, or special characters), and no more than three consecutive characters from each category. [IA-5, IA-5(1), IA-5(4)] Password ageing and history is enforced through a combination of PAM and LDAP. By default, the previous 24 passwords cannot be reused. [IA-5(1)(e)]
There are a number of default passwords in SIMP that are required for installation. Each implementation requires the user to change the default passwords and protect the new passwords. In addition, there are embedded passwords within the SIMP system that are used due to a lack of software-supported alternatives.

3.2.6 Access Control

This section describes the various levels of access control, including account management, access enforcement, information flow enforcement, separation of duties, least privilege, session controls, permitted actions without identification and authentication, security attributes, and remote access.

3.2.7 Account Management

Account management procedures should be created and maintained for each implementation of SIMP. The procedures should include the information listed in NIST 800-53 control AC-2. SIMP has the mechanisms in place to enforce most account management policies. The mechanisms for account management have several default settings including:

- Central account management using OpenLDAP. [AC-2(1)]
- Password expiration. Local accounts expire 35 days after password expiration. [AC-2(3)] LDAP accounts do not expire automatically due to inactivity; implementations should audit LDAP accounts regularly.
- Auditing of administrative actions to capture local account creation and modifications to LDAP accounts is done via the /var/log/slapd_audit.log file for ldap accounts and /var/log/audit.log for local accounts. [AC-2(4)]
- Shell session timeouts after 15 minutes of inactivity. [AC-2(5)] This can be circumvented by running a command that opens an endless pipe such as /bin/cat. However, this command cannot be enforced more heavily due to the high likelihood of breaking system applications. If the optional gnome module is used, the GNOME screen saver will lock the screen after 15 minutes of inactivity.
- Assignment of users into groups locally or centrally via LDAP. [AC-2(7)] By default, SIMP will have an administrators groups that has the ability to run sudo. Implementations should further define administrators or user groups and limit them with the Puppet sudo class.

3.2.8 Access Enforcement

SIMP uses the implementation of Discretionary Access Control (DAC) that is native to Linux. Specific file permissions have been assigned based on published security guidance for Red Hat, CentOS, and UNIX.

Default permissions on files created by users are enforced with user file access mask settings (using the umask command) that allow only the owner to read and write to the file. Implementations may further extend the access control in UNIX by restricting access to application files or using the file Access Control List (ACL) commands getfacl and setacl. Users of SIMP should not change file permissions on operating system files as it may decrease the overall security of the system. If a group needs access to a particular file or directory, use the setfacl command to allow the necessary access without lessening the permissions on the system. [AC-3]

3.2.9 Information Flow Enforcement

IPtables on each SIMP system is controlled by the IPtables Puppet module. When developing a new module, the IPtables rules needed for an application should be included with the module by calling the appropriate methods from the IPtables module. The end result should be a running IPtables rule set that includes the default SIMP rules and any rules needed for applications. The default communications allowed are included in Default Server Ports and Default Client Ports. [AC-4]
Default Server Ports

Default Client Ports

3.2.10 Separation of Duties

SIMP enforces separation of duties using account groups. Groups are created with each implementation to separate roles or duties properly. The SIMP team recommends that this management be done using posixGroups in LDAP for full operating System support. [AC-5]

3.2.11 Least Privilege

SIMP does not allow root to directly SSH into a system. The root user must be at a console (or at a virtual instance of the physical console) to log on. Otherwise, users must log on as themselves and perform privileged commands using sudo or sudosh. [AC-6]

NIST 800-53 least privilege security controls give people access to objects only as needed. SIMP provides only the needed software, services, and ports to allow the system to be functional and scalable. The system then relies on a given implementation to perform proper account management and user role assignments. [AC-6]

3.2.12 Session Controls

SIMP provides a number of security features for sessions. These features include:

- Accounts are locked after five invalid log on attempts over a 15-minute period. The account is then locked for 15 minutes. No administrator action is required to unlock an account. [AC-7]
- System banners are presented to a user both before and after logging on. The default banner should be customized for each implementation. [AC-8]
- After a successful log on, the date, time, and source of the last log on is presented to the user. The number of failed log on attempts since the last log on is also provided. [AC-9 and AC-9(1)]
- A limit of 10 concurrent SSH sessions are allowed per user. This can be further limited if an implementation decides it is set too high. Given the way SSH is used in operational settings, this default value is reasonable. [AC-10]
- Session lock only applies if the windowmanager::gnome module is used. Sessions lock automatically after 15 minutes of inactivity. Users must authenticate their access with valid credentials to reestablish a session. [AC-11]

3.2.13 Permitted Actions without Identification and Authentication

SIMP has a number of applications that do not require both identification and authentication. These services are listed below along with an explanation of why these aspects are not required. Implementations should include any additional services that do require identification and/or authentication. [AC-14]

Table: Actions Without Identification and Authentication

3.2.14 Security Attributes

SELinux is now available in SIMP. SELinux is an implementation of mandatory access control. It can be set to enforcing mode during the SIMP configuration or turned on at a later time. All of the SIMP packaged modules have been designed to work with SELinux set to enforcing. [AC-16]
3.2.15 Remote Access

Remote access in SIMP is performed over SSH, specifically using the OpenSSH software. OpenSSH provides both confidentiality and integrity of remote access sessions. The SSH IPtables rules allow connections from any host. SSH relies on other Linux mechanisms to provide identification and authentication of a user. As discussed in the auditing section, user actions are audited with the audit daemon and sudosh. [AC-17]

3.2.16 Systems and Communications Protection

The following sections provide information regarding application partitioning, shared resources, and various levels of protection for systems and communications.

3.2.17 User and Administration Application Separation (Application Partitioning)

SIMP can be used in a variety of ways. The most common is a platform for hosting other services or applications. In that case, there are only administrative users present. Users with accounts will be considered as a type of privileged user.

SIMP can also be used as a platform for workstations or general users performing non-administrative activities. In both cases, general users with accounts on an individual host are allowed access to the host using the `pam::access` module, so long as they have an account on the target host. No user may perform or have access to administrative functions unless given sudo or sudosh privileges via Puppet.

3.2.18 Shared Resources

There are several layers of access control that prevent the unauthorized sharing of resources in SIMP. Account access, operating system DAC settings, and the use of PKI collectively prevent resources from being shared in ways that were not intended. [SC-4]

3.2.19 Denial of Service Protection

SIMP has limited ability to prevent or limit the effects of Denial of Service (DoS) attacks. The primary measures in place are to drop improperly formatted packets using IPtables and Kernel configurations such as syncookies. [SC-5]

3.2.20 Boundary Protection

SIMP does not provide boundary protection. [SC-7]

3.2.21 Transmission Security

SIMP traffic is protected with protocols that provide confidentiality and integrity of data while in transit. The tables in Information Flow Enforcement describe the protocols used to encrypt traffic and explain the protocols that cannot be protected at the transmission layer. SSH, SSL, and TLS all provide data transmission integrity and confidentiality. The software that controls them on Red Hat and CentOS are OpenSSH and OpenSSL. The SIMP team takes industry guidance into consideration when configuring these services. For example, the list the cryptographic ciphers available is limited to the highest ciphers that SIMP needs. All others are removed. [SC-8, SC-9, SC-23, SC-7]
3.2.22 Single User Mode

SIMP systems have a password requirement for single user mode. In the event maintenance needs to be performed at a system console, users must be in possession of the root password before they can be authenticated. Grub passwords are also set to prevent unauthorized modifications to boot parameters. [SC-24]

3.2.23 PKI and Cryptography

SIMP has two native certificate authorities. The first is known as Fake CA. A local certificate authority is used to create properly formed server certificates if an implementation does not have other means of obtaining them. Many SIMP services require certificates; therefore, SIMP provides this tool for testing or for situations where other certificates are not available. The second certificate authority, Puppet CA, is built into Puppet. Puppet creates, distributes, and manages certificates that are specifically for Puppet. More information on the Puppet CA can be found in the Puppet Labs security documentation. [SC-17, SC-13]

Warning
Fake CA certificates should not be used in an operational setting.

3.2.24 Mobile Code

SIMP does not use mobile code; however, there are not any particular tools that will prevent its use. [SC-18]

3.2.25 Protection of Information at Rest

There are no additional protections for information at rest beyond operating system capabilities in SIMP. There are also no measures in place to encrypt or sign data before transmission. Each implementation should determine how to further protect information at rest. [SC-28]

3.2.26 Audit and Accountability

This section discusses the content, storage, and protection of auditable events.

3.2.27 Auditable Events

Audidt and rsyslog provide the foundation for SIMP auditing. Audidt performs the majority of the security-related events; however, other Linux logs also have security information in them, which are captured using rsyslog.

The default auditable events for SIMP were developed based on several industry best practices including those from the SCAP Security Guide and several government configuration guides. The suggested rules by those guides were fine-tuned so the audit daemon would not fill logs with useless records or reduce performance. These guides should be referenced for a detailed explanation of why rules are applied. Additional justification can be found in the comments of the SIMP audit rules found in the appendix of this guide. [AU-2]

The SIMP development team reviews every release of the major security guides for updated auditable events suggestions. Each of those suggestions is reviewed and applied if deemed applicable. [AU-2(3)] Privileged commands are audited as part of the SIMP auditing configuration. This is accomplished by monitoring sudo commands with auditd. Keystrokes for administrators that use sudosh are also logged. Each session can be replayed using sudosh-replay. [AU-2(4)]
3.2.28 Content of Audit Records

Audit records capture the following information [AU-3]:

- Date and Time
- UID and GID of the user performing the action
- Command
- Event ID
- Key
- Node Hostname/IP Address
- Login Session ID
- Executable

3.2.29 Audit Storage

Audit logs are stored locally on a separate partition in the `/var/log` directory. The size of this partition is configurable. Other default audit storage configurations include:

- A syslog log is written when the audit partition has 75MB free. (This can be changed to e-mail, if e-mail infrastructure is in place.) [AU-5(a), AU-5(1)]
- The log file rotates once it reaches 30MB.

3.2.30 Audit Reduction and Response

SIMP provides a means to capture the proper information for audit records and stores them centrally. Each implementation must decide and document how it reduces, analyzes, and responds to audit events. [AU-5]

Auditd, like all services in SIMP, is controlled by Puppet. Stopping the service without disabling Puppet means the service will always be started automatically during a Puppet run. The files that control the audit configuration will also revert to their original state if changed manually on a client node. In the event auditd fails, the system will continue to operate. Several security guides have suggested that the system should shut down if auditd fails for any reason. However, SIMP will not shut down, but will provide an alert via syslog when this happens. [AU-5(1)]

SIMP also comes with an optional module for the Elasticsearch/Logstash/Kibana (ELK) stack. These three open source tools can be combined to parse, index, and visualize logs. There are also SIMP provided dashboards for the Kibana web interface. Implementations can build their own dashboards to meet local security or functional needs for log reduction and management. [AU-6]

3.2.31 Protection of Audit Information

The primary means of protecting the audit logs is through the use of file permissions. Audit records are stored in the `/var/log` directory and can only be accessed by `root`. Audit logs are rotated off daily if the implementation has not developed a way of offloading the logs to another location where they can be backed up. Lastly, if the `rsyslog::stock::log_server` module is implemented, logs are transmitted to the log server over a TLS protected link.
3.2.32 Time Synchronization

Each SIMP client (including the Puppet Master) has NTPD enabled by default. Part of the installation directs the clients to a time server. If no servers are available, the SIMP clients can use the Puppet Master as the central time source. Audit logs receive their time stamp from the local server’s system clock; therefore, the SIMP client must be connected to a central time source for time stamps in audit logs to be accurate.

3.3 Operational Security

This chapter contains SIMP security concepts that are related to the operational security controls in NIST 800-53.

3.3.1 Configuration Management

This section describes the management of various configurations within SIMP.

Baseline Configurations

SIMP baselines include configuration settings and Puppet modules. Currently, baselines are maintained for both Red Hat/CentOS 6.x, and Red Hat/CentOS 7.x. Each configuration item that is managed by a Puppet module has an RPM installed on the Puppet Master in the form of `pupmod-name-x.x.x-x`. This process allows for one main SIMP baseline to be maintained and modules to be upgraded easily. An overall SIMP RPM is also installed on the Puppet Master, which denotes the version number of SIMP that is installed. [CM-2, CM-2(2), CM-2(3), CM-6]

SIMP installs a minimal set of RPMs, which can be found in ?. RPMs, services, and IPtables rules all use a deny-all, but allow-by-exception module. Additional RPMs must be installed by each implementation. Services must be declared explicitly or they will be disabled by Puppet; IPtables rules must allow a service explicitly. [CM-2(5)]

Managing Configuration Changes

Configuration change approvals are managed by each implementation; SIMP only provides the mechanisms to apply changes on clients. A combination of Puppet, rsync, and YUM is used to apply those changes across all (or selected) Puppet clients. All changes made are audited with auditd or are logged to other files via syslog. [CM-3(a), CM-3(3)]

UNIX systems are made up of hundreds of configuration files that can contain dozens of settings. SIMP does not make an attempt to manage all of the settings in every file. Instead, critical operating system files or files that need to be controlled centrally are managed. Implementations can manage additional files if they are deemed necessary. [CM-6]

Security Verification and Flaw Remediation

SIMP cannot detect flaws automatically; each implementation is responsible for tracking flaws. However, SIMP provides a way for flaws to be fixed across all clients. One or all of the following can help automate flaw remediation [CM-6, SI-2, SI-2(1), SI-2(4)]:

- **Puppet**: Apply a configuration change to files that are managed by Puppet.
- **rsync**: Use this mechanism to deliver a file to a client. This can be used with or without Puppet to synchronize files.
- **YUM**: Update packages nightly with YUM. Placing an updated package in YUM and running a YUM update manually, or allowing time for the cron job to run, will ensure packages on all clients are updated. Otherwise, a cron job will perform a daily update of packages with YUM.
- **PSSH**: Allow commands to run across a set of nodes with the PSSH utility. Through the use of keys, this becomes a powerful way to run a one-time operation against a large number of nodes.

The extent of security verification that is performed currently is based on changes to files that Puppet or the Advanced Intrusion Detection Environment (AIDE) provides. There are also Security Content Automation Protocol (SCAP) profiles available from the SCAP-Security-Guide project that check security configuration settings. [SI-6]

**Malicious Code Protection**

For most environments, SIMP will use ClamAV to protect against malicious code. Rsync is used to push out new definitions, which should be updated by the local administrator regularly. SIMP also comes with a `mcafee:uvscan` module that manages an installation of uvscan, if it is preferred. The module can configure `.dat` file updates to occur over rsync.

Both the ClamAV and McAfee modules provide a method to run a scan via cron on a customer scheduled basis. [SI-3] SIMP also comes with the `chkrootkit` tool to check for `rootkits`. The tool runs as a cron job and places its output into syslog. [SI-3]

**Software and Information Integrity**

Unauthorized changes to a local client can be detected by Puppet or AIDE (for any file managed by Puppet). In the event that a managed file is changed locally, Puppet will revert the file back to its original state. It is important to note that this is a function of Puppet and is intended to be more of a configuration management feature rather than a security feature. If a Puppet client has been compromised, the Puppet Master may not have the ability to retake control over that client. However, the Puppet Master can configure all other nodes to deny traffic from the compromised node if they are configured by the administrator to do so. There are additional configuration files that are checked by AIDE, which is triggered by a cron job. AIDE logs any detected file changes in syslog. Each implementation may add additional files that are managed by Puppet or watched by AIDE. The AIDE baseline database is updated periodically to handle the installation and updating of system RPMs and reduce false positives. [SI-7, SI-7(1), SI-7(2), SI-7(3)]

### 3.3.2 Remote Maintenance

Remote maintenance can be performed on SIMP using SSH. Local maintenance can be performed at the console or via serial port (if available). SSH sessions are tracked and logged using the security features built into SIMP. Console access requires someone to have access to the physical (or virtual) console along with the `root` password. Auditing of those actions also occurs in accordance with the configured audit policy. It is up to the implementation to decide how to distribute authentication information for remote maintenance. [MA-4, MA-4(1), MA-6]

### 3.3.3 Incident Response

While Puppet is not intended to be a security product primarily, its features help provide security functionality such as dynamic reconfigurations and wide-scale consistent mitigation application. If an implementation chooses, they can leverage Puppet’s ability to reconfigure systems as part of incident response. [IR]

### 3.3.4 Contingency Planning

SIMP does not provide any direct support for contingency planning. Some of the mechanisms provided by SIMP might be used to support an implementation’s contingency plan.
3.3.5 System Backup

SIMP comes with a module called backuppc. This module provides a base configuration of the BackupPC software and allows Puppet servers and clients to perform backups.

3.4 Information System Management

This chapter contains SIMP security concepts that are related to the management security controls in NIST 800-53.

3.4.1 Risk Assessment

This section describes the process of identifying risks within a system.

3.4.2 SIMP Self Risk Assessment

Risk can be found in any system. The SIMP team is constantly evaluating the system and the settings to minimize inherit risk. Most risks can be mitigated by processes and procedures at the implementation level. The following table describes the known areas in SIMP. [RA-1]

| Table: SIMP Risk |

3.4.3 Vulnerability Scanning

The SIMP development and security team performs regular vulnerability scanning of the product using commercial and open source tools. Results and mitigations for findings from those tools can be provided upon request. [CA-2, RA-5]

3.4.4 Security Assessment and Authorization

Assessment and authorization varies by implementation. Implementations are encouraged to use documentation artifacts provided by the SIMP team to assist with assessment and authorization. [CA-2]

3.5 Security Concepts Appendices

3.5.1 Default Files Watched by AIDE

```
/boot NORMAL
/bin NORMAL
/sbin NORMAL
/lib NORMAL
/opt NORMAL
/usr NORMAL
/root NORMAL
!/usr/src
!/usr/tmp
/etc PERMS
!/etc/mtab
!/etc/.~
```
3.5. Security Concepts Appendices

/etc/exports NORMAL
/etc/fstab NORMAL
/etc/passwd NORMAL
/etc/group NORMAL
/etc/gshadow NORMAL
/etc/shadow NORMAL
/etc/security/opasswd NORMAL
/etc/hosts.allow NORMAL
/etc/hosts.deny NORMAL
/etc/sudoers NORMAL
/etc/skel NORMAL
/etc/logrotate.d NORMAL
/etc/resolv.conf DATAONLY
/etc/nscd.conf NORMAL
/etc/securetty NORMAL
/etc/profile NORMAL
/etc/bashrc NORMAL
/etc/bash_completion.d/ NORMAL
/etc/login.defs NORMAL
/etc/zprofile NORMAL
/etc/zshrc NORMAL
/etc/zlogin NORMAL
/etc/zlogout NORMAL
/etc/profile.d/ NORMAL
/etc/X11/ NORMAL
/etc/yum.conf NORMAL
/etc/yumex.conf NORMAL
/etc/yumex.profiles.conf NORMAL
/etc/yum/ NORMAL
/etc/yum.repos.d/ NORMAL
/var/log LOG
!/var/log/sa
!/var/log/aide/aide.log
!/var/log/aide/aide.report
/etc/audit/ LSPP
/etc/libaudit.conf LSPP
/usr/sbin/stunnel LSPP
/var/spool/at LSPP
/etc/at.allow LSPP
/etc/at.deny LSPP
/etc/cron.allow LSPP
/etc/cron.d/ LSPP
/etc/cron.daily/ LSPP
/etc/cron.hourly/ LSPP
/etc/cron.monthly/ LSPP
/etc/cron.weekly/ LSPP
/etc/crontab LSPP
/var/spool/cron/root LSPP
/etc/login.defs LSPP
/etc/securetty LSPP
/var/log/faillog LSPP
/var/log/lastlog LSPP
/etc/hosts LSPP
/etc/sysconfig LSPP
/etc/inittab LSPP
/etc/init SPP
/etc/grub LSPP
/etc/rc.d LSPP
3.5.2 Audit Rules

## For audit 1.6.5 and higher
##
# Ignore errors
# This may sound counterintuitive, but we'd rather skip bad rules and load the
# rest than miss half the file. Warnings are still logged in the daemon
# restart output.
-1

## Remove any existing rules
-D

## Continue loading rules on failure.
# Particularly with the automatically generated nature of these rules in
# Puppet, it is possible that one or more may fail to load. We want to continue
# in that case so that we audit as much as possible.
-c

## Increase buffer size to handle the increased number of messages.
## Feel free to increase this if the machine panic's
# Default: 8192
-b 16394

## Set failure mode to panic
# Default: 2
-f 2

## Rate limit messages
# Default: 0
# If you set this to non-zero, you almost definitely want to set -f to 1 above.
-r 0

## Get rid of all anonymous and daemon junk. It clogs up the logs and doesn't
# do anyone # any good.
-a exit,never -F auid!=4294967295
# Ignore system services. In most guides this is tagged onto every rule but
# that just makes for more processing time.
-a exit,never -F auid!=0 -F auid<500

## unsuccessful file operations
# CCE-26712-0
# CCE-26651-0
-a always,exit -F arch=b64 -S creat -S mkdir -S mkod -S link -S symlink -S mkdirat -S mkodat -S link -S symlink -S openat -S open -S close -S rename -S truncate -S ftruncate -S rmdir -S unlink -S unlinkat -F exit=-EACCES -k access
-a always,exit -F arch=b64 -S creat -S mkdir -S mkod -S link -S symlink -S mkdirat -S mkodat -S link -S symlink -S openat -S open -S close -S rename -S truncate -S ftruncate -S rmdir -S unlink -S unlinkat -F exit=-EPERM -k access
-a always,exit -F arch=b32 -S creat -S mkdir -S mkod -S link -S symlink -S mkdirat -S mkodat -S link -S symlink -S openat -S open -S close -S rename -S truncate -S ftruncate -S rmdir -S unlink -S unlinkat -F exit=-EACCES -k access
-a always,exit -F arch=b32 -S creat -S mkdir -S mkod -S link -S symlink -S mkdirat -S mkodat -S link -S symlink -S openat -S open -S close -S rename -S truncate -S ftruncate -S rmdir -S unlink -S unlinkat -F exit=-EPERM -k access

# Permissions auditing
# CCE-26280-8
# CCE-27173-4
# CCE-27174-2
# CCE-27175-9
# CCE-27177-5
# CCE-27178-3
# CCE-27179-1
# CCE-27180-9
# CCE-27181-7
# CCE-27182-5
# CCE-27183-3
# CCE-27184-1
# CCE-27185-8
-a always,exit -F arch=b64 -S chown -S fchmod -S fchmodat -S fchown -S fchownat -S lchown -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S fremovexattr -k perm_mod
-a always,exit -F arch=b32 -S chown -S fchmod -S fchmodat -S fchown -S fchownat -S lchown -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S fremovexattr -k perm_mod

# Audit useful items that someone does when su'ing to root.
# Had to add an entry at the top for getting rid of anonymous records. They
# are only moderately useful and contain *way* too much noise since this covers
# things like cron as well.
-a always,exit -F arch=b64 -F auid!=0 -F uid=0 -S capset -S mknod -S pivot_root -S quotactl -S setsid -S setuid -S swapoff -S swapon -k su-root-activity
-a always,exit -F arch=b32 -F auid!=0 -F uid=0 -S capset -S mknod -S pivot_root -S quotactl -S setsid -S setuid -S swapoff -S swapon -k su-root-activity

# Audit the execution of suid and sgid binaries.
# CCE-26457-2
-a always,exit -F arch=b64 -F euid=0 -F uid!=0 -S execve -k suid-root-exec
-a always,exit -F arch=b32 -F euid=0 -F uid!=0 -S execve -k suid-root-exec

## Audit the loading and unloading of kernel modules.
# CCE-26611-4
-w /sbin/insmod -p x -k modules
-w /sbin/rmmod -p x -k modules
-w /sbin/modprobe -p x -k modules
-a always,exit -F arch=b64 -S init_module -S delete_module -k modules
-a always,exit -F arch=b32 -S init_module -S delete_module -k modules

## Things that could affect time
# CCE-27172-6
# CCE-27203-9
# CCE-27169-2
# CCE-27170-0

3.5. Security Concepts Appendices 109
-a exit,always -F arch=b32 -S adjtimex -S stime -S clock_settime -S settimeofday -k audit_time_rules
- a exit,always -F arch=b64 -S adjtimex -S clock_settime -S settimeofday -k audit_time_rules

# CCE-27172-6
-w /etc/localtime -p wa -k audit_time_rules

## Things that could affect system locale
# CCE-26648-6
- a always,exit -F arch=b32 -S sethostname -S setdomainname -k audit_network_modifications
- a always,exit -F arch=b64 -S sethostname -S setdomainname -k audit_network_modifications
-w /etc/issue -p wa -k audit_network_modifications
-w /etc/issue.net -p wa -k audit_network_modifications
-w /etc/hosts -p wa -k audit_network_modifications
-w /etc/sysconfig/network -p wa -k audit_network_modifications

# Mount options.
# CCE-26573-6
- a always,exit -F arch=b32 -S mount -S umount -S umount2 -k mount
- a always,exit -F arch=b64 -S mount -S umount -S umount2 -k mount

# Audit umask changes.
# This is uselessly noisy.
# -a exit,always -S umask -k umask

# CCE-26664-3
-w /etc/group -p wa -k audit_account_changes
-w /etc/group- -p wa -k audit_account_changes
-w /etc/passwd -p wa -k audit_account_changes
-w /etc/passwd- -p wa -k audit_account_changes
-w /etc/gshadow -p wa -k audit_account_changes
-w /etc/shadow -p wa -k audit_account_changes
-w /etc/shadow- -p wa -k audit_account_changes
-w /etc/security/opasswd -p wa -k audit_account_changes

# CCE-26657-7
-w /etc/selinux/ -p wa -k MAC-policy

# CCE-26691-6
-w /var/log/faillog -p wa -k logins
-w /var/log/lastlog -p wa -k logins

# CCE-26610-6
-w /var/run/utmp -p wa -k session
-w /var/run/btmp -p wa -k session
-w /var/run/wtmp -p wa -k session

# CCE-26662-7
-w /etc/sudoers -p wa -k CFG_sys

# Generally good things to audit.
-w /var/spool/at -p wa -k CFG_sys
-w /etc/at.deny -p wa -k CFG_sys
-w /etc/cron.deny -p wa -k CFG_cron
-w /etc/cron.d -p wa -k CFG_cron
-w /etc/cron.daily -p wa -k CFG_cron
-w /etc/cron.hourly -p wa -k CFG_cron
-w /etc/cron.monthly -p wa -k CFG_cron
-w /etc/cron.weekly -p wa -k CFG_cron
3.5. Security Concepts Appendices

```bash
-w /etc/crontab -p wa -k CFG_cron
-w /etc/anacrontab -p wa -k CFG_cron
-w /etc/login.defs -p wa -k CFG_sys
-w /etc/securetty -p wa -k CFG_shell
-w /etc/profile -p wa -k CFG_shell
-w /etc/bashrc -p wa -k CFG_shell
-w /etc/csh.cshrc -p wa -k CFG_shell
-w /etc/csh.login -p wa -k CFG_shell
-w /etc/sysconfig -p wa -k CFG_sys
-w /etc/inittab -p wa -k CFG_sys
-w /etc/rc.d/init.d -p wa -k CFG_sys
-w /etc/rc.local -p wa -k CFG_sys
-w /etc/rc.sysinit -p wa -k CFG_sys
-w /etc/xinetd.d -p wa -k CFG_sys
-w /etc/ld.so.conf -p wa -k CFG_sys
-w /etc/ld.so.conf.d -p wa -k CFG_sys
-w /etc/sysctl.conf -p wa -k CFG_sys
-w /etc/modprobe.d/00_simp_blacklist.conf -p wa -k CFG_sys
-w /etc/modprobe.conf.d -p wa -k CFG_sys
-w /etc/pam.d -p wa -k CFG_pam
-w /etc/pam_smb.conf -p wa -k CFG_pam
-w /etc/aliases -p wa -k CFG_sys
-w /etc/ssh/sshd_config -p wa -k CFG_sys
-w /etc/issue -p wa -k CFG_sys
-w /etc/issue.net -p wa -k CFG_sys
-w /etc/snmp/snmpd.conf -p wa -k CFG_sys
-w /etc/resolv.conf -p wa -k CFG_sys
-w /etc/nsswitch.conf -p wa -k CFG_sys
-w /etc/hosts -p wa -k CFG_sys
-w /etc/hosts.allow -p wa -k CFG_sys
-w /etc/hosts.deny -p wa -k CFG_sys
-w /etc/yum.conf -p wa -k yum-config
-w /etc/yumrepos.d -p wa -k yum-config
-a exit,always -F arch=b32 -S ptrace -k paranoid
-a exit,always -F arch=b64 -S ptrace -k paranoid
-a always,exit -F arch=b32 -S personality -k paranoid
-a always,exit -F arch=b64 -S personality -k paranoid
-w /etc/aide.conf -p wa -k CFG_aide
-w /etc/aide.conf.d/default.aide -p wa -k CFG_aide
-w /etc/rc.d/init.d/auditd -p wa -k auditd
-w /var/log/audit.log -p wa -k audit-logs
-w /etc/pam_d LDAP.conf -p a -k CFG_etc_ldap
-w /etc/ntp.conf -p wa -k CFG_ntp
-w /etc/ntp/keys -p wa -k CFG_ntp
-w /etc/ntp/ntpservers -p wa -k CFG_ntp
-w /etc/pki/private -p wa -k PKI
-w /etc/pki/public -p wa -k PKI
-w /etc/pki/cacerts -p wa -k PKI
-w /etc/pki/private/ws69.kw.awesome.sauce.pem -p wa -k PKI
-w /etc/pki/public/ws69.kw.awesome.sauce.pub -p wa -k PKI
-w /var/log/audit.log.1 -p rwa -k audit-logs
```
3.5.3 Default Kickstart Files

Default Puppet Master Kickstart file (contains default RPMs)

```bash
# Use the following Ruby code to generate your password hashes:
# ruby -r 'digest/sha2' -e 'puts "password".crypt("$6$" + rand(36**8).to_s(36))'

# Use the following command to generate your grub password hash:
# grub2-mkpasswd-pbkdf2

# Replace the following strings in this file
# #BOOTPASS# - Your hashed bootloader password
# #ROOTPASS# - Your hashed root password
# #KSSERVER# - The IP address of your YUM server
# #YUMSERVER# - The IP address of your YUM server
# #LINUXDIST# - The LINUX Distribution you are kickstarting
# - Current CASE SENSITIVE options: RedHat CentOS

authconfig --enableshadow --passalgo=sha512
bootloader --location=mbr --append="console=ttyS1,57600 console=tty1" --iscrypted --password=#BOOTPASS#

rootpw --iscrypted #ROOTPASS#
zerombr
firewall --enabled --ssh
firstboot --disable
logging --level=info
network --bootproto= dhcp
reboot
```
selinux --permissive
timezone --utc GMT
install
skipx

%include /tmp/repo-include
text
keyboard us
lang en_US
url --url http://#KSSERVER#/yum/#LINUXDIST#/7/x86_64

%include /tmp/part-include

%packages --nobase
-sendmail
-sysklogd
acl
aide
anacron
audit
bzip2
coolkey
crontabs
cryptsetup-luks
dhclient
git
gnupg
iptables
iptables-ipv6
irqbalance
krb5-workstation
libaio
libutempter
logrotate
logwatch
lsof
lsscsi
mdadm
microcode_ctl
mutt
net-snmp
net-tools
netlabel_tools
ntp
openssh-clients
openssh-server
pam_krb5
pam_pkcs11
pciutils
psacct
quota
redhat-lsb
rpm
rsync
rsyslog
smartmontools

3.5. Security Concepts Appendices
sssd
stunnel
subversion
sudo
sysstat
tcp_wrappers
tmpwatch
unzip
usbutils
vim-enhanced
vlock
wget
which
zip
# Puppet stuff
rsync
facter
puppet

# In case of broken repo, these should be installed.
hdparm
kbd
libhugetlbfs
policycoreutils
prelink
rootfiles
selinux-policy-targeted
setserial
sysfsutils
udftools

# Don't install these
-rhn-check
-rhn-setup
-rhnsd
-subscription-manager
-yum-rhn-plugin
%end

%pre
ksserver="#KSSERVER#"
wget -O /tmp/diskdetect.sh http://$ksserver/ks/diskdetect.sh;
chmod 750 /tmp/diskdetect.sh;
	/tmp/diskdetect.sh;
wget -O /tmp/repodetect.sh http://$ksserver/ks/repodetect.sh;
chmod 750 /tmp/repodetect.sh;
	/tmp/repodetect.sh '7' $ksserver;
%end

%post
ostype="#LINUXDIST#"
if [ $ostype == "CentOS" ]; then
    sed -i '/enabled=/d' /etc/yum.repos.d/CentOS-Base.repo;
    sed -i '^[.,\]/ a\
    enabled=0' /etc/yum.repos.d/CentOS-Base.repo;
fi
ksserver="#KSSERVER#"
# Notify users that bootstrap will run on firstboot
echo "Welcome to SIMP! If this is firstboot, SIMP bootstrap is scheduled to run. If this host is not autosigned by Puppet, sign your Puppet certs to begin bootstrap. Otherwise, it should already be running! Tail /root/puppet.bootstrap.log for details. Wait for completion and reboot.

To remove this message, delete /root/.bootstrap_msg" > /root/.bootstrap_msg
sed -i "2i if [-f /root/.bootstrap_msg ]\nthen\n cat /root/.bootstrap_msg\nfi" /root/.bashrc
source /root/.bashrc

# Enable the firstboot bootstrapping script.
wget --no-check-certificate -O /etc/init.d/runpuppet http://$ksserver/ks/runpuppet;
chmod 700 /etc/rc.d/init.d/runpuppet;
chkconfig --add runpuppet;
chkconfig --level 35 runpuppet on;
%end

3.5.4 SIMP Common RPMs

CentOS RPMs

Table 3.1: CentOS RPMs

<table>
<thead>
<tr>
<th>RPM</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>apr-util-ldap-1.5.2-6.el7.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>apr-util-mysql-1.5.2-6.el7.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>apr-util-nss-1.5.2-6.el7.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>apr-util-odbc-1.5.2-6.el7.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>apr-util-openssl-1.5.2-6.el7.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>apr-util-pgsql-1.5.2-6.el7.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>apr-util-sqlite-1.5.2-6.el7.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>bash-4.2.45-5.el7_0.x86_64.rpm</td>
<td>CentOS Updates</td>
</tr>
<tr>
<td>bash-doc-4.2.45-5.el7_0.x86_64.rpm</td>
<td>CentOS Updates</td>
</tr>
<tr>
<td>httpd-2.4.6-18.el7.centos.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>httpd-devel-2.4.6-18.el7.centos.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>httpd-manual-2.4.6-18.el7.centos.noarch.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>httpd-tools-2.4.6-18.el7.centos.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>mod_ldap-2.4.6-18.el7.centos.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>mod_proxy_html-2.4.6-18.el7.centos.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>mod_session-2.4.6-18.el7.centos.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>mod_ssl-2.4.6-18.el7.centos.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>mod_wsgi-3.4-12.el7_0.x86_64.rpm</td>
<td>CentOS Everything Disc</td>
</tr>
<tr>
<td>openssh-ldap-6.4p1-8.el7.x86_64.rpm</td>
<td>CentOS DVD</td>
</tr>
<tr>
<td>ruby-2.0.0.353-20.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>ruby-doc-2.0.0.353-20.el7.noarch.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>ruby-irb-2.0.0.353-20.el7.noarch.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>ruby-libs-2.0.0.353-20.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>ruby-tcltk-2.0.0.353-20.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>selinux-policy-3.12.1-153.el7_0.13.noarch.rpm</td>
<td>CentOS Updates</td>
</tr>
<tr>
<td>selinux-policy-devel-3.12.1-153.el7_0.13.noarch.rpm</td>
<td>CentOS Updates</td>
</tr>
<tr>
<td>selinux-policy-doc-3.12.1-153.el7_0.13.noarch.rpm</td>
<td>CentOS Updates</td>
</tr>
</tbody>
</table>

Continued on next page

3.5. Security Concepts Appendices
### Table 3.1 – continued from previous page

<table>
<thead>
<tr>
<th>RPM</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>selinux-policy-minimum-3.12.1-153.el7_0.13.noarch.rpm</td>
<td>CentOS Updates</td>
</tr>
<tr>
<td>selinux-policy-mls-3.12.1-153.el7_0.13.noarch.rpm</td>
<td>CentOS Updates</td>
</tr>
<tr>
<td>selinux-policy-sandbox-3.12.1-153.el7_0.13.noarch.rpm</td>
<td>CentOS Updates</td>
</tr>
<tr>
<td>selinux-policy-targeted-3.12.1-153.el7_0.13.noarch.rpm</td>
<td>CentOS Updates</td>
</tr>
<tr>
<td>sendmail-milter-8.14.7-4.el7.x86_64.rpm</td>
<td>CentOS DVD</td>
</tr>
</tbody>
</table>

### Common RPMs

<table>
<thead>
<tr>
<th>RPM</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>chkrootkit-0.50-4.el7.x86_64.rpm</td>
<td><a href="https://dl.fedoraproject.org/pub/fedora/linux/updates/20/SRPMS/chkrootkit-0.50-4.fc20.src.rpm">https://dl.fedoraproject.org/pub/fedora/linux/updates/20/SRPMS/chkrootkit-0.50-4.fc20.src.rpm</a></td>
</tr>
<tr>
<td>clamav-0.98.4-1.el7.x86_64.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-data-0.98.4-1.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-data-empty-0.98.4-1.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-db-0.97.6-1.el6.x86_64.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-filesystem-0.96-1.x86_64.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-filesystem-0.98.4-1.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-lib-0.98.4-1.el7.x86_64.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-milter-0.98.4-1.el7.x86_64.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-milter-systemd-0.98.4-1.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-scanner-0.98.4-1.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-scanner-systemd-0.98.4-1.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-server-0.98.4-1.el7.x86_64.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-server-systemd-0.98.4-1.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-server-sysvinit-0.98.4-1.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-unofficial-sigs-3.7.1-6.el6.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>clamav-update-0.98.4-1.el7.x86_64.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>elasticsearch-1.0.1.noarch.rpm</td>
<td><a href="https://download.elasticsearch.org/elasticsearch/elasticsearch/elasticsearch-1.0.1.noarch.rpm">https://download.elasticsearch.org/elasticsearch/elasticsearch/elasticsearch-1.0.1.noarch.rpm</a></td>
</tr>
<tr>
<td>elasticsearch-1.3.2.noarch.rpm</td>
<td><a href="https://download.elasticsearch.org/elasticsearch/elasticsearch/elasticsearch-1.3.2.noarch.rpm">https://download.elasticsearch.org/elasticsearch/elasticsearch/elasticsearch-1.3.2.noarch.rpm</a></td>
</tr>
<tr>
<td>elasticsearch-curator-1.1.1-0.el7.noarch.rpm</td>
<td><a href="https://github.com/onyxpoint/curator">https://github.com/onyxpoint/curator</a></td>
</tr>
<tr>
<td>es2unix-1.6.1-0.el7.noarch.rpm</td>
<td><a href="https://github.com/onyxpoint/es2unix">https://github.com/onyxpoint/es2unix</a></td>
</tr>
<tr>
<td>facter-2.2.0-1.el7.x86_64.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7Server/products/x86_64/facter-2.2.0-1.el7.x86_64.rpm">http://yum.puppetlabs.com/el/7Server/products/x86_64/facter-2.2.0-1.el7.x86_64.rpm</a></td>
</tr>
<tr>
<td>gweb-2.1.8-1.noarch.rpm</td>
<td><a href="http://sourceforge.net/projects/ganglia/">http://sourceforge.net/projects/ganglia/</a></td>
</tr>
<tr>
<td>incron-0.5.10-8.el7.x86_64.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>jenkins-1.474-1.1.noarch.rpm</td>
<td><a href="http://pkg.jenkins-ci.org/redhat/">http://pkg.jenkins-ci.org/redhat/</a></td>
</tr>
<tr>
<td>kibana-3.1.0.SIMP-0.noarch.rpm</td>
<td><a href="https://download.elasticsearch.org/kibana/kibana/kibana-3.1.0.tar.gz">https://download.elasticsearch.org/kibana/kibana/kibana-3.1.0.tar.gz</a></td>
</tr>
<tr>
<td>libconfuse-2.7-7.el7.x86_64.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/7/x86_64/repo_view/libconfuse.html">http://dl.fedoraproject.org/pub/epel/7/x86_64/repo_view/libconfuse.html</a></td>
</tr>
<tr>
<td>libev-4.15-3.el7.x86_64.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/libev-4.15-3.el7.x86_64.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/libev-4.15-3.el7.x86_64.rpm</a></td>
</tr>
<tr>
<td>libnfnetlink-1.0.1-2.el7.x86_64.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/libnfnetlink-1.0.1-2.el7.x86_64.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/libnfnetlink-1.0.1-2.el7.x86_64.rpm</a></td>
</tr>
<tr>
<td>libsepol-2.2.2-6.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>libsepol-2.2.2-6.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>libsepol-2.2.2-6.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>libsepol-2.2.2-6.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>libsepol-2.2.2-6.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
</tbody>
</table>
### Table 3.2 – continued from previous page

<table>
<thead>
<tr>
<th>RPM</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>libsepol-static-2.1.9-3.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>libyaml-0.1.4-10.el7.x86_64.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/7/SRPMS/libyaml-0.1.4-10.el7.x86_64.rpm">http://dl.fedoraproject.org/pub/epel/7/SRPMS/libyaml-0.1.4-10.el7.x86_64.rpm</a></td>
</tr>
<tr>
<td>logstash-1.4.2-1.2-0f5b51.noarch.rpm</td>
<td><a href="https://download.elasticsearch.org/logstash/logstash/packages/centos/logstash-1.4.2-1.2-0f5b51.noarch.rpm">https://download.elasticsearch.org/logstash/logstash/packages/centos/logstash-1.4.2-1.2-0f5b51.noarch.rpm</a></td>
</tr>
<tr>
<td>logstash-contrib-1.4.2-1_ebd73ef.noarch.rpm</td>
<td><a href="https://download.elasticsearch.org/logstash/logstash/packages/centos/logstash-contrib-1.4.2-1_ebd73ef.noarch.rpm">https://download.elasticsearch.org/logstash/logstash/packages/centos/logstash-contrib-1.4.2-1_ebd73ef.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-2.6.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-2.6.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-2.6.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-2.7.0-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-2.7.0-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-2.7.0-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-actionpolicy-auth-2.1.0-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-actionpolicy-auth-2.1.0-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-actionpolicy-auth-2.1.0-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-client-2.7.0-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-client-2.7.0-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-client-2.7.0-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-common-2.7.0-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-common-2.7.0-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-common-2.7.0-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-filemgr-agent-1.0.2-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-filemgr-agent-1.0.2-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-filemgr-agent-1.0.2-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-filemgr-client-1.0.2-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-filemgr-client-1.0.2-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-filemgr-client-1.0.2-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-filemgr-common-1.0.2-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-filemgr-common-1.0.2-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-filemgr-common-1.0.2-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-iptables-agent-3.0.2-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-iptables-agent-3.0.2-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-iptables-agent-3.0.2-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-iptables-client-3.0.2-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-iptables-client-3.0.2-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-iptables-client-3.0.2-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-iptables-common-3.0.2-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-iptables-common-3.0.2-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-iptables-common-3.0.2-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-nettest-agent-3.0.4-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nettest-agent-3.0.4-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nettest-agent-3.0.4-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-nettest-client-3.0.4-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nettest-client-3.0.4-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nettest-client-3.0.4-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-nettest-common-3.0.4-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nettest-common-3.0.4-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nettest-common-3.0.4-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-nrpe-agent-3.0.3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nrpe-agent-3.0.3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nrpe-agent-3.0.3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-nrpe-client-3.0.3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nrpe-client-3.0.3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nrpe-client-3.0.3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-nrpe-common-3.0.3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nrpe-common-3.0.3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-nrpe-common-3.0.3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-package-agent-4.3.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-package-agent-4.3.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-package-agent-4.3.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-package-client-4.3.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-package-client-4.3.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-package-client-4.3.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-package-common-4.3.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-package-common-4.3.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-package-common-4.3.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-puppet-agent-1.9.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-agent-1.9.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-agent-1.9.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-puppet-client-1.9.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-client-1.9.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-client-1.9.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-puppet-client-1.9.3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-client-1.9.3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-client-1.9.3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-puppet-common-1.9.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-common-1.9.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-common-1.9.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-puppet-common-1.9.3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-common-1.9.3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-puppet-common-1.9.3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-service-agent-3.1.3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-service-agent-3.1.3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-service-agent-3.1.3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-service-client-3.1.3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-service-client-3.1.3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-service-client-3.1.3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-service-common-3.1.3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-service-common-3.1.3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-service-common-3.1.3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-shell-agent-0.0.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-shell-agent-0.0.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-shell-agent-0.0.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-shell-client-0.0.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-shell-client-0.0.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-shell-client-0.0.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mcollective-shell-common-0.0.1-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-shell-common-0.0.1-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/Server/products/x86_64/mcollective-shell-common-0.0.1-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>mod_passenger-4.0.41-1.el7.x86_64.rpm</td>
<td><a href="https://github.com/plusion/passenger">https://github.com/plusion/passenger</a></td>
</tr>
<tr>
<td>mysql-connector-python-1.1.6-1.el7.noarch.rpm</td>
<td>EPEL</td>
</tr>
<tr>
<td>passenger-service-1.0.0-1.el7.noarch.rpm</td>
<td>SIMP Custom</td>
</tr>
<tr>
<td>pdsh-2.2.9-1el7.x86_64.rpm</td>
<td><a href="https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2">https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2</a></td>
</tr>
<tr>
<td>pdsh-debuginfo-2.2.9-1el7.x86_64.rpm</td>
<td><a href="https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2">https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2</a></td>
</tr>
<tr>
<td>pdsh-mod-dshgroup-2.2.9-1el7.x86_64.rpm</td>
<td><a href="https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2">https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2</a></td>
</tr>
<tr>
<td>pdsh-mod-machines-2.2.9-1el7.x86_64.rpm</td>
<td><a href="https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2">https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2</a></td>
</tr>
<tr>
<td>pdsh-mod-netgroup-2.2.9-1el7.x86_64.rpm</td>
<td><a href="https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2">https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2</a></td>
</tr>
<tr>
<td>pdsh-rcmd-exec-2.2.9-1el7.x86_64.rpm</td>
<td><a href="https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2">https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2</a></td>
</tr>
<tr>
<td>pdsh-rcmd-ssh-2.2.9-1el7.x86_64.rpm</td>
<td><a href="https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2">https://pdsh.googlecode.com/files/pdsh-2.29.tar.bz2</a></td>
</tr>
</tbody>
</table>

### 3.5. Security Concepts Appendices
<table>
<thead>
<tr>
<th>RPM</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>perl-DateTime-Format-Mail-0.3001-17.el7.noarch.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/perl-DateTime-Format-Mail-0.3001-17.el7.noarch.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/perl-DateTime-Format-Mail-0.3001-17.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>perl-DateTime-Format-W3CDTF-0.06-5.el7.noarch.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/perl-DateTime-Format-W3CDTF-0.06-5.el7.noarch.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/perl-DateTime-Format-W3CDTF-0.06-5.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>perl-XML-RSS-1.54-1.el7.noarch.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/perl-XML-RSS-1.54-1.el7.noarch.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/perl-XML-RSS-1.54-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>pssh-2.3.1.SIMP-5.el7.noarch.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/pssh-2.3.1-5.el7.noarch.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/pssh-2.3.1-5.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>pupmod-electrical-elsearch-0.1-2.5.noarch.rpm</td>
<td><a href="https://github.com/elsearch/puppet-elsearch">https://github.com/elsearch/puppet-elsearch</a></td>
</tr>
<tr>
<td>pupmod-electrical-logstash-0.3-4-2.noarch.rpm</td>
<td><a href="https://github.com/logstash/puppet-logstash">https://github.com/logstash/puppet-logstash</a></td>
</tr>
<tr>
<td>pupmod-puppetlabs-activemq-0.2-0-1.noarch.rpm</td>
<td><a href="https://github.com/puppetlabs/puppetlabs-activemq">https://github.com/puppetlabs/puppetlabs-activemq</a></td>
</tr>
<tr>
<td>puppet-3.7-4.1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/products/x86_64/puppet-3.7-4.1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/products/x86_64/puppet-3.7-4.1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>puppetdb-2.2-2.1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7Server/products/x86_64/puppetdb-2.2-2.1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7Server/products/x86_64/puppetdb-2.2-2.1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>puppetdb-terminus-2.2-2.1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7Server/products/x86_64/puppetdb-terminus-2.2-2.1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7Server/products/x86_64/puppetdb-terminus-2.2-2.1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>puppetlabs-stdlib-3.2-0.1.noarch.rpm</td>
<td><a href="https://github.com/puppetlabs/puppetlabs-stdlib/releases/tag/3.2-0.1">https://github.com/puppetlabs/puppetlabs-stdlib/releases/tag/3.2-0.1</a></td>
</tr>
<tr>
<td>puppetserver-1.0.2-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/products/x86_64/puppetserver-1.0.2-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/products/x86_64/puppetserver-1.0.2-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>python-elsearch-1.0.0-0.el7.centos.noarch.rpm</td>
<td><a href="https://github.com/elsearch/puppet-elsearch">https://github.com/elsearch/puppet-elsearch</a></td>
</tr>
<tr>
<td>python-redis-2.10-3-1.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/products/x86_64/puppet-redis-2.10-3-1.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/products/x86_64/puppet-redis-2.10-3-1.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>python-simplejson-3.3.3-1.el7.x86_64.rpm</td>
<td>EPEL</td>
</tr>
<tr>
<td>python-unittests2-0.5-1-6.el7.noarch.rpm</td>
<td>EPEL Updates</td>
</tr>
<tr>
<td>rrdaool-1.4-6.el7.x86_64.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/rrdtool-1.4-6.el7.x86_64.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/rrdtool-1.4-6.el7.x86_64.rpm</a></td>
</tr>
<tr>
<td>ruby-augaeas-0.5-0-1.el7.x86_64.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/ruby-augaeas-0.5-0-1.el7.x86_64.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/ruby-augaeas-0.5-0-1.el7.x86_64.rpm</a></td>
</tr>
<tr>
<td>ruby-ldap-0.9.10-17el7.x86_64.rpm</td>
<td><a href="http://mirror.pnl.gov/fedora/linux/releases/20/Everything/source/SRPMS/r/ruby-ldap-0.9.10-17.fc20.src.rpm">http://mirror.pnl.gov/fedora/linux/releases/20/Everything/source/SRPMS/r/ruby-ldap-0.9.10-17.fc20.src.rpm</a></td>
</tr>
<tr>
<td>ruby-rgen-0.6-5-2.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/products/x86_64/puppet-rgen-0.6-5-2.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/products/x86_64/puppet-rgen-0.6-5-2.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>ruby-shadow-2.2-2.0-2.el7.x86_64.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/products/x86_64/puppet-shadow-2.2-2.0-2.el7.x86_64.rpm">http://yum.puppetlabs.com/el/7/products/x86_64/puppet-shadow-2.2-2.0-2.el7.x86_64.rpm</a></td>
</tr>
<tr>
<td>rubygem-bignodecimal-1.2-20.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>rubygem-caipistrano-2.12-0-1.noarch.rpm</td>
<td><a href="http://rubygems.org/gems/caipistrano">http://rubygems.org/gems/caipistrano</a></td>
</tr>
<tr>
<td>rubygem-daemon_controller-1.2-0-1.noarch.rpm</td>
<td><a href="http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/rubygem-daemon_controller-1.2-0-1.noarch.rpm">http://dl.fedoraproject.org/pub/epel/beta/7/SRPMS/rubygem-daemon_controller-1.2-0-1.noarch.rpm</a></td>
</tr>
<tr>
<td>rubygem-deep_merge-1.0.0-2.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/products/x86_64/puppet-deep_merge-1.0.0-2.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/products/x86_64/puppet-deep_merge-1.0.0-2.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>rubygem-ffi-1.4-0-2.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>rubygem-highline-1.6-11-5.el7.noarch.rpm</td>
<td><a href="http://yum.puppetlabs.com/el/7/products/x86_64/puppet-highline-1.6-11-5.el7.noarch.rpm">http://yum.puppetlabs.com/el/7/products/x86_64/puppet-highline-1.6-11-5.el7.noarch.rpm</a></td>
</tr>
<tr>
<td>rubygem-io-console-0.4-2.20.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>rubygem-json-1.7.7-20.el7.x86_64.rpm</td>
<td><a href="http://rubygems.org/gems/json">http://rubygems.org/gems/json</a></td>
</tr>
<tr>
<td>rubygem-securepuppet-0.4-4.1-1.el7.x86_64.rpm</td>
<td><a href="http://rubygems.org/gems/securepuppet">http://rubygems.org/gems/securepuppet</a></td>
</tr>
<tr>
<td>rubygem-psych-2.0-0-20.el7.x86_64.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>rubygem-rack-1.5-2-1.el7.noarch.rpm</td>
<td><a href="http://rubygems.org/gems/rack">http://rubygems.org/gems/rack</a></td>
</tr>
<tr>
<td>rubygem-rake-0.9-6-20.el7.noarch.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>rubygem-rake-complier-0.9-3-1.el7.noarch.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>rubygem-rdoc-4.0-0-20.el7.noarch.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
<tr>
<td>rubygem-stomp-1.3-2-1.el7.noarch.rpm</td>
<td><a href="http://rubygems.org/gems/stomp">http://rubygems.org/gems/stomp</a></td>
</tr>
<tr>
<td>rubygem-stomp-doc-1.3-2-1.el7.noarch.rpm</td>
<td><a href="http://rubygems.org/gems/stomp-doc">http://rubygems.org/gems/stomp-doc</a></td>
</tr>
<tr>
<td>rubygems-0.14-20.el7.noarch.rpm</td>
<td>Recompiled from vendor SRPM</td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.2 – continued from previous page

<table>
<thead>
<tr>
<th>RPM</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>simp-hiera-1.3.3-1.el7.noarch.rpm</td>
<td><a href="https://github.com/onyxpoint/hiera">https://github.com/onyxpoint/hiera</a></td>
</tr>
<tr>
<td>simp-lastbind-2.4.23-0.x86_64.rpm</td>
<td><a href="https://github.com/gcp/openldap/tree/master/contrib/slapd-modules">https://github.com/gcp/openldap/tree/master/contrib/slapd-modules</a></td>
</tr>
<tr>
<td>simp-ppolicy-check-password-2.4.39-0el7.x86_64.rpm</td>
<td>Source Unknown - To be corrected</td>
</tr>
<tr>
<td>sudosh2-1.0.2-2.el7.x86_64.rpm</td>
<td><a href="http://sourceforge.net/projects/sudosh/">http://sourceforge.net/projects/sudosh/</a></td>
</tr>
<tr>
<td>tanukiwrapper-3.5.9-1.el6.x86_64.rpm</td>
<td><a href="http://wrapper.tanukisoftware.org/">http://wrapper.tanukisoftware.org/</a></td>
</tr>
</tbody>
</table>

RHEL RPMs

Table 3.3: RHEL RPMs

<table>
<thead>
<tr>
<th>RPM</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>apr-util-ldap-1.5.2-6.el7.x86_64.rpm</td>
<td>rhel-7-optional</td>
</tr>
<tr>
<td>mod_ldap-2.4.6-31.el7.x86_64.rpm</td>
<td>rhel-7-optional</td>
</tr>
<tr>
<td>openssh-ldap-6.6.1p1-11.el7.x86_64.rpm</td>
<td>rhel-7-optional</td>
</tr>
<tr>
<td>sendmail-milter-8.14.7-4.el7.x86_64.rpm</td>
<td>rhel-7-optional</td>
</tr>
</tbody>
</table>

3.5.5 SIMP SCTM

This SCTM was developed based on the National Institute of Standards and Technology (NIST) Specical Publication 800-53 (Revision 3) controls that SIMP currently meets. Empty contents means SIMP does not meet that control. Implementations are free to take these tables and use them as a starting point for any accreditation activities that follow NIST 800-53.

SIMP SCTM Technical Controls

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-1</td>
<td>Access Control Policy and Procedures</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-2(1)</td>
<td>Account Management (Control Enhancement)</td>
<td>Access Control</td>
<td>LDAP is used to centrally manage accounts. Local accounts can optionally be added and managed by puppet.</td>
</tr>
<tr>
<td>AC-2(2)</td>
<td>Account Management (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-2(3)</td>
<td>Account Management (Control Enhancement)</td>
<td>Access Control</td>
<td>Inactive local accounts expire 35 days after password expiration. LDAP accounts can be set to expire in LDAP and using PAM. There is no automated method (included with SIMP) to check inactive LDAP accounts. Implementations should address inactive LDAP accounts with automated or administrative measures.</td>
</tr>
<tr>
<td>AC-2(4)</td>
<td>Account Management (Control Enhancement)</td>
<td>Access Control</td>
<td>Local account creation is audited with auditd. (as are all of root’s actions). Sudo logs all commands for someone running sudo. This will not work if the SIMP implementation uses specific sudo rules. Instead, sudo actions are logged using auditd. Ldap modifications are logged in the ldap logs.</td>
</tr>
<tr>
<td>AC-2(5)</td>
<td>Account Management (Control Enhancement)</td>
<td>Access Control</td>
<td>Shell accounts are logged out after 15 minutes of inactivity</td>
</tr>
<tr>
<td>AC-2(6)</td>
<td>Account Management (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-2(7)</td>
<td>Account Management (Control Enhancement)</td>
<td>Access Control</td>
<td>SIMP has a default administrators group (700) that users can be assigned to. Additional roles and groups are up to the implementations. Role changes are logged in the LDAP logs.</td>
</tr>
<tr>
<td>AC-3</td>
<td>Access Enforcement</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-3(2)</td>
<td>Access Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-3(3)</td>
<td>Access Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td>DAC has been built into Unix for a long time and is expected to work. Implementations may want to check that user assignments to groups properly enforce DAC they way they expect. New as of SIMP 5.0 is the use of MAC. All stock SIMP modules work with MAC enabled.</td>
</tr>
<tr>
<td>AC-3(4)</td>
<td>Access Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td>DAC has been built into Unix for a long time and is expected to work. Implementations may want to check that user assignments to groups properly enforce DAC they way they expect.</td>
</tr>
<tr>
<td>AC-3(5)</td>
<td>Access Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td>SIMP implements file permissions per the SCAP-Security-Guide (SSG) RHEL7 guidance. There are some exceptions of file permissions being more or less restrictive than the guide. Mitigations and responses to those variances will be published once final RHEL7 SCAP content is available.</td>
</tr>
<tr>
<td>AC-3(6)</td>
<td>Access Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-4(1)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td>IPTables enforces flow control to the puppet master and clients. The default rules allow the services needed for kick start and puppet (and SSH of course). IPTables is managed by puppet so that any user modifications to /etc/sysconfig/iptables is rewritten with the rules from the manifest. The rules can and should be tailored per implementation.</td>
</tr>
<tr>
<td>AC-4(2)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(3)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(4)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(5)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(6)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(7)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(8)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(9)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(10)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(11)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(12)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-4(13)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(14)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(15)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(16)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-4(17)</td>
<td>Information Flow Enforcement (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-5</td>
<td>Separation of Duties</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-6</td>
<td>Least Privilege</td>
<td>Access Control</td>
<td>SIMP was built using a minimalist approach. Only the services, applications (RPMs and their dependencies), and network rules that are needed are implemented. Adding additional services, users, or software are done using built in RedHat/CentOS features or puppet. For example, services cannot be manually added without first registering them with puppet.</td>
</tr>
<tr>
<td>AC-6(1)</td>
<td>Least Privilege (Control Enhancement)</td>
<td>Access Control</td>
<td>File permissions and administrative functions are denied to users who are not administrators using Unix DAC. Roles can be defined by a implementation. Typically it’s done using ldap groups and sudosh. Suoders rules can be set for roles that need a limited set of commands/functions.</td>
</tr>
</tbody>
</table>

Continued on next page
### Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-6(2)</td>
<td>Least Privilege (Control Enhancement)</td>
<td>Access Control</td>
<td>Direct remote root login is not allowed on SIMP. Users must assume their role first (defined in LDAP or locally). There is a local simp user on the puppet master that has a password assigned. That allows for emergency maintenance via SSH. Single user mode is password protected, but will allow direct access before escalation. Protection of the single user mode and simp user’s password is up to the implementation. Privilege escalation is performed using sudosh or sudo. Most implementations will use sudosh for global admins and sudo for roles that need minimal admin ability. Lastly, serial port access does allow direct root login (/etc/securetty). Implementations may further restrict this at the risk.</td>
</tr>
<tr>
<td>AC-6(3)</td>
<td>Least Privilege (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-6(4)</td>
<td>Least Privilege (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-6(5)</td>
<td>Least Privilege (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-6(6)</td>
<td>Least Privilege (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-7</td>
<td>Unsuccessful Login Attempts</td>
<td>Access Control</td>
<td>SIMP locks accounts after 5 invalid attempts over 15 minutes span. It then keeps the account locked for 15 minutes. After that, the account is unlocked automatically.</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-7(1)</td>
<td>Unsuccessful Login Attempts (Control Enhancement)</td>
<td>Access Control</td>
<td>An account is never locked to a point an admin must unlock it. It will continue to be unlocked after 15 minutes. This should meet most modern policies. It can be further restricted if required by local policies.</td>
</tr>
<tr>
<td>AC-7(2)</td>
<td>Unsuccessful Login Attempts (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-8</td>
<td>System Use Notification</td>
<td>Access Control</td>
<td>SIMP displays a default banner prior to login. Implementations must customize that banner for their use.</td>
</tr>
<tr>
<td>AC-9</td>
<td>Previous Logon (Access) Notification</td>
<td>Access Control</td>
<td>SIMP uses the pam_lastlog.so module to display last login information.</td>
</tr>
<tr>
<td>AC-9(1)</td>
<td>Previous Logon (Access) Notification (Control Enhancement)</td>
<td>Access Control</td>
<td>SIMP uses the pam_lastlog.so module to display last login information.</td>
</tr>
<tr>
<td>AC-9(2)</td>
<td>Previous Logon (Access) Notification (Control Enhancement)</td>
<td>Access Control</td>
<td>SIMP uses the pam_lastlog.so module to display last login information, including the number of failed login attempts since the last logon.</td>
</tr>
<tr>
<td>AC-9(3)</td>
<td>Previous Logon (Access) Notification (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-10</td>
<td>Concurrent Session Control</td>
<td>Access Control</td>
<td>The default value for concurrent sessions in SIMP is 10 (/etc/security/limits.conf). Given the variety of system usage to include automated processes, it could impact functionality if this value were set lower. It can be tailored to a lower value if the implementation determines that number will not impact functionality.</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-11</td>
<td>Session Lock</td>
<td>Access Control</td>
<td>Terminal sessions do not enforce a session lock so this control is technically not implemented. However, it’s mitigated by forcing inactive sessions to log out. If the gnome module is applied, SIMP locks a gnome session after 5 minutes.</td>
</tr>
<tr>
<td>AC-14</td>
<td>Permitted Actions without Identification or Authentication</td>
<td>Access Control</td>
<td>SIMP provides several services that do not require authentication. Most require some form of identification. These are documented in the SIMP Security Concepts and is kept current for that version. Individual modules are not yet documented.</td>
</tr>
<tr>
<td>AC-14(1)</td>
<td>Permitted Actions without Identification or Authentication (Control Enhancement)</td>
<td>Access Control</td>
<td>Justifications to those services that do not require identification and authentication can be found in the SIMP Security Concepts document.</td>
</tr>
<tr>
<td>AC-16</td>
<td>Security Attributes</td>
<td>Access Control</td>
<td>New in SIMP 5.0 is the usage of MAC via SELinux. This is optional for each implementation and can be turned off at any time. All of the stock SIMP modules work with SELinux enabled and have the least restrictive MAC policies enforced. These policies assign each object a SELinux user, role, type, and level. These characteristics are used to define a context for each object.</td>
</tr>
<tr>
<td>AC-16(1)</td>
<td>Security Attributes (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-16(2)</td>
<td>Security Attributes (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-16(3)</td>
<td>Security Attributes (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-16(4)</td>
<td>Security Attributes (Control Enhancement)</td>
<td>Access Control</td>
<td>SeLinux user, role, type, and level are the security attributes that are associated with each object with SELinux enabled in SIMP.</td>
</tr>
<tr>
<td>AC-16(5)</td>
<td>Security Attributes (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-17</td>
<td>Remote Access</td>
<td>Access Control</td>
<td>By default, external connections are not allowed with the exception of SSH. This is documented in the SIMP user manual. Implementations have the ability to override this with the understanding that puppet controls Iptables.</td>
</tr>
<tr>
<td>AC-17(1)</td>
<td>Remote Access (Control Enhancement)</td>
<td>Access Control</td>
<td>The extent of monitoring remote connections is done by auditd and syslog. The contents of the remote session is not logged. The keystrokes of users with su-dosh shells are all logged.</td>
</tr>
<tr>
<td>AC-17(2)</td>
<td>Remote Access (Control Enhancement)</td>
<td>Access Control</td>
<td>Remote access is limited to SSH. SSH (openssh on centos/rhel) provides both confidentiality and integrity of the remote session.</td>
</tr>
<tr>
<td>AC-17(3)</td>
<td>Remote Access (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-17(4)</td>
<td>Remote Access (Control Enhancement)</td>
<td>Access Control</td>
<td>This control is enforced via other access control mechanisms already covered in 800-53. Namely, AC-6. By default, SSH in SIMP will allow anyone to connect. Once identification and authentication is performed, access control to privileged commands is enforced as usual.</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-17(5)</td>
<td>Remote Access (Control Enhancement)</td>
<td>Access Control</td>
<td>Auditd provides logging of failed access attempts. It’s up to the implementation to perform a level of inspection of these unauthorized events. Auditd does this by default. Other checks will ensure auditd is running and registered with puppet.</td>
</tr>
<tr>
<td>AC-17(6)</td>
<td>Remote Access (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-17(7)</td>
<td>Remote Access (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-17(8)</td>
<td>Remote Access (Control Enhancement)</td>
<td>Access Control</td>
<td>This control is only met by defining all connections that SIMP allows internally and externally. For now, since this is a remote access control, it should suffice to continue to note that the only remote access protocol allowed by default is SSH.</td>
</tr>
<tr>
<td>AC-18</td>
<td>Wireless Access</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-18(1)</td>
<td>Wireless Access (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-18(2)</td>
<td>Wireless Access (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-18(3)</td>
<td>Wireless Access (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-18(4)</td>
<td>Wireless Access (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-18(5)</td>
<td>Wireless Access (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-19</td>
<td>Access Control for Mobile Devices</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-19(1)</td>
<td>Access Control for Mobile Devices (Control Enhance-ment)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-19(2)</td>
<td>Access Control for Mobile Devices (Control Enhance-ment)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-19(3)</td>
<td>Access Control for Mobile Devices (Control Enhance-ment)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-19(4)</td>
<td>Access Control for Mobile Devices (Control Enhance-ment)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-20</td>
<td>Use of External Information Systems</td>
<td>Access Control</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC-20(1)</td>
<td>Use of External Information Systems (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-20(2)</td>
<td>Use of External Information Systems (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-21</td>
<td>User-Based Collaboration and Information Sharing</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-21(1)</td>
<td>User-Based Collaboration and Information Sharing (Control Enhancement)</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AC-22</td>
<td>Publicly Accessible Content</td>
<td>Access Control</td>
<td></td>
</tr>
<tr>
<td>AU-1</td>
<td>Audit and Accountability Policy and Procedures</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-2</td>
<td>Auditable Events</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
</tbody>
</table>

1. SIMP audit rules were built by using industry best practices gathered over the years. The heaviest reliance has been on the SCAP-Security Guide (SSG). SIMP aims for a balance between performance and operational needs so the settings are rarely an exact match from these guides. The list of events that audited are by auditd can be found in appendix of the Security Concepts document. b. Implementation Specific c. Rational is for audit setting is provided in SSG. d. Threat information is specific to the implementation. Auditd and syslog facility can always be fine tuned for each implementation.
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AU-2(3)</td>
<td>Auditable Events (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>SIMP is constantly reviewing the audit rules for accuracy, relevance, and performance. Rules are added and in some cases removed as information becomes available.</td>
</tr>
<tr>
<td>AU-2(4)</td>
<td>Auditable Events (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>Privileged user commands are logged using sudosh and auditd (sudo actions). By default, users in the administrators group can run sudosh. All of the key strokes (except things that are not echoed back to the screen like passwords) are logged to /var/log/sudosh.log and can be sent to syslog. If an implementation sets up specific sudo actions for other groups or users, those actions are logged with auditd.</td>
</tr>
<tr>
<td>AU-3</td>
<td>Content of Audit Records</td>
<td>Audit and Accountability</td>
<td>The linux audit daemon contains event type, date/time, host, and outcome of events by default.</td>
</tr>
<tr>
<td>AU-3(1)</td>
<td>Content of Audit Records (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>There are a number of events that are captured beyond the auditd. The SIMP syslog module captures additional log events from apache, ldap, puppet, messages.log, and secure.log.</td>
</tr>
<tr>
<td>AU-3(2)</td>
<td>Content of Audit Records (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>By default, the SIMP syslog module logs locally. There is an option to send the syslog events to a central location. Instructions for implementing a syslog server are provided in the User Guide. Lastly, a combination of elasticsearch, logstash, and kibana (ELK) can be applied to filter, index, and search logs. Puppet modules are provided for the ELK stack.</td>
</tr>
</tbody>
</table>
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AU-4</td>
<td>Audit Storage Capacity</td>
<td>Audit and Accountability</td>
<td>The audit partition is configured as a separation partition from the system files, reducing the likelihood of audit interfering with system operations. Implementations can change this but it’s highly discouraged.</td>
</tr>
<tr>
<td>AU-5</td>
<td>Response to Audit Processing Failures</td>
<td>Audit and Accountability</td>
<td>1. Implementation Specific: b. The audit.conf file configures the system to log to syslog when disk space becomes low. If the disk becomes full, the audit daemon will be suspended, but the system will remain active. This is contrary to some industry guidance to put the system into single user mode when disk space becomes an issue. Implementations may wish to change the default behaviour at the risk of stopping the system from functioning.</td>
</tr>
<tr>
<td>AU-5(1)</td>
<td>Response to Audit Processing Failures (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>SIMP provides a warning (to syslog) when the disk has 75MB free. Each log file can be up to 30MB.</td>
</tr>
<tr>
<td>AU-5(2)</td>
<td>Response to Audit Processing Failures (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-5(3)</td>
<td>Response to Audit Processing Failures (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-5(4)</td>
<td>Response to Audit Processing Failures (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>SIMP will not shut down a system by default. Implementation can configure this option at the own risk in the auditd.conf file.</td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AU-6</td>
<td>Audit Review, Analysis, and Reporting</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-6(1)</td>
<td>Audit Review, Analysis, and Reporting (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-6(3)</td>
<td>Audit Review, Analysis, and Reporting (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>The ELK modules provide implementations with one means to centralize, review, and recognize trends in SIMP logs.</td>
</tr>
<tr>
<td>AU-6(4)</td>
<td>Audit Review, Analysis, and Reporting (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>The ELK modules provide implementations with one means to centralize, review, and recognize trends in SIMP logs.</td>
</tr>
<tr>
<td>AU-6(5)</td>
<td>Audit Review, Analysis, and Reporting (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>The ELK modules provide implementations with one means to centralize, review, and recognize trends in SIMP logs. The logs sent to syslog can be customized to include logs from any application. They would then be in a central place for viewing and aggregation by users of the Kibana interface.</td>
</tr>
<tr>
<td>AU-6(6)</td>
<td>Audit Review, Analysis, and Reporting (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-6(7)</td>
<td>Audit Review, Analysis, and Reporting (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-6(9)</td>
<td>Audit Review, Analysis, and Reporting (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-7</td>
<td>Audit Reduction and Report Generation</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
### Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AU-7(1)</td>
<td>Audit Reduction and Report Generation (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>While not true audit reduction, RedHat does allow someone with access to audit logs to perform filters using the journald. If audit logs are forwarded to a syslog server, it’s not difficult for an admin to security officer to run batch filters against all of the audit records. As of SIMP 4.0.5, an optional Logstash, Kibana, and Elasticsearch modules can be applied. If applied, they provide centralized and indexed logs. An implementation can then perform searches against the logs or provide alerts to other parts of their infrastructure.</td>
</tr>
<tr>
<td>AU-8</td>
<td>Time Stamps</td>
<td>Audit and Accountability</td>
<td>Auditd uses the system clock to time stamp audit events.</td>
</tr>
<tr>
<td>AU-8(1)</td>
<td>Time Stamps (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td>Time is an essential component of puppet. Therefore, NTPD is used to synchronize puppet clients with the puppet server. That default configuration can be changed to synchronize puppet each server/client with another time source.</td>
</tr>
<tr>
<td>AU-9</td>
<td>Protection of Audit Information</td>
<td>Audit and Accountability</td>
<td>File system permissions and SELinux protect the content of /var/log/audit and /etc/audit/*</td>
</tr>
<tr>
<td>AU-9(1)</td>
<td>Protection of Audit Information (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-9(2)</td>
<td>Protection of Audit Information (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-9(3)</td>
<td>Protection of Audit Information (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
<tr>
<td>AU-9(4)</td>
<td>Protection of Audit Information (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
### Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Method</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AU-10</td>
<td>Non-repudiation</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-10(1)</td>
<td>Non-repudiation (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-10(2)</td>
<td>Non-repudiation (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-10(3)</td>
<td>Non-repudiation (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-10(4)</td>
<td>Non-repudiation (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-10(5)</td>
<td>Non-repudiation (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-12(1)</td>
<td>Audit Generation (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
<td>1. Auditd provides the audit generation capability and is running on all SIMP systems by default. b. The audit.rules file configures events that are audited. c. The audit.rules applies the list of audit rules defined in SIMP Security Concepts document. Auditd stamps audit records with the system time. The system time is obtained from a central time source and synchronized between SIMP systems.</td>
</tr>
<tr>
<td>AU-11</td>
<td>Audit Record Retention</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-12</td>
<td>Audit Generation</td>
<td>Audit and Accountability</td>
<td></td>
<td>Auditd provides logging in standard formats. Additionally, logs that are sent through syslog adhere to that standard.</td>
</tr>
<tr>
<td>AU-12(1)</td>
<td>Audit Generation (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-13</td>
<td>Monitoring For Information Disclosure</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-14</td>
<td>Session Audit</td>
<td>Audit and Accountability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AU-14(1)</td>
<td>Session Audit (Control Enhancement)</td>
<td>Audit and Accountability</td>
<td></td>
<td>Sessions that use the sudo shell have all keystrokes recorded. Those sessions can be viewed in text format or replayed to the screen.</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA-1</td>
<td>Identification and Authentication Policy and Procedures</td>
<td>Identification and Authentication</td>
<td>Method</td>
</tr>
<tr>
<td>IA-2(1)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-2(2)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-2(3)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-2(4)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-2(5)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-2(6)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-2(7)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-2(8)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
</tbody>
</table>

The authentication mechanisms used within SIMP are all resistant to replay attacks by default. Known vulnerabilities can occur in the protocols. As they are known, vendors release patches, which must then be applied by the implementation. Privileged accounts use the same protocols as unprivileged accounts.

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA-2(9)</td>
<td>User Identification and Authentication (Organizational Users) (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>The authentication mechanisms used within SIMP are all resistant to replay attacks by default. Known vulnerabilities can occur in the protocols. As they are known, vendors release patches, which must then be applied by the implementation.</td>
</tr>
<tr>
<td>IA-3</td>
<td>Device Identification and Authentication</td>
<td>Identification and Authentication</td>
<td>Identification of each puppet client occurs before an IP address can be assigned. This is controlled using DHCP (each client must have an address bound by MAC address). Devices identification and authentication with puppet occurs using SSL certificates. The clients must each have a SSL certificate installed to establish a valid session with the puppet master.</td>
</tr>
<tr>
<td>IA-3(1)</td>
<td>Device Identification and Authentication (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-3(2)</td>
<td>Device Identification and Authentication (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-3(3)</td>
<td>Device Identification and Authentication (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>DHCP is used to statically define the IP addresses of each puppet client.</td>
</tr>
<tr>
<td>IA-4</td>
<td>Identifier Management</td>
<td>Identification and Authentication</td>
<td>Local accounts expire 35 days after their passwords expire. There is no mechanism implemented to detect inactive LDAP accounts. Implementations might wish to mitigate this by regularly reviewing and removing unneeded accounts.</td>
</tr>
<tr>
<td>IA-4(1)</td>
<td>Identifier Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-4(2)</td>
<td>Identifier Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-4(3)</td>
<td>Identifier Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA-4(4)</td>
<td>Identifier Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-4(5)</td>
<td>Identifier Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-5</td>
<td>Authenticator Management</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
</tbody>
</table>

3. Authenticator strength is enforced using `pam_crack_lib.so`. This works for user defined passwords on local and LDAP accounts. E. It’s up to the implementation to change the values for the various passwords. F. Password history is set to 24 by default in SIMP and enforced with `pam`. G. For local accounts, password aging is set to 180 days. It’s set to the same in LDAP, but enforced at the time of account creation using ldifs. LDAP subsequently uses PAM to enforce the aging. Key based passwordless logins do not enforce aging. Upon generation, server and puppet certificates can also be set to expire. H. Authenticators for local and LDAP account are protected using operating system access controls. The server certificates are also protected using operating system controls.
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA-5(1)</td>
<td>Authenticator Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>1. Authenticator strength is enforced using pam_crack_lib.so. This works for user defined passwords on local and LDAP accounts. Administrators can bypass PAM and set weak passwords in LDAP. Under normal circumstances, users would be forced to change their password at login, at which point pam enforced complexity. b. Not enforced c. Hashed passwords are built into linux (/etc/shadow and /etc/pam.d/system-auth pam_unix.so). LDAP password changed by users are done through pam before getting placed in LDAP. Manual LDAP password are created using the slapasswd command. d. Password minimum and maximum lifetimes are enforced through /etc/login.defs and ldap. e. By default, the previous 24 passwords can not be reused.</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA-5(2)</td>
<td>Authenticator Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>Puppet comes with a self contained public key infrastructure. Though just used for puppet, it operates as a full PKI. So the certificate path is validated. SSL certificates that are used for SSL and TLS also have certificate path validation built into the protocol. Note: SSH Keys are not considered PKI.</td>
</tr>
<tr>
<td>IA-5(3)</td>
<td>Authenticator Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>Pam cracklib enforces password complexity rules on Redhat and CentOS. Additional tools to check authenticator strength can be used in operational settings.</td>
</tr>
<tr>
<td>IA-5(4)</td>
<td>Authenticator Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>The simp-config utility gives each implementation an opportunity to change default passwords at build time. It’s up to the implementation to change the values for the various passwords.</td>
</tr>
<tr>
<td>IA-5(5)</td>
<td>Authenticator Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>Authenticators are protected with operating system access control and file permissions.</td>
</tr>
<tr>
<td>IA-5(6)</td>
<td>Authenticator Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>Plaintext passwords are only used when application support no other means of providing a password.</td>
</tr>
<tr>
<td>IA-5(7)</td>
<td>Authenticator Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td>Plaintext passwords are not echoed back to the screen.</td>
</tr>
<tr>
<td>IA-5(8)</td>
<td>Authenticator Management (Control Enhancement)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>IA-6</td>
<td>Authenticator Feedback</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA-7</td>
<td>Cryptographic Module Authentication</td>
<td>Identification and Authentication</td>
<td>Redhat 7 and the several modules are being evaluated for FIPS 140 compliance. Implementations should check the FIPS site for updates on this evaluation. The SIMP team will also continue to evaluate the status and any relevant settings that need to be applied as a result of this evaluation.</td>
</tr>
<tr>
<td>IA-8</td>
<td>Identification and Authentication (Non-Organizational Users)</td>
<td>Identification and Authentication</td>
<td></td>
</tr>
<tr>
<td>SC-1</td>
<td>System and Communications Protection Policy and Procedures</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-2</td>
<td>Application Partitioning</td>
<td>System and Communications Protection</td>
<td>The spirit of this control is providing logical separation so that users are not able to access administrative functions. There is no notion of partitioning within SIMP. There are access control enforcement that can be proven through tests on those controls. If this control is allocated to SIMP alone, it’s unlikely it can be met. Since SIMP is the infrastructure that applications would use, showing that application users cannot access the SIMP environment is a better way to prove this control is met.</td>
</tr>
</tbody>
</table>

Continued on next page
### Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-2(1)</td>
<td>Application Partitioning</td>
<td>System and Communications Protection</td>
<td>The spirit of this control is providing logical separation so that users are not able to access administrative functions. There is no notion of partitioning within SIMP. There are access control enforcement that can be proven through tests on those controls. If this control is allocated to SIMP alone, it’s unlikely it can be met. Since SIMP is the infrastructure that applications would use, showing that application users cannot access the SIMP environment is a better way to prove this control is met.</td>
</tr>
<tr>
<td>SC-3</td>
<td>Security Function Isolation</td>
<td>System and Communications Protection</td>
<td>The spirit of this control is providing logical separation so that users are not able to access administrative functions. There is no notion of partitioning within SIMP. There are access control enforcement that can be proven through tests on those controls. If this control is allocated to SIMP alone, it’s unlikely it can be met. Since SIMP is the infrastructure that applications would use, showing that application users cannot access the SIMP environment is a better way to prove this control is met.</td>
</tr>
<tr>
<td>SC-3(1)</td>
<td>Security Function Isolation</td>
<td>System and Communications Protection</td>
<td>System and Communications Protection</td>
</tr>
<tr>
<td>SC-3(2)</td>
<td>Security Function Isolation</td>
<td>System and Communications Protection</td>
<td>System and Communications Protection</td>
</tr>
<tr>
<td>SC-3(3)</td>
<td>Security Function Isolation</td>
<td>System and Communications Protection</td>
<td>System and Communications Protection</td>
</tr>
<tr>
<td>SC-3(4)</td>
<td>Security Function Isolation</td>
<td>System and Communications Protection</td>
<td>System and Communications Protection</td>
</tr>
<tr>
<td>SC-3(5)</td>
<td>Security Function Isolation</td>
<td>System and Communications Protection</td>
<td>System and Communications Protection</td>
</tr>
</tbody>
</table>

Continued on next page

---

3.5. Security Concepts Appendices

141
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-4</td>
<td>Information In Shared Resources</td>
<td>System and Communications Protection</td>
<td>While difficult for the SIMP team to prove, object reuse has been part of previous versions of RedHat common criteria testing. That testing focusing on Files system objects, IPC objects and Memory objects. Any issues discovered within the platform that cause object reuse issues are likely to be address in security patches provided by the vendor.</td>
</tr>
<tr>
<td>SC-4(1)</td>
<td>Information In Shared Resources (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-5</td>
<td>Denial of Service Protection</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-5(1)</td>
<td>Denial of Service Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-5(2)</td>
<td>Denial of Service Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-6</td>
<td>Resource Priority</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7</td>
<td>Boundary Protection</td>
<td>System and Communications Protection</td>
<td>Most of this control deals with a separate boundary interface (FW etc.). There is a part of this control that deals with controlling network access at key internal boundary points. Since SIMP implements IPTables on all hosts (by default), each node might be considered an internal boundary. Note – internal boundaries are more likely implemented via vlans or internal layer 3 devices.</td>
</tr>
<tr>
<td>SC-7(1)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(2)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(3)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(4)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-7(5)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td>Iptables, as configured by default, blocks all incoming traffic except for what is explicitly allowed.</td>
</tr>
<tr>
<td>SC-7(6)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(7)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(8)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(9)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(10)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(11)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(12)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td>IPTables is the host based firewall implementation on RedHat/CentOS.</td>
</tr>
<tr>
<td>SC-7(13)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(14)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(15)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(16)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(17)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-7(18)</td>
<td>Boundary Protection (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
### Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-8</td>
<td>Transmission Integrity</td>
<td>System and Communications Protection</td>
<td>With the exception of the services needed for kick-start, most communications within SIMP are protected by SSH or SSL. Implementations can add additional services or modules that do not use SSH or SSL. The SIMP Security Concepts document details the default allowed protocols and the mechanisms in place to protect them. It's also worth noting that the SIMP team has taken every measure possible to remove encryption ciphers available to operating system applications. In the event this breaks an application, implementations might have to add those ciphers back.</td>
</tr>
<tr>
<td>SC-8(1)</td>
<td>Transmission Integrity (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td>With the exception of the services needed for kick-start, most communications within SIMP are protected by SSH or SSL. Implementations can add additional services or modules that do not use SSH or SSL. The SIMP Security Concepts document details the default allowed protocols and the mechanisms in place to protect them. It's also worth noting that the SIMP team has taken every measure possible to remove encryption ciphers available to operating system applications. In the event this breaks an application, implementations might have to add those ciphers back.</td>
</tr>
<tr>
<td>SC-8(2)</td>
<td>Transmission Integrity (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td>With the exception of the services needed for kick-start, most communications within SIMP are protected by SSH or SSL. Implementations can add additional services or modules that do not use SSH or SSL. The SIMP Security Concepts document details the default allowed protocols and the mechanisms in place to protect them. It's also worth noting that the SIMP team has taken every measure possible to remove encryption ciphers available to operating system applications. In the event this breaks an application, implementations might have to add those ciphers back.</td>
</tr>
</tbody>
</table>

Continued on next page
### Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-9</td>
<td>Transmission Confidentiality</td>
<td>System and Communications Protection</td>
<td>With the exception of the services needed for kickstart, most communications within SIMP are protected by SSH or SSL. Implementations can add additional services or modules that do not use SSH or SSL. The SIMP Security Concepts document details the default allowed protocols and the mechanisms in place to protect them. It’s also worth noting that the SIMP team has taken every measure possible to remove encryption ciphers available to operating system applications. In the event this breaks an application, implementations might have to add those ciphers back.</td>
</tr>
<tr>
<td>SC-9(1)</td>
<td>Transmission Confidentiality (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td>With the exception of the services needed for kickstart, most communications within SIMP are protected by SSH or SSL. Implementations can add additional services or modules that do not use SSH or SSL. The SIMP Security Concepts document details the default allowed protocols and the mechanisms in place to protect them. It’s also worth noting that the SIMP team has taken every measure possible to remove encryption ciphers available to operating system applications. In the event this breaks an application, implementations might have to add those ciphers back.</td>
</tr>
<tr>
<td>SC-9(2)</td>
<td>Transmission Confidentiality (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-10</td>
<td>Network Disconnect</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-11</td>
<td>Trusted Path</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-12</td>
<td>Cryptographic Key Establishment and Management</td>
<td>System and Communications Protection</td>
<td>In an operational setting, SIMP does not establish keys. It does come with the ability to create server keys using a custom application known as “FakeCA”. SSH keys can also be established using standard Unix command line tools. In an operational settings, both sets of keys should be obtained from valid key infrastructures. There is also a CA that puppet uses to generate and manage keys for puppet only.</td>
</tr>
<tr>
<td>SC-12(1)</td>
<td>Cryptographic Key Establishment and Management (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-12(2)</td>
<td>Cryptographic Key Establishment and Management (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-12(3)</td>
<td>Cryptographic Key Establishment and Management (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-12(4)</td>
<td>Cryptographic Key Establishment and Management (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-12(5)</td>
<td>Cryptographic Key Establishment and Management (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-13</td>
<td>Use of Cryptography</td>
<td></td>
<td>The forms of cryptography used are applied through SSH, SSL, and TLS. Red-Hat FIPs mode enabling is on the near term horizon for SIMP. Once enabled, it will be documented here and should allow implementations to further explain how this control is being met. There are several unencrypted protocols used on the puppet server (Apache/YUM, DHCPD, TFTP, and DNS). The Security Concepts document provides additional details on default services/protocols that are used.</td>
</tr>
<tr>
<td>SC-13(1)</td>
<td>Use of Cryptography (Control Enhancement)</td>
<td></td>
<td>The forms of cryptography used are applied through SSH, SSL, and TLS. There are several unencrypted protocols used on the puppet server (Apache/YUM, DHCPD, TFTP, and DNS) that are documented in the Security Concepts document.</td>
</tr>
<tr>
<td>SC-13(2)</td>
<td>Use of Cryptography (Control Enhancement)</td>
<td></td>
<td>The forms of cryptography used are applied through SSH, SSL, and TLS. There are several unencrypted protocols used on the puppet server (Apache/YUM, DHCPD, TFTP, and DNS) that are documented in the Security Concepts document.</td>
</tr>
<tr>
<td>SC-13(3)</td>
<td>Use of Cryptography (Control Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-13(4)</td>
<td>Use of Cryptography (Control Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-14</td>
<td>Public Access Protections</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-15</td>
<td>Collaborative Computing Devices</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
<th>Implementation Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-15(1)</td>
<td>Collaborative Computing Devices (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td>In an operational setting, SIMP does not establish keys. It does come with the ability to create server keys using a custom application know as “FakeCA”. SSH keys can also be established using standard unix command line tools. In an operational settings, both sets of keys should be obtained from valid key infrastructures. There is also a CA that puppet uses to generate and manage keys for puppet only.</td>
</tr>
<tr>
<td>SC-15(2)</td>
<td>Collaborative Computing Devices (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-15(3)</td>
<td>Collaborative Computing Devices (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-16</td>
<td>Transmission of Security Attributes</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-16(1)</td>
<td>Transmission of Security Attributes (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-17</td>
<td>Public Key Infrastructure Certificates</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-18</td>
<td>Mobile Code</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-18(1)</td>
<td>Mobile Code (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-18(2)</td>
<td>Mobile Code (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-18(3)</td>
<td>Mobile Code (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-18(4)</td>
<td>Mobile Code (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-19</td>
<td>Voice Over Internet Protocol</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-20</td>
<td>Secure Name/Address Resolution Service (Authoritative Source)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-20(1)</td>
<td>Secure Name/Address Resolution Service (Authoritative Source) (Control Enhance-</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ment)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control ID</td>
<td>Control Name</td>
<td>Control Family</td>
<td>SIMP Implementation</td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>------------------------------------------------------------------------------</td>
<td>--------------------------------------</td>
<td>---------------------</td>
<td></td>
</tr>
<tr>
<td>SC-21</td>
<td>Secure Name/Address Resolution Service (Recursive or Caching Resolver)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-21(1)</td>
<td>Secure Name/Address Resolution Service (Recursive or Caching Resolver) (Con-</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-22</td>
<td>Architecture and Provisioning for Name/Address Resolution Service</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-23</td>
<td>Session Authenticity</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-23(1)</td>
<td>Session Authenticity (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-23(2)</td>
<td>Session Authenticity (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-23(3)</td>
<td>Session Authenticity (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC-23(4)</td>
<td>Session Authenticity (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-24</td>
<td>Fail in Known State</td>
<td>System and Communications Protection</td>
<td>The forms of cryptography used are applied through SSH, SSL, and TLS. There are several unencrypted protocols used on the puppet server (Apache/YUM, DHCPD, TFTP, and DNS) that are documented in the Security Concepts document.</td>
</tr>
<tr>
<td>SC-25</td>
<td>Thin Nodes</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-26</td>
<td>Honeypots</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-26(1)</td>
<td>Honeypots (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-27</td>
<td>Operating System-Independent Applications</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-28</td>
<td>Protection of Information at Rest</td>
<td>System and Communications Protection</td>
<td>Confidentiality of data at rest is achieved using the operating system access control. Integrity is only checked for critical operating system files. Implementations have the ability to extend the integrity checking of AIDE to include additional files that are not frequently changed.</td>
</tr>
<tr>
<td>SC-28</td>
<td>Protection of Information at Rest (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-29</td>
<td>Heterogeneity</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-30</td>
<td>Virtualization Techniques</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-30(1)</td>
<td>Virtualization Techniques (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-30(2)</td>
<td>Virtualization Techniques (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-31</td>
<td>Covert Channel Analysis</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-31(1)</td>
<td>Covert Channel Analysis (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-32</td>
<td>Information System Partitioning</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-33</td>
<td>Transmission Preparation Integrity</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
### Table 3.4 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC-34</td>
<td>Non-modifiable Executable Programs</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-34(1)</td>
<td>Non-modifiable Executable Programs (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
<tr>
<td>SC-34(2)</td>
<td>Non-modifiable Executable Programs (Control Enhancement)</td>
<td>System and Communications Protection</td>
<td></td>
</tr>
</tbody>
</table>

Table: SIMP SCTM

**SIMP SCTM Operational Controls**

Table: SIMP SCTM

**SIMP SCTM Management Controls**

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>AT-2(1)</td>
<td>Security Awareness (Control Enhancement)</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>AT-3</td>
<td>Security Training</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>AT-3(1)</td>
<td>Security Training (Control Enhancement)</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>AT-3(2)</td>
<td>Security Training (Control Enhancement)</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>AT-4</td>
<td>Security Training Records</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>AT-5</td>
<td>Contacts with Security Groups and Associations</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>CM-1</td>
<td>Configuration Management Policy and Procedures</td>
<td>Configuration Management</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training</td>
<td>Awareness and Training</td>
<td>SIMP has strictly enforced version control during development. The baseline files for SIMP</td>
</tr>
<tr>
<td></td>
<td>Policy and Procedures</td>
<td></td>
<td>are kept and maintained in a git repository. Files are packaged and a series of auto tests</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>are performed on each release. Once released, there is a version number associated for</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>distribution. Additionally, custom puppet modules are in the form of RPMs and have version</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>numbers associated with them. All documentation is also built with source code.</td>
</tr>
<tr>
<td>CM-2</td>
<td>Baseline Configuration</td>
<td>Configuration Management</td>
<td>SIMP has strictly enforced version control during development. The baseline files for SIMP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>are kept and maintained in a git repository. Files are packaged and a series of auto tests</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>are performed on the release. Once released, there is a version number associated for</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>distribution. All documentation is also built with source code.</td>
</tr>
<tr>
<td>CM-2(1)</td>
<td>Baseline Configuration (Control</td>
<td>Configuration Management</td>
<td>All old versions of SIMP remain in the code repository.</td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM-2(2)</td>
<td>Baseline Configuration (Control</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM-2(3)</td>
<td>Baseline Configuration (Control</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM-2(4)</td>
<td>Baseline Configuration (Control</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
### Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>CM-2(5)</td>
<td>Baseline Configuration (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-2(6)</td>
<td>Baseline Configuration (Control Enhancement)</td>
<td>Configuration Management</td>
<td>1. SIMP provides a minimal list of packages and services installed. The minimal list of packages can be found in kickstart files and the appendix of this document. Additional packages are installed by each implementation or as SIMP modules are applied. b. It's not feasible to technically deny additional applications from being installed. There is nothing in SIMP that can stop and RPM from being applied. Applications that require network access to service activation must be registered with puppet.</td>
</tr>
<tr>
<td>CM-3</td>
<td>Configuration Change Control</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-3(1)</td>
<td>Configuration Change Control (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-3(2)</td>
<td>Configuration Change Control (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>CM-3(3)</td>
<td>Configuration Change Control (Control Enhancement)</td>
<td>Configuration Management</td>
<td>Configuration changes in SIMP are automated using a combination of puppet, yum, and rsync. While not all files on an operating system are managed by those mechanisms, many are. Changes to critical files that are managed by puppet, revert back to their original state. These mechanisms were not meant to defeat an attack by a malicious insider.</td>
</tr>
<tr>
<td>CM-3(4)</td>
<td>Configuration Change Control (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-4</td>
<td>Security Impact Analysis</td>
<td>Configuration Management</td>
<td>All features or bugs in SIMP are vetted through the development process by being placed on the product backlog and discussed with the entire team. There is a security representative on the SIMP team that is part of that vetting process.</td>
</tr>
<tr>
<td>CM-4(1)</td>
<td>Security Impact Analysis (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-4(2)</td>
<td>Security Impact Analysis (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-5</td>
<td>Access Restrictions for Change</td>
<td>Configuration Management</td>
<td>SIMP can only meet the enforcement part of this control. The remainder must be met by the environment that SIMP is implemented in. Changes to a SIMP based systems are enforced with built in Unix/LDAP groups. Only someone with sudo or sudosh access (usually an admin group) can apply changes to the environment</td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td>SIMP can only meet the enforcement part of this control. The remainder must be met by the environment that SIMP is implemented in. Changes to a SIMP based systems are enforced with built in Unix/LDAP groups. Only someone with sudo or sudosh access (usually an admin group) can apply changes to the environment</td>
</tr>
<tr>
<td>CM-5(1)</td>
<td>Access Restrictions for Change (Control Enhancement)</td>
<td>Configuration Management</td>
<td>Redhat and Centos packages are signed with gpg keys. Those keys are vendor specific. Package installation occurs only when those gpgkeys are validate using the installed gpg public keys for the operating system. SIMP specific RPMS that were developed are signed using keys generate by the development team.</td>
</tr>
<tr>
<td>CM-5(2)</td>
<td>Access Restrictions for Change (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-5(3)</td>
<td>Access Restrictions for Change (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-5(4)</td>
<td>Access Restrictions for Change (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-5(5)</td>
<td>Access Restrictions for Change (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-5(6)</td>
<td>Access Restrictions for Change (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-5(7)</td>
<td>Access Restrictions for Change (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
</tbody>
</table>

Most of the critical files that are managed by puppet cannot be permanently changed on a puppet client without disabling puppet and rsync. If they are changed, puppet will revert them back to their original state.

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Pro-</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td></td>
<td>cedures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM-6</td>
<td>Configuration Settings</td>
<td>Configuration Management</td>
<td>Part “d” of this control is met by SIMP. The others are not. SIMP uses puppet to monitor changes to configuration settings. If changes to puppet controlled settings are manually made, they revert back to their original state.</td>
</tr>
<tr>
<td>CM-6(1)</td>
<td>Configuration Settings</td>
<td>Configuration Management</td>
<td>The puppet master is the central point of management for a SIMP system. While not required, the puppet master usually hosts a kickstart server so that clients are built the same every time.</td>
</tr>
<tr>
<td>CM-6(2)</td>
<td>Configuration Settings</td>
<td>Configuration Management</td>
<td>Puppet is not intended to be a security mechanism to prevent unauthorized changes to files. For files that are managed by puppet that changed, they will revert back to their original state. This control is really about protecting from unauthorized changes so access control to the puppet master should suffice to meet it. Changes to files are audited using auditd. Puppet changes are also audited. It’s up to the implementation to perform altering on those changes.</td>
</tr>
<tr>
<td>CM-6(3)</td>
<td>Configuration Settings</td>
<td>Configuration Management</td>
<td>This control is not fully met by SIMP. It’s important to point out that SIMP does provide logging of events to syslog. It’s currently up to the implementation to alert on those events.</td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>CM-7</td>
<td>Least Functionality</td>
<td>Configuration Management</td>
<td>There isn’t an explicit list of services that SIMP denies. Instead, it was built to provide only the essential functionality. Additional services get added only as needed.</td>
</tr>
<tr>
<td>CM-7(1)</td>
<td>Least Functionality (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-7(2)</td>
<td>Least Functionality (Control Enhancement)</td>
<td>Configuration Management</td>
<td>Applications can be installed, but new services will not run unless first registered with puppet. Additionally, puppet modules must be modified to ensure that IPtables opens up the necessary services. Minimally, for a service to remain active, it must be registered with puppet or the svckill.rb script will stop them. To be clear, there is nothing in SIMP that prevents the installation of RPMs (from the command line or YUM).</td>
</tr>
<tr>
<td>CM-7(3)</td>
<td>Least Functionality (Control Enhancement)</td>
<td>Configuration Management</td>
<td>The registration process for ports, protocols, and services are handled via puppet.</td>
</tr>
<tr>
<td>CM-8</td>
<td>Information System Component Inventory</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-8(1)</td>
<td>Information System Component Inventory (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-8(2)</td>
<td>Information System Component Inventory (Control Enhancement)</td>
<td>Configuration Management</td>
<td>To the extent possible, puppet tracks clients that are within it’s control. It’s not meant to be a true inventory mechanism.</td>
</tr>
<tr>
<td>CM-8(3)</td>
<td>Information System Component Inventory (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-8(4)</td>
<td>Information System Component Inventory (Control Enhancement)</td>
<td>Configuration Management</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Proce-</td>
<td>Awareness and Training</td>
<td>Method</td>
</tr>
<tr>
<td></td>
<td>dures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM-8(5)</td>
<td>Information System Component Inventory (Control</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM-8(6)</td>
<td>Information System Component Inventory (Control</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM-9</td>
<td>Configuration Management Plan</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td>CM-9(1)</td>
<td>Configuration Management Plan (Control Enhance-</td>
<td>Configuration Management</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ment)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CP-1</td>
<td>Contingency Planning Policy and Procedures</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-2</td>
<td>Contingency Plan</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-2(1)</td>
<td>Contingency Plan (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-2(2)</td>
<td>Contingency Plan (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-2(3)</td>
<td>Contingency Plan (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-2(4)</td>
<td>Contingency Plan (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-2(5)</td>
<td>Contingency Plan (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-2(6)</td>
<td>Contingency Plan (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-3</td>
<td>Contingency Training</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-3(1)</td>
<td>Contingency Training (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-3(2)</td>
<td>Contingency Training (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-4</td>
<td>Contingency Plan Testing and Exercises</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-4(1)</td>
<td>Contingency Plan Testing and Exercises (Control</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CP-4(2)</td>
<td>Contingency Plan Testing and Exercises (Control</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CP-4(3)</td>
<td>Contingency Plan Testing and Exercises (Control</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CP-6</td>
<td>Alternate Storage Site</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-6(1)</td>
<td>Alternate Storage Site (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>CP-6(2)</td>
<td>Alternate Storage Site (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-6(3)</td>
<td>Alternate Storage Site (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-7</td>
<td>Alternate Processing Site</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-7(1)</td>
<td>Alternate Processing Site (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-7(2)</td>
<td>Alternate Processing Site (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-7(3)</td>
<td>Alternate Processing Site (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-7(4)</td>
<td>Alternate Processing Site (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-7(5)</td>
<td>Alternate Processing Site (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-8</td>
<td>Telecommunications Services</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-8(1)</td>
<td>Telecommunications Services (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-8(2)</td>
<td>Telecommunications Services (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-8(3)</td>
<td>Telecommunications Services (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-8(4)</td>
<td>Telecommunications Services (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-9</td>
<td>Information System Backup</td>
<td>Contingency Planning</td>
<td>The BackupPC module is not currently available in SIMP 5.0.</td>
</tr>
<tr>
<td>CP-9(1)</td>
<td>Information System Backup (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-9(2)</td>
<td>Information System Backup (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-9(3)</td>
<td>Information System Backup (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-9(5)</td>
<td>Information System Backup (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
</tbody>
</table>

**Continued on next page**
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>CP-9(6)</td>
<td>Information System Backup (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-10</td>
<td>Information System Recovery and Reconstitution</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-10(1)</td>
<td>Information System Recovery and Reconstitution (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-10(2)</td>
<td>Information System Recovery and Reconstitution (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-10(3)</td>
<td>Information System Recovery and Reconstitution (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-10(4)</td>
<td>Information System Recovery and Reconstitution (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-10(5)</td>
<td>Information System Recovery and Reconstitution (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>CP-10(6)</td>
<td>Information System Recovery and Reconstitution (Control Enhancement)</td>
<td>Contingency Planning</td>
<td></td>
</tr>
<tr>
<td>IR-1</td>
<td>Incident Response Policy and Procedures</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-2</td>
<td>Incident Response Training</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-2(1)</td>
<td>Incident Response Training (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-2(2)</td>
<td>Incident Response Training (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-3</td>
<td>Incident Response Testing and Exercises</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-3(1)</td>
<td>Incident Response Testing and Exercises (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-4</td>
<td>Incident Handling</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-4(1)</td>
<td>Incident Handling (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
</tbody>
</table>

The BackupPC module is not currently available in SIMP 5.0.
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td>If an implementation chooses, they can leverage puppet’s ability to reconfigure systems as part of incident response. While puppet is not intended to be a security product, its features can help provide security functionality such as dynamic reconfigurations.</td>
</tr>
<tr>
<td>IR-4(2)</td>
<td>Incident Handling (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-4(3)</td>
<td>Incident Handling (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-4(4)</td>
<td>Incident Handling (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-4(5)</td>
<td>Incident Handling (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-5</td>
<td>Incident Monitoring</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-5(1)</td>
<td>Incident Monitoring (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-6</td>
<td>Incident Reporting</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-6(1)</td>
<td>Incident Reporting (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-6(2)</td>
<td>Incident Reporting (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-7</td>
<td>Incident Response Assistance</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-7(1)</td>
<td>Incident Response Assistance (Control Enhancement)</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>IR-8</td>
<td>Incident Response Plan</td>
<td>Incident Response</td>
<td></td>
</tr>
<tr>
<td>MA-1</td>
<td>System Maintenance Policy and Procedures</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-2</td>
<td>Controlled Maintenance</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-2(1)</td>
<td>Controlled Maintenance (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-2(2)</td>
<td>Controlled Maintenance (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-3</td>
<td>Maintenance Tools</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-3(1)</td>
<td>Maintenance Tools (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-3(2)</td>
<td>Maintenance Tools (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-3(3)</td>
<td>Maintenance Tools (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Pro-</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td></td>
<td>cedures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MA-3(4)</td>
<td>Maintenance Tools (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-4</td>
<td>Non-Local Maintenance</td>
<td>Maintenance</td>
<td>Remote maintenance</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>can be performed on</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SIMP using SSH or</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>direct console</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>access. SSH sessions</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>are tracked and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>logged using the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>security features</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>built into SIMP.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Console access</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>requires someone to</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>have access to the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>physical (or virtual)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>console along with</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the root password.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Auditing of those</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>actions also occurs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>in accordance with</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the configured audit</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>policy. It’s up to</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the implementation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>to decide how to</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>distribute authenti-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>cation information</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>for remote mainte-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>nance.</td>
</tr>
<tr>
<td>MA-4(1)</td>
<td>Non-Local Maintenance</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Control Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MA-4(2)</td>
<td>Non-Local Maintenance</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Control Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MA-4(3)</td>
<td>Non-Local Maintenance</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Control Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MA-4(4)</td>
<td>Non-Local Maintenance</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Control Enhancement)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MA-4(5)</td>
<td>Non-Local Maintenance</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Control Enhancement)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
## Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Method Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AF-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>MA-4(6)</td>
<td>Non-Local Maintenance (Control Enhancement)</td>
<td>Maintenance</td>
<td>Remote maintenance is performed using SSH. SSH inherently provides confidentiality and integrity of data while in transit.</td>
</tr>
<tr>
<td>MA-4(7)</td>
<td>Non-Local Maintenance (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-5</td>
<td>Maintenance Personnel</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-5(1)</td>
<td>Maintenance Personnel (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-5(2)</td>
<td>Maintenance Personnel (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-5(3)</td>
<td>Maintenance Personnel (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-5(4)</td>
<td>Maintenance Personnel (Control Enhancement)</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MA-6</td>
<td>Timely Maintenance</td>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>MP-1</td>
<td>Media Protection Policy and Procedures</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-2</td>
<td>Media Access</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-2(1)</td>
<td>Media Access (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-2(2)</td>
<td>Media Access (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-4</td>
<td>Media Storage</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-5</td>
<td>Media Transport</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-5(1)</td>
<td>Media Transport (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-5(2)</td>
<td>Media Transport (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-5(3)</td>
<td>Media Transport (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-5(4)</td>
<td>Media Transport (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-6</td>
<td>Media Sanitization</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-6(1)</td>
<td>Media Sanitization (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-6(2)</td>
<td>Media Sanitization (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-6(3)</td>
<td>Media Sanitization (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-6(4)</td>
<td>Media Sanitization (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>MP-6(5)</td>
<td>Media Sanitization (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>MP-6(6)</td>
<td>Media Sanitization (Control Enhancement)</td>
<td>Media Protection</td>
<td></td>
</tr>
<tr>
<td>PE-1</td>
<td>Physical and Environmental Protection Policy and Procedures</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-2</td>
<td>Physical Access Authorizations</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-2(1)</td>
<td>Physical Access Authorizations (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-2(2)</td>
<td>Physical Access Authorizations (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-2(3)</td>
<td>Physical Access Authorizations (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-3</td>
<td>Physical Access Control</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-3(1)</td>
<td>Physical Access Control (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-3(2)</td>
<td>Physical Access Control (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-3(3)</td>
<td>Physical Access Control (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-3(4)</td>
<td>Physical Access Control (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-3(5)</td>
<td>Physical Access Control (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-3(6)</td>
<td>Physical Access Control (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-4</td>
<td>Access Control for Transmission Medium</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-5</td>
<td>Access Control for Output Devices</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-6</td>
<td>Monitoring Physical Access</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-6(1)</td>
<td>Monitoring Physical Access (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-6(2)</td>
<td>Monitoring Physical Access (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-7</td>
<td>Visitor Control</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-7(1)</td>
<td>Visitor Control (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>Control ID</td>
<td>Control Name</td>
<td>Control Family</td>
<td>SIMP Implementation Method</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------------------</td>
<td>---------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>AI-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>PE-7(2)</td>
<td>Visitor Control (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-8</td>
<td>Access Records</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-8(1)</td>
<td>Access Records (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-8(2)</td>
<td>Access Records (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-9</td>
<td>Power Equipment and Power Cabling</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-9(1)</td>
<td>Power Equipment and Power Cabling (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-9(2)</td>
<td>Power Equipment and Power Cabling (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-10</td>
<td>Emergency Shutoff</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-10(1)</td>
<td>Emergency Shutoff (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-11</td>
<td>Emergence Power</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-11(1)</td>
<td>Emergence Power (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-11(2)</td>
<td>Emergence Power (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-12</td>
<td>Emergency Lighting</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-12(1)</td>
<td>Emergency Lighting (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-13</td>
<td>Fire Protection</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-13(1)</td>
<td>Fire Protection (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-13(2)</td>
<td>Fire Protection (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-13(3)</td>
<td>Fire Protection (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-13(4)</td>
<td>Fire Protection (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-14</td>
<td>Temperature and Humidity Controls</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-14(1)</td>
<td>Temperature and Humidity Controls (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>PE-14(2)</td>
<td>Temperature and Humidity Controls (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-15</td>
<td>Water Damage Protection</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-15(1)</td>
<td>Water Damage Protection (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-16</td>
<td>Delivery and Removal</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-17</td>
<td>Alternate Work Site</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-18</td>
<td>Location of Information System Components</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-18(1)</td>
<td>Location of Information System Components (Control Enhancement)</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>PE-19</td>
<td>Information Leakage</td>
<td>Physical and Environmental Protection</td>
<td></td>
</tr>
<tr>
<td>SI-1</td>
<td>System and Information Integrity Policy and Procedures</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td>Patches that are part of the software base for SIMP are tested within the development environment. There is automated testing that is constantly being extended to test more features. There are times that patches to the base operating system (Centos or RedHat) are needed to resolve issues in SIMP. Those are also tested at build time, but require additional testing by implementations as patches are released from vendors. It’s also important to note that SIMP is packaged and delivered decoupled with the operating system source files. It’s up to the implementation to test vendor specific patches that are not part of the SIMP code base. Flaws are tracked using the software project management tool Redmine.</td>
</tr>
<tr>
<td>SI-2(1)</td>
<td>Flaw Remediation (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-2(2)</td>
<td>Flaw Remediation (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-2(3)</td>
<td>Flaw Remediation (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-2(4)</td>
<td>Flaw Remediation (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td>SIMP uses the yellowdog update manager (YUM) to deliver software patches to clients. Each installation usually has at least one YUM repository. There is also a cronjob running that runs once per day. It’s the responsibility of the implementation to get patches to the yum server. Once they are there, the cron job will perform a yum update and the patches will be applied.</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>SI-3</td>
<td>Malicious Code Protection</td>
<td>System and Information Integrity</td>
<td>SIMP has modules available for mcafee and ClamAV. The ClamAV implementations need need to provide their own version of the mcafee software for the module to work. That module comes with the ability to sync data updates to clients via rsync. The module does NOT specify how often and what files systems should be scanned. SIMP also implements the open source tool chkrootkit that comes installed by default.</td>
</tr>
<tr>
<td>SI-3(1)</td>
<td>Malicious Code Protection (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td>The provided anti-virus modules are installed via puppet modules. Those modules include the ability to sync data file updates via rsync. Therefore, all management of malicious code detection is done centrally.</td>
</tr>
<tr>
<td>SI-3(2)</td>
<td>Malicious Code Protection (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-3(3)</td>
<td>Malicious Code Protection (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-3(4)</td>
<td>Malicious Code Protection (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-3(5)</td>
<td>Malicious Code Protection (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-3(6)</td>
<td>Malicious Code Protection (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4</td>
<td>Information System Monitoring Tools and Techniques</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(1)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(2)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>SI-4(3)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(4)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(5)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(6)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(7)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(8)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(9)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(10)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(11)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(12)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(13)</td>
<td>Information System Monitoring Tools and Techniques (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Pro-</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>SI-4(14)</td>
<td>Information System Monitoring Tools and Techni-</td>
<td>System and Information In-</td>
<td></td>
</tr>
<tr>
<td>SI-4(15)</td>
<td>quees (Control Enhancement)</td>
<td>tegrity</td>
<td></td>
</tr>
<tr>
<td>SI-4(16)</td>
<td>Information System Monitoring Tools and Techni-</td>
<td>System and Information In-</td>
<td></td>
</tr>
<tr>
<td>SI-4(17)</td>
<td>quees (Control Enhancement)</td>
<td>tegrity</td>
<td></td>
</tr>
<tr>
<td>SI-5</td>
<td>System Alerts, Advisories, and Directives</td>
<td>System and Information In-</td>
<td></td>
</tr>
<tr>
<td>SI-5(1)</td>
<td>System Alerts, Advisories, and Directives (Con-</td>
<td>tegrity</td>
<td></td>
</tr>
<tr>
<td>SI-6</td>
<td>Security Functionality Verification</td>
<td>System and Information In-</td>
<td></td>
</tr>
</tbody>
</table>

The only part of the control (a) that is met by SIMP, is the tracking of security alerts for products that are part of the code base. The development team subscribes to message boards for the main products (puppet) that are part of the packaging. Red-Hat/Centos advisories are also tracked out of necessity but since ALL the OS files are not part of SIMP delivery, patches are not our direct responsibility.

SIMP comes with an optional module to install and perform regular runs of the SCAP-Security-Guide (the checks for RHEL 7 are not yet complete/finalized). Doing so will report (for a user defined frequency) OVAL results of security settings of a host against SSG recommendations.

Continued on next page
## Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td>SIMP comes with an optional module to install and perform regular runs of the SCAP-Security-Guide. Doing so will report (for a user defined frequency) OVAL results of security settings of a host against SSG recommendations.</td>
</tr>
<tr>
<td>SI-6(1)</td>
<td>Security Functionality Verification (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td>SIMP comes with an optional module to install and perform regular runs of the SCAP-Security-Guide. Doing so will report (for a user defined frequency) OVAL results of security settings of a host against SSG recommendations.</td>
</tr>
<tr>
<td>SI-6(2)</td>
<td>Security Functionality Verification (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td>SIMP comes with an optional module to install and perform regular runs of the SCAP-Security-Guide. Doing so will report (for a user defined frequency) OVAL results of security settings of a host against SSG recommendations.</td>
</tr>
<tr>
<td>SI-6(3)</td>
<td>Security Functionality Verification (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td>SIMP comes with an optional module to install and perform regular runs of the SCAP-Security-Guide. Doing so will report (for a user defined frequency) OVAL results of security settings of a host against SSG recommendations.</td>
</tr>
<tr>
<td>SI-7</td>
<td>Software and Information Integrity</td>
<td>System and Information Integrity</td>
<td>SIMP comes with AIDE installed. Puppet also serves the purpose of checking the integrity of files. During each client run, a change in file integrity means the file needs to be restored to its original state.</td>
</tr>
<tr>
<td>SI-7(1)</td>
<td>Software and Information Integrity (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td>AIDE baselines are not performed beyond initial install unless otherwise configured. Implementations can re-baseline the database.</td>
</tr>
<tr>
<td>SI-7(2)</td>
<td>Software and Information Integrity (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>SI-7(3)</td>
<td>Software and Information Integrity (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td>AIDE is managed by puppet and is therefore centrally managed.</td>
</tr>
<tr>
<td>SI-7(4)</td>
<td>Software and Information Integrity (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-8</td>
<td>Spam Protection</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-8(1)</td>
<td>Spam Protection (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-8(2)</td>
<td>Spam Protection (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-9</td>
<td>Information Input Restrictions</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-10</td>
<td>Information Input Validation</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-11</td>
<td>Error Handling</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-13</td>
<td>Predictable Failure Prevention</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-13(1)</td>
<td>Predictable Failure Prevention (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-13(2)</td>
<td>Predictable Failure Prevention (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-13(3)</td>
<td>Predictable Failure Prevention (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td>SI-13(4)</td>
<td>Predictable Failure Prevention (Control Enhancement)</td>
<td>System and Information Integrity</td>
<td></td>
</tr>
<tr>
<td><strong>Control ID</strong></td>
<td><strong>Control Name</strong></td>
<td><strong>Control Family</strong></td>
<td><strong>SIMP Implementation Method</strong></td>
</tr>
<tr>
<td><strong>Control ID</strong></td>
<td><strong>Control Name</strong></td>
<td><strong>Control Family</strong></td>
<td><strong>SIMP Implementation Method</strong></td>
</tr>
<tr>
<td><strong>CA-1</strong></td>
<td>Security Assessment and Authorization Policies</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td><strong>CA-2</strong></td>
<td>Security Assessments</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td><strong>CA-2(1)</strong></td>
<td>Security Assessments (Control Enhancement)</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td><strong>CA-2(2)</strong></td>
<td>Security Assessments (Control Enhancement)</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td><strong>CA-3</strong></td>
<td>Information System Connections</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td><strong>CA-3(1)</strong></td>
<td>Information System Connections (Control Enhancement)</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td>Control ID</td>
<td>Control Name</td>
<td>Control Family</td>
<td>SIMP Implementation</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------</td>
<td>------------------------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td>Method</td>
</tr>
<tr>
<td>CA-3(2)</td>
<td>Information System Connections (Control Enhancement)</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td>CA-5</td>
<td>Plan of Action and Milestones</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td>CA-5(1)</td>
<td>Plan of Action and Milestones (Control Enhancement)</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td>CA-6</td>
<td>Security Authorization</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td>CA-7</td>
<td>Continuous Monitoring</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td>CA-7(1)</td>
<td>Continuous Monitoring (Control Enhancement)</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td>CA-7(2)</td>
<td>Continuous Monitoring (Control Enhancement)</td>
<td>Security Assessment and Authorization</td>
<td></td>
</tr>
<tr>
<td>PI-1</td>
<td>Security Planning Policy and Procedures</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PL-2</td>
<td>System Security Plan</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PL-2(1)</td>
<td>System Security Plan (Control Enhancement)</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PL-2(2)</td>
<td>System Security Plan (Control Enhancement)</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PL-4</td>
<td>Rules of Behavior</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PL-4(1)</td>
<td>Rules of Behavior (Control Enhancement)</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PL-5</td>
<td>Privacy Impact Assessment</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PL-6</td>
<td>Security-Related Activity</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PS-1</td>
<td>Personnel Security Policy and Procedures</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PS-2</td>
<td>Position Categorization</td>
<td>Planning</td>
<td></td>
</tr>
<tr>
<td>PS-3(2)</td>
<td>Personnel Screening (Control Enhancement)</td>
<td>Planning</td>
<td></td>
</tr>
</tbody>
</table>

The SIMP installation manual provides instructions for the installation of the product in a manner that is compliant with a multitude of security controls.

Security Plans are provided for specific implementations. The SIMP team will continue to develop security documentation that can be used as a resource for implementation specific System Security Plans.

TODO: Develop SIMP specific SSP.
### Table 3.5 – continued from previous page

<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>RA-1</td>
<td>Risk Assessment Policy and Procedures</td>
<td>Risk Assessment</td>
<td></td>
</tr>
<tr>
<td>RA-2</td>
<td>Security Categorization</td>
<td>Risk Assessment</td>
<td></td>
</tr>
<tr>
<td>RA-3</td>
<td>Risk Assessment</td>
<td>Risk Assessment</td>
<td></td>
</tr>
<tr>
<td>RA-5</td>
<td>Vulnerability Scanning</td>
<td>Risk Assessment</td>
<td></td>
</tr>
<tr>
<td>RA-5(1)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td>The SIMP team performs a variety of security testing as part of the development process. Compliance and configuration checking is done using SSG. SIMP makes every effort to address problems discovered by these tools. Some configuration settings will not align with tools since the product was meant to be used for operational settings where some security features cause a loss in functionality. Implementations have the option of further hardening their system further at the risk of losing some functionality.</td>
</tr>
<tr>
<td>RA-5(2)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td>SCAP-Security-Guide is the two primary tool used to check for suspected configuration errors. Puppet also continues to protect clients against unwanted changes.</td>
</tr>
<tr>
<td>RA-5(3)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td>SCAP-Security-Guide is the two primary tool used to check for suspected configuration errors. Puppet also continues to protect clients against unwanted changes.</td>
</tr>
<tr>
<td>RA-5(4)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td>Regular vulnerability scanning is performed during development of SIMP.</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>RA-5(5)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td>The compliance tools require that privileged accounts be used to perform testing.</td>
</tr>
<tr>
<td>RA-5(6)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td></td>
</tr>
<tr>
<td>RA-5(7)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td>Only part of this requirement is met. SIMP can detect when any software is installed via auditd and syslog. Services that are not registered with puppet will not operate without user intervention. Those changes are also audited. SIMP does not provide the ability to alert on those actions, however, Logstash filters or Elasticsearch queries can be applied if needed.</td>
</tr>
<tr>
<td>RA-5(8)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td></td>
</tr>
<tr>
<td>RA-5(9)</td>
<td>Vulnerability Scanning (Control Enhancement)</td>
<td>Risk Assessment</td>
<td></td>
</tr>
<tr>
<td>SA-1</td>
<td>System and Services Acquisition Policy and Procedures</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-2</td>
<td>Allocation of Resources</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-3</td>
<td>Life Cycle Support</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-4</td>
<td>Acquisitions</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-4(1)</td>
<td>Acquisitions (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-4(2)</td>
<td>Acquisitions (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-4(3)</td>
<td>Acquisitions (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-4(4)</td>
<td>Acquisitions (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-4(5)</td>
<td>Acquisitions (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-4(6)</td>
<td>Acquisitions (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
<th>Control Family</th>
<th>SIMP Method Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
<td>Awareness and Training</td>
<td></td>
</tr>
<tr>
<td>SA-4(7)</td>
<td>Acquisitions (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-5</td>
<td>Information System Documentation</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-5(1)</td>
<td>Information System Documentation (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-5(2)</td>
<td>Information System Documentation (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-5(3)</td>
<td>Information System Documentation (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-5(4)</td>
<td>Information System Documentation (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-5(5)</td>
<td>Information System Documentation (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-6</td>
<td>Software Usage Restrictions</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-6(1)</td>
<td>Software Usage Restrictions</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-7</td>
<td>User Installed Software</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-8</td>
<td>Security Engineering Principles</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-9</td>
<td>External Information System Services</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-9(1)</td>
<td>External Information System Services (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-10</td>
<td>Developer Configuration Management</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-10(1)</td>
<td>Developer Configuration Management (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-10(2)</td>
<td>Developer Configuration Management (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-11</td>
<td>Developer Security Testing</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-11(1)</td>
<td>Developer Security Testing (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
<tr>
<td>SA-11(2)</td>
<td>Developer Security Testing (Control Enhancement)</td>
<td>System and Service Acquisition</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Control ID</th>
<th>Control Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-1</td>
<td>Security Awareness and Training Policy and Procedures</td>
</tr>
<tr>
<td>SA-11(3)</td>
<td>Developer Security Testing (Control Enhancement)</td>
</tr>
<tr>
<td>SA-12</td>
<td>Supply Chain Protection</td>
</tr>
<tr>
<td>SA-12(1)</td>
<td>Supply Chain Protection (Control Enhancement)</td>
</tr>
<tr>
<td>SA-12(2)</td>
<td>Supply Chain Protection (Control Enhancement)</td>
</tr>
<tr>
<td>SA-12(3)</td>
<td>Supply Chain Protection (Control Enhancement)</td>
</tr>
<tr>
<td>SA-12(4)</td>
<td>Supply Chain Protection (Control Enhancement)</td>
</tr>
<tr>
<td>SA-12(5)</td>
<td>Supply Chain Protection (Control Enhancement)</td>
</tr>
<tr>
<td>SA-12(6)</td>
<td>Supply Chain Protection (Control Enhancement)</td>
</tr>
<tr>
<td>SA-12(7)</td>
<td>Supply Chain Protection (Control Enhancement)</td>
</tr>
<tr>
<td>SA-13</td>
<td>Trustworthiness</td>
</tr>
<tr>
<td>SA-14</td>
<td>Critical Information System Components</td>
</tr>
<tr>
<td>SA-14(1)</td>
<td>Critical Information System Components (Control Enhancement)</td>
</tr>
</tbody>
</table>

Table: Management Controls
Legal Notice This material may only be distributed subject to the terms and conditions set forth in the GNU Free Documentation License (GFDL), V1.2 or later (the latest version is available at “http://www.gnu.org/licenses/fdl.txt.

The SIMP Development Team makes no representation about the suitability of the SIMP product for any purpose. It is provided “as is” without expressed or implied warranty. If SIMP is modified in any way, except for designed customization, identify the new copy as a variant of SIMP.

Additional products are distributed as part of the SIMP suite, including but not limited to, Puppet, CentOS, and Red Hat Enterprise Linux (RHEL). By using SIMP, the user agrees to abide by the licenses for the included products, particularly the purchase of RHEL licenses if using the Red Hat version of SIMP.

At this time, RHEL is the only product requiring payment for a commercial license in the SIMP infrastructure.
Contact

If you have questions please contact the SIMP team. simp@simp.org
If you are looking for assistance please email the SIMP mailing lists. simp@simp.org
Indices and tables

- genindex
- search
Index

A
Access Control List (ACL), 87
Advanced Intrusion Detection Environment (AIDE), 22, 87
Auditd, 87

C
Central Processing Unit (CPU), 22
Community Enterprise Operating System (CentOS), 22, 88

D
Domain Name System (DNS), 22, 88
Dynamic Host Configuration Protocol (DHCP), 22, 88

F
Fully Qualified Domain Name (FQDN), 22, 88

H
Hard Disk Drive (HDD), 22, 88

I
Internet Protocol 6 Tables (IP6tables), 22, 88
Internet Protocol Tables (IPtables), 22, 88

K
Kerberos (Krb5), 88
Key Distribution Center (KDC), 88

L
Lightweight Directory Access Protocol (LDAP), 22, 88

N
Network Address Translation (NAT), 88
Network File System (NFS), 88

P
Parallel Secure Shell (PSSH), 88
Pluggable Authentication Modules (PAM), 22, 88

Practical Extraction and Report Language (PERL), 88
Preboot Execution Environment (PXE), 23, 88
Privacy Enhanced Mail (PEM), 23, 88
Public Key Infrastructure (PKI), 23, 88
Puppet, 23, 89

R
Random Access Memory (RAM), 23, 89
Red Hat, 23, 89
Red Hat Enterprise Linux (RHEL), 23, 89
Resource Package Manager (RPM), 89
RSA, 89
Ruby, 23, 89

S
Secure Shell (SSH), 23, 89
Secure Sockets Layer (SSL), 23, 89
SIMP, 23, 89
SSH File Transfer Protocol (SFTP), 89
Sudosh, 89

T
Transport Layer Security (TLS), 23
Transport Layer Security (TSL), 89
Trivial File Transfer Protocol (TFTP), 23, 89
TTY, 90

V
Virtual Machine (VM), 23, 90
Virtual Network Computing (VNC), 90

W
Wide Area Network (WAN), 90

Y
Yellowdog Updater, Modified (YUM), 23, 90