The Machine

System Administration and User Manual

Software Version 2.0.0

Table of Contents

1   Introduction

1.1   Document Purpose

This document describes the installation, management and use of The Machine, version 2.0.0, released on 02/08/2016.

1.2   System Purpose

The Machine is an automation and server control system. a sudoers file management system that removes the need for administrators to manage the sudoers file directly. By managing the sudoers file with software, the risk of user interaction and potential sudo breakage is removed as the system monitors its own sudoers file writes to ensure syntactic accuracy. As an additional benefit of shifting management to a separate system, sudoers file edits are now audited, ensuring accountability, and changes must be approved by a second administrator, ensuring accuracy.

1.3   Definitions

Term/Acronym	Definition/Full Description
TM	The Machine.
DSMS	Distributed Sudoers Management System.
DHCP	Dynamic Host Control Protocol.
RHEL	Red Hat Enterprise Linux.
TM Server	The server or servers where the The Machine is, or will be, installed.
Remote Server	The server or servers which will receive the sudoers, DNS, or Reverse Proxy files produced by the TM Server. Note that the TM Server may also be classified as a 'Remote Server' if the TM Server's local sudoers file will be managed by the TM Server software.
DSMS System	A name for the collective DSMS software, including web files, sudoers generation mechanism and distribution mechanism.
CDSF	Currently Distributed Sudoers File.
SSH	Secure Shell. A secure method of communication between two systems.
SFTP	Secure File Transfer Protocol. A secure method of file transfer between two systems. SFTP is a subsystem of SSH.
chroot (chroot jail)	Change Root. A system security enhancement to change a user's root directory, which restricts what that user can access on a system.
CPAN	Comprehensive Perl Archive Network. The CPAN network is a collection of publically available perl modules.
DNS	Domain Name System.
BIND	Berkeley Internet Name Domain server. The DNS server whose configuration TM creates.
FQDN	Fully Qualified Domain Name.
MD5	Message Digest. Used to verify data integrity of the CDSF, and during the initial DSMS System deployment.

CGI	Common Gateway Interface.
HTML	HyperText Markup Language.
NTP	Network Time Protocol.
LDAP	Lightweight Directory Access Protocol
AD	Active Directory
DES	Data Encryption Standard
AES	Advanced Encryption Standard
VM	Virtual Machine

1.4   Document Standards

This document follows formatting standards throughout to make explanations clearer. Some of these explanations include the use of colour for clarity or highlighting, so it is advised that you do not use a monochrome copy of this document.

1.4.1   Commands

Commands are displayed in a grey box, with a blue border. The command that is meant to be run is highlighted in red. The text in black describes what the system is likely to return. Consider the following example:

```
echo 'Hello'
Hello
```

The command that you run should've been *echo 'Hello'*, as highlighted in red, and the response that the server should've given is Hello, as highlighted in black. Each time you are required to run a command, please read the description carefully, as running the command may not be required depending on certain pre-existing conditions or your intentions for the system. Additionally, some commands must be run locally on the TM Server, while others must be run on Remote Servers that TM will write the sudoers file to, so understanding which system the command is meant for is important.

1.4.2   Command Variables

Some commands may require modification before applying them to a system to make them applicable to your setup. Command components that may require modification are highlighted in purple. Consider the following example:

```
useradd -m -d /home/transport -s /bin/sh transport
```

The useradd command and its options are meant to be typed by the user, because it is highlighted in red. The username 'transport' is highlighted in purple, as this name may not be applicable to all systems and could require the user to change it to a value more appropriate to the system for which it is being applied.

1.4.3   Highlighting

Some outputs are important, or otherwise difficult to distinguish in a block of text, or difficult to describe in writing. These are often highlighted to facilitate the instructions or explanation. Consider the following example:

```
2048 94:3f:80:d5:48:45:92:b1:ea:aa:fe:c9:6e:bb:60:be /etc/ssh/ssh_host_rsa_key.pub (RSA)
```

The text highlighted in yellow is an important part of the instructions, but would be difficult to define in a text description alone. To avoid confusing the important text with the prefixed 2048 value, the important text is highlighted in yellow.

2   Installation

2.1   System Requirements

2.1.1   Host Server Requirements

You are welcome to attempt to run TM on any server that you think may be capable to run it. TM is made from a collection of common components and requires a common system to run it, often referred to as a LAMP stack - that is, Linux, Apache, MySQL, Perl. The System Requirements below should therefore be considered as confirmed working defaults rather than minimum requirements.

2.1.1.1   Red Hat Enterprise Linux

The system is known to work on Red Hat Enterprise Linux 6 and CentOS 6 or above. Systems with the same basic components to TM are confirmed working on Debian and Ubuntu based systems (Squeeze and 10.04 and above, respectively), but the repository package names may differ from this manual.

The TM Server Automated Installation process also expects the yum package manager and default RHEL directory locations, which are known to differ on Debian systems. For instance, the default Apache configuration file on RHEL based systems is at */etc/httpd/conf/httpd.conf*, whereas on Debian based systems it is located at */etc/apache2/apache2.conf*.

If you are comfortable and familiar with the differences and are capable of working around them, by all means use a different Linux distribution. If you get TM working stably and securely on a non-RHEL based system, please feedback your configuration and workarounds and this document can be expanded to include those.

You will require root access on the TM Server for the initial setup.

2.1.1.2   Apache 2

Apache, due to its widespread use and stability, is the only web server that TM has been tested and confirmed working on. Feel free to try a different web server, but it cannot be reasonably supported. If you get TM working stably and securely on a non-Apache based system, please feedback your configuration and workarounds and this document can be expanded to include those. Apache 2 should be installed from your distribution's base repository.

2.1.1.3   mod_ssl

mod_ssl is an Apache requirement for supporting SSL connections. mod_ssl should be installed from your distribution's base repository.

2.1.1.4   openssl

openssl is required to create self-signed certificates to enable Apache to serve data over a secure (HTTPS) connection. openssl is not required if you intend on providing your own certificate set. openssl should be installed from your distribution's base repository.

2.1.1.5   MySQL 5

MySQL (or MariaDB) is used by TM because of its common use, common expertise, easy maintainability and flexibility, as well as having licence requirements that do not incur a fiscal cost. TM uses the DBI interface between perl and MySQL, so if you wish to use a different database it should be straightforward, however this has not been tested and is not supported. If you get TM working stably and securely on a non-MySQL based system, please feedback your configuration and workarounds and this document can be expanded to include those. MySQL should be installed from your distribution's base repository.

2.1.1.6   visudo

visudo is required for the syntax checking mechanism of the DSMS sudoers build and deployment process. visudo should be installed from your distribution's base repository.

2.1.1.7   md5sum

md5sum is required for sudoers version control, as well as confirming a successful deployment by way of checksum. md5sum should be installed from your distribution's base repository.

2.1.1.8   cut

cut is required for several command line interactions. cut should be installed from your distribution's base repository.

2.1.1.9   cp

cp is required for several command line interactions, including sudoers version control and sudoers file restoration if a fault is detected. cp should be installed from your distribution's base repository.

2.1.1.10   head

head is required for several command line interactions. head should be installed from your distribution's base repository.

2.1.1.11   ls

ls is required for several command line interactions. ls should be installed from your distribution's base repository.

2.1.1.12   nmap

nmap is required for several command line interactions. nmap should be installed from your distribution's base repository.

2.1.1.13   perl 5.10

Perl 5.10 or above is required. Perl should be installed from your distribution's base repository.

2.1.1.14   perl Modules

It is recommended that the below perl modules are installed via CPAN where possible, as CPAN always installs the latest version which may include security and bug fixes that the modules available in the RHEL base repository have not yet been afforded due to package maintenance delays. However, if this is an offline system, you may have to install from a local RHEL base repository, or compile from source. To cover all situations, all required modules are included in the 'Perl Modules' directory in the root of the DSMS package.

CPAN is available via the perl-CPAN package in the base repository of RHEL, and depends on *'gcc'* to compile the modules. 'gcc' can be removed after the modules are installed.

2.1.1.15   perl Module - strict

strict is used to ensure that perl runs in the safest possible mode. It is part of the core module set, and so is included with perl 5.10. strict is required.

2.1.1.16   perl Module - DBI

DBI is used to interface with the database. It is included in the base set of RHEL packages as 'perl-DBI', can be installed via CPAN using *perl -MCPAN -e 'install DBI'* or can be compiled from source. The DBI module's source is included in the Perl Modules folder. DBI is required.

2.1.1.17   perl Module - HTML::Table

HTML::Table is used to dynamically build tables in the web interface. It can be installed via CPAN using *perl -MCPAN -e 'install HTML::Table'* or can be compiled from source. The HTML::Table module's source is included in the Perl Modules folder. HTML::Table is required.

2.1.1.18   perl Module - Digest::SHA

Digest::SHA is used to hash passwords by using the sha512_hex routine. It is included in the base set of RHEL packages as 'perl-Digest-SHA', can be installed via CPAN using *perl -MCPAN -e 'install Digest::SHA'* or can be compiled from source. The Digest::SHA module's source is included in the Perl Modules folder. Digest::SHA is required.

2.1.1.19   perl Module - POSIX

POSIX is used for time calculations and display by using the strftime routine. It is part of the core module set, and so is included with perl 5.10. POSIX is required.

2.1.1.20   perl Module - MIME::Lite

MIME::Lite is used to send account recovery emails. It is included in the base set of RHEL packages as 'perl-MIME-Lite', can be installed via CPAN using *perl -MCPAN -e 'install MIME::Lite'* or can be compiled from source. The MIME::Lite module's source is included in the Perl Modules folder. MIME::Lite is required.

2.1.1.21   perl Module - CGI

CGI is used to interface with the client's browser and display web pages, as well as for some authentication components. It is included in the base set of RHEL packages as 'perl-CGI', can be installed via CPAN using *perl -MCPAN -e 'install CGI'* or can be compiled from source. The CGI module's source is included in the Perl Modules folder. CGI is required.

2.1.1.22   perl Module - CGI::Carp

CGI::Carp is used to interface with the client's browser and display fatal errors through the routine fatalsToBrowser. It is included as part of the CGI module. CGI::Carp is required.

2.1.1.23   perl Module - CGI::Session

CGI::Session is used to interface with the client's browser and display web pages, as well as for some authentication components. It is included in the base set of RHEL packages as 'perl-CGI-Session', can be installed via CPAN using *perl -MCPAN -e 'install CGI::Session'* or can be compiled from source. The CGI::Session module's source is included in the Perl Modules folder. CGI::Session is required.

2.1.1.24   Perl Module - Crypt::DES

Crypt::DES is used to encrypt data. It can be installed via CPAN using *perl -MCPAN -e 'install Crypt::DES'* or can be compiled from source. The Crypt::DES module's source is included in the Perl Modules folder. Crypt::DES is required.

2.1.1.25   Perl Module - Crypt::Rijndael

Crypt::Rijndael is used to encrypt data. It can be installed via CPAN using *perl -MCPAN -e 'install Crypt::Rijndael'* or can be compiled from source. The Crypt::Rijndael module's source is included in the Perl Modules folder. Crypt::Rijndael is required.

2.1.1.26   perl Module - Date::Parse

Date::Parse is used for time calculations. It is included in the base set of RHEL packages as 'perl-DateTime', can be installed via CPAN using *perl -MCPAN -e 'install Date::Parse',* can be compiled from source or is included when you install Date::Time. The Date::Parse module's source is included in the Perl Modules folder. Date::Parse is required.

2.1.1.27   perl Module - Time::HiRes

Time::HiRes is used for time calculations. It is included in the base set of RHEL packages as 'perl-Time-HiRes', can be installed via CPAN using *perl -MCPAN -e 'install Time::HiRes'* or can be compiled from source. The Time::HiRes module's source is included in the Perl Modules folder. Time::HiRes is required.

2.1.1.28   perl Module – Time::Date

Time::Date is used for time calculations. It can be installed via CPAN using *perl -MCPAN -e 'install Time::Date'* or can be compiled from source. The Time::Date module's source is included in the Perl Modules folder. Time::Date is required.

2.1.1.29   perl Module - Text::Diff::HTML

Text::Diff::HTML is used for comparing the differences between two HTML outputs. It can be installed via CPAN using *perl -MCPAN -e 'install Text::Diff::HTML'* or can be compiled from source. The Text::Diff::HTML module's source is included in the Perl Modules folder. Text::Diff::HTML is required.

2.1.1.30   perl Module - Parallel::ForkManager

Parallel::ForkManager is used for multithreading parts of The Machine. It can be installed via CPAN using *perl -MCPAN -e 'install Parallel::ForkManager'* or can be compiled from source. The Parallel::ForkManager module's source is included in the Perl Modules folder. Parallel::ForkManager is required.

2.1.1.31   perl Module - Net::LDAP

Net::LDAP is used for authenticating access requests against an LDAP or AD server. It can be installed via CPAN using *perl -MCPAN -e 'install Net::LDAP'* or can be compiled from source. The Net::LDAP module's source is included in the Perl Modules folder. Net::LDAP is required if you wish to connect The Machine to an LDAP or AD server for authentication.

2.1.1.32   perl Module - Crypt::CBC

Crypt::CBC is used for encrypting and decrypting data. It can be installed via CPAN using *perl -MCPAN -e 'install Crypt::CBC'* or can be compiled from source. The Crypt::CBC module's source is included in the Perl Modules folder. Crypt::CBC is required.

2.1.1.33   perl Module - Net::IPv4Addr

Net::IPv4Addr is used for IP calculations. It can be installed via CPAN using *perl -MCPAN -e 'install Net::IPv4Addr'* or can be compiled from source. The Net::IPv4Addr module's source is included in the Perl Modules folder. Net::IPv4Addr is required.

2.1.1.34   perl Module - Net::IP::XS

Net::IP::XS is used for IP calculations. It can be installed via CPAN using *perl -MCPAN -e 'install Net::IP::XS'* or can be compiled from source. The Net::IP::XS module's source is included in the Perl Modules folder. Net::IP::XS is required.

2.1.1.35   perl Module - Net::Ping::External

Net::Ping::External is used for IP calculations. It can be installed via CPAN using *perl -MCPAN -e 'install Net::Ping::External'* or can be compiled from source. The Net::Ping::External module's source is included in the Perl Modules folder. Net::Ping::External is required.

2.1.1.36   perl Module - Net::SSH::Expect

Net::SFTP::Foreign is used for the secure distribution of sudoers files. It can be installed via CPAN using *perl -MCPAN -e 'install Net::SFTP::Foreign'* or can be compiled from source. The Net::SFTP::Foreign module's source is included in the Perl Modules folder. Net::SFTP::Foreign is required.

2.1.1.37 perl Module - Net::SFTP::Foreign

Net::SFTP::Foreign is used for the secure distribution of sudoers files. It can be installed via CPAN using *perl -MCPAN -e 'install Net::SFTP::Foreign'* or can be compiled from source. The Net::SFTP::Foreign module's source is included in the Perl Modules folder. Net::SFTP::Foreign is required.

2.1.1.38 perl Module - Git::Wrapper

Git::Wrapper is used for the secure distribution of sudoers files. It can be installed via CPAN using *perl -MCPAN -e 'install* Git::Wrapper*'* or can be compiled from source. The Git::Wrapper module's source is included in the Perl Modules folder. Git::Wrapper is required.

2.1.2   Remote Server Requirements

TM is designed so that the Remote Servers require only minimal changes to their existing configurations, and usually require no additional software. The following are minimum Remote Server requirements:

- RHEL 6 or CentOS 6 or above (other distributions may work, but are unsupported)
- SSH with an SFTP subsystem
- Reachable via SSH by the TM Server
- root access for the initial setup
- A working Cron system
- visudo (for a final consistency check during Cron)
- Local access to securely determine the server's host RSA key (not required, but advisable)

2.1.3   Client Requirements

TM is designed so that the clients require no changes to their existing configuration, and usually require no additional software. The following are minimum Client requirements:

- A screen resolution equal to or exceeding 1024x768
- A standards compliant browser, such as Firefox or Chrome (Internet Explorer is NOT supported)
- Can reach the TM Server via HTTPS (port 443)
- Cookie support is required for CGI authentication control
- Javascript support is required for toggle switches

2.2   TM Server Automated Installation

2.2.1   Automated Installer Tasks

The automated installation process assumes that the server has not previously been configured. It provides some checks before making changes, however it would be implausible to check every component of a server for potential conflicts. The onus is on the person performing the automated installation to be sure that installing TM on a server does not conflict with existing software or configuration.

The automated installation process performs the following actions:

- Execution Checks - The script ensures it is running in the extracted directory.

- OS Dependency Checks - The script checks to see if the OS is supported.

- Package Dependency Checks - A list of required Packages and Commands is held in the tar archive, and these must be present before the installation can progress.

- Check Perl Modules - The perl modules required to use this utility are verified to see if they are installed.

- Code Rollout - The perl code is distributed to Apache's public file directory.

- Update Apache Config - The Apache config files are updated with the site-specific configuration.

- Generate SSL Keys - The SSL Keys required for Apache are generated.

- Generate SQL Schema - The format of the Database and the User rights required are submitted.

- Network and Security Setup - The Network Security (IPTables) and OS Security (SELinux) are configured.

2.2.2   Automated Installer Interaction

When an error is encountered, the script will usually exit without prompt, but some failures can require interaction with the installer. The most commonly required interaction is confirmation of an action which may result in the destruction of existing configuration. If this occurs, you will be presented with the following actions to choose from:

- R - Run. This action runs the command requested, modifying the referenced existing configuration.

- S - Skip. Avoids the command but moves to the next step, leaving the referenced existing configuration as is.

- Q - Quit. This option exits the script and allows for further manual analysis.

The script can be re-run but will start from scratch. Any steps that can potentially overwrite data will prompt as above.

2.2.3   Extracting the Package

Upload the latest DSMS package, sudoers-release-1.10.0.tar.gz, to the /tmp directory on the TM Server through any means you wish. As root on the TM Server, run the following:

```
cd /tmp
tar -xzf sudoers-release-1.10.0.tar.gz
```

2.2.4   Running the Automated Installer

Included in the root directory of the tar archive is the *install_sudoers_util.sh* script. This script is designed to check various dependencies and facilitate the installation of the DSMS System.

The script should be initiated from the extracted tar archive directory. As root on the TM Server, run the following to initiate the automated installer:

```
cd /tmp/sudoers
/install_sudoers_util.sh
```

2.3   TM Server Manual Installation

2.3.1   Extracting the Package

Upload the latest DSMS package, sudoers-release-1.10.0.tar.gz, to the /tmp directory on the TM Server through any means you wish. As root on the TM Server, run the following:

```
cd /tmp
tar -xzf sudoers-release-1.10.0.tar.gz
```

2.3.2 File Integrity Checking

Before installing TM files, we must check them for integrity to ensure they're not corrupt or incomplete. As root on the TM Server, run the following:

```
cd /tmp/sudoers
md5sum -c checksums
```

You should inspect every returned line for any failures. A successful checksum returns an OK message for each matching file which looks like this:

```
./HTTP/index.cgi: OK
```

A checksum failure looks like this:

```
./HTTP/index.cgi: FAILED
```

If you identify any failed checksums, source a new sudoers-release-1.10.0.tar.gz file and begin the installation process again from Extracting the Package.

2.3.3 Moving the HTTP Files

After all the files checksum correctly, we need to move the files into the HTTP root directory. The HTTP root directory is usually */var/www/html*, but if you have a different HTTP root directory, you should use the directory that's appropriate to your system. As root on the TM Server, run the following:

```
mv /tmp/sudoers/HTTP/* /var/www/html
```

2.3.4 File Permissions

To improve security, we need to assign as restrictive permissions to the files as possible. The locations of one file and one folder are defined later in the Sudoers_Location (the file) and Sudoers_Storage (the folder) sections of the Common Parameter Configuration section. The defaults for these are 'sudoers' and 'sudoers-storage/' respectively, so in this example the default names will be used, however you may need to substitute these values with your custom changes if required. If you do not run SELinux, omit the last two lines. As root on the TM Server, run the following:

```
cd /var/www/html
touch sudoers
mkdir sudoers-storage/
chown root:apache *.cgi
chmod 650 *.cgi
chown root:apache common.pl register.pl checkin.pl
chmod 650 common.pl register.pl checkin.pl
chown root:root sudoers-build.pl distribution.pl
chmod 100 sudoers-build.pl distribution.pl
chown root:apache environmental-defaults sudoers
chmod 640 environmental-defaults sudoers
chown -R root:apache format.css favicon.ico resources/
chmod -R 440 format.css favicon.ico resources/
chown root:apache resources/ resources/imgs/ resources/imgs/buttons/
chmod 550 resources/ resources/imgs/ resources/imgs/buttons/
chown -R root:root sudoers-storage/
chmod -R 700 sudoers-storage/
```

```
semanage fcontext -a -t httpd_sys_script_exec_t "/var/www/html(/.*)?"
restorecon -vRF /var/www/html
```

2.3.5   Apache Configuration

Because TM uses perl as its main driver, Apache must be configured to recognise TM files as perl files for the system to function correctly. We'll also perform some general administrative tasks here, like setting the server name and server contact. As root, run the following on the TM Server:

```
sed -e 's/^DirectoryIndex/DirectoryIndex index.cgi/' -i /etc/httpd/conf/httpd.conf
sed -e 's/^ServerAdmin root@localhost/ServerAdmin ben@nwk1.com/' -I /etc/httpd/conf/httpd.conf
sed -e 's/^#ServerName/ServerName/' -i /etc/httpd/conf/httpd.conf
sed -e 's/^ServerName www.example.com:80/ServerName DSMS/' -i /etc/httpd/conf/httpd.conf
echo '
# Distributed Sudoers Management System CGI Handlers
AddHandler cgi-script .cgi .pl
<Files ~ "\.pl$">
    Options +ExecCGI
</Files>
<Files ~ "\.cgi$">
    Options +ExecCGI
</Files>
' >> /etc/httpd/conf/httpd.conf
/etc/init.d/httpd restart
```

2.3.6   Apache SSL Configuration

You are highly advised to use a secure connection to the DSMS System. If you already have a certificate set to apply to this server, you may skip this step. The command variables form part of the certificate variables - the values here don't really matter, but they do help to determine system ownership and provide a good starting point (i.e. a contact) for certificate renewal. The final variable, 3562, determines when this certificate will expire in days. 3562 is ten years, give or take a day. The certificate set is given the name 'DSMS' to avoid overwriting any existing certificates already in */etc/pki*; there is little reason to change this, unless you do happen to have existing certificates named DSMS. As root, run the following on the TM Server:

```
cd /tmp
openssl genrsa -out DSMS.key 4096
openssl req -new -key DSMS.key -out DSMS.csr<<EOF
NZ
IT Operations
IT Operations
Distributed Sudoers Management System
DSMS
DSMS
ben@nwk1.com


EOF
openssl x509 -req -days 3652 -in DSMS.csr -signkey DSMS.key -out DSMS.crt
cp DSMS.crt /etc/pki/tls/certs
rm -f DSMS.crt
cp DSMS.key /etc/pki/tls/private/DSMS.key
rm -f DSMS.key
cp DSMS.csr /etc/pki/tls/private/DSMS.csr
rm -f DSMS.csr
restorecon -vRF /etc/pki
```

Apache needs to be made aware of the new certificates. As root run the following on the TM Server; after modifying the HTTP root and default HTTPS port if applicable:

```
echo '
# Distributed Sudoers Management System SSL Configuration
<VirtualHost *:443>
    SSLEngine on
    SSLCertificateFile /etc/pki/tls/certs/DSMS.crt
    SSLCertificateKeyFile /etc/pki/tls/private/DSMS.key
    <Directory /var/www/html>
       AllowOverride All
    </Directory>
    DocumentRoot /var/www/html
    ServerName DSMS
</VirtualHost>
' > /etc/httpd/conf.d/DSMS.conf
/etc/init.d/httpd restart
```

2.3.7 MySQL Configuration

TM has been designed to separate system administrative data with sudoers data for increased security. By default, these two sets of data are stored in separate database schemas which make setting access more flexible and more robust. For instance, the Management database could be stored on a separate server to reduce risk of tampering from users with local access and elevated privileges. By default, the TM Server uses a local MySQL installation, although this is easily configurable to be a remote host. The examples below assume a local installation with the default schema and user names - see the Sudoers_Owner_ID

For changing the ownership of the sudoers file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

2.3.7.1 Sudoers_Group_ID

For changing the group ownership of the sudoers file after it's created by the DSMS build process, we need to specify a group. It is recommended to run the DSMS build process as a root cron job (as defined in Cron Configuration), but Apache will need to reach this file to display its live contents on the web panel. If you modified Apache based on the Apache Configuration section, the group ownership should usually be 'apache'. However, on some systems, Apache Server doesn't run as the 'apache' user, such as when it runs as 'httpd', so you must specify the appropriate group ownership for TM to read the sudoers file.

Defaults:

```
my $Group = 'apache';
```

2.3.7.2 DNS_Zone_Master_File

This is the zone master file use for defining zones. You should include the full path.

Defaults:

```
my $DNS_Zone_Master_File = '/etc/bind/named.conf.local';
```

2.3.7.3 DNS_Internal_Location

This is the path that the system writes the temporary Internal DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_Internal_Location = '/etc/bind/master-internal';
```

2.3.7.4 DNS_External_Location

This is the path that the system writes the temporary External DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_External_Location = '/etc/bind/master-external';
```

2.3.7.5 DNS_Internal_SOA

This is the SOA data that's written at the top of the Internal DNS file. It's recommended to keep the serial as default.

Defaults:

```
my $Email = 'postmaster@example.com';
my $TTL = '86400';              # 1 day
my $Serial = `date +%s`;        # Epoch
      $Serial =~ s/\n//;
my $Refresh = '10800';          # 3 hours
my $Retry = '3600';             # 1 hour
my $Expire = '2419200';         # 4 weeks
my $Minimum = '86400';          # 1 day
my $NS1 = 'ns1.example.com';
my $NS2 = 'ns2.example.com';
my $NS3 = 'ns3.example.com';
```

2.3.7.6 DNS_External_SOA

This is the SOA data that's written at the top of the External DNS file. It's recommended to keep the serial as default.

Defaults:

```
my $Email = 'postmaster@example.com';
my $TTL = '86400';              # 1 day
my $Serial = `date +%s`;        # Epoch
      $Serial =~ s/\n//;
my $Refresh = '10800';          # 3 hours
my $Retry = '3600';             # 1 hour
my $Expire = '2419200';         # 4 weeks
my $Minimum = '86400';          # 1 day
my $NS1 = 'ns1.example.com';
my $NS2 = 'ns2.example.com';
my $NS3 = 'ns3.example.com';
```

2.3.7.7   DNS_Storage

This is the directory where replaced DNS files are stored. You do not need a trailing slash.

Defaults:

```
my $DNS_Storage = '../Storage/DNS';
```

2.3.7.8   DNS_Owner_ID

For changing the ownership of the DNS file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

2.3.7.9   DNS_Group_ID

For changing the group ownership of the DNS file after it's created, we need to specify a group owner. It is recommended to set this to be group owned by BIND, so the group name is usually 'bind'.

Defaults:

```
my $Group = 'bind';
```

2.3.7.10   Reverse_Proxy_Location

This is the path that the system writes the temporary reverse proxy files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the reverse proxy config. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Location = '../Storage/tmp/ReverseProxy';
```

2.3.7.11   Proxy_Redirect_Location

This is the path that the system writes the temporary proxy redirect files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the proxy redirect config. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Location = '../Storage/tmp/ ReverseProxy';
```

2.3.7.12   Reverse_Proxy_Storage

This is the directory where replaced reverse proxy files are stored. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Storage = '../Storage/ReverseProxy';
```

2.3.7.13   Proxy_Redirect_Storage

This is the directory where replaced proxy redirect files are stored. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Storage = '../Storage/ReverseProxy';
```

2.3.7.14 DShell_Job_Log_Location

This is the directory where job logs are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_Job_Log_Location = '../Storage/D-Shell/Job-Log';
```

2.3.7.15 DShell_tmp_Location

This is the directory where D-Shell temporary files are are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_tmp_Location = '../Storage/D-Shell/tmp';
```

2.3.7.16 DShell_WaitFor_Timeout

The default time that a *WAITFOR statement will wait before bailing out. Can be overridden manually by issuing *WAITFORnn, where nn is the timeout in seconds. nn can be any number.

Defaults:

```
my $DShell_WaitFor_Timeout = 1800; # 30 minutes
```

DB_Management and DB_Sudoers sections which contain details of which variables to set to define an alternate database host, database port, database schema names and database usernames. If this is not a new installation, you are highly advised to follow the Upgrading process instead, as the following instructions assume a new installation and **steps below may overwrite existing configurations**.

If this is a new instance on MySQL, you should first set the root password. This should be a highly complex password. As root, run the following on the TM Server:

```
mysqladmin -u root password '<YOUR NEW PASSWORD>'
```

Once set, you should be able import the full database schema. The following step assumes that you completed the Extracting the Package step and that the DSMS installation files remain in */tmp/sudoers*. Re-run the extraction steps if the files no longer exist. As root on the TM Server, run the following (when prompted, enter your MySQL root password):

```
cd /tmp/sudoers/Configs/SQL/
mysql -u root -p < Full_Schema.sql
```

The newly imported schema does not yet have privileges assigned to it. If you wish to assign custom usernames and passwords, you should skip this step and grant privileges manually. To assign default privileges run the following as root on the TM Server (when prompted, enter your MySQL root password):

```
cd /tmp/sudoers/Configs/SQL/
mysql -u root -p < Default_Users.sql
```

To ensure that the permissions have been applied correctly, and to ensure that the users didn't already exist and have extra privileges, you should run the following as root on the TM Server:

```
echo 'show grants for Management@localhost; show grants for Sudoers@localhost;' | mysql -u root -p
```

```
Enter password: <YOUR MYSQL ROOT PASSWORD>
Grants for Management@localhost
GRANT USAGE ON *.* TO 'Management'@'localhost' IDENTIFIED BY PASSWORD
'*99F49D92B5730C682FA7B5B21689F26188A71D3E'
GRANT SELECT, INSERT, UPDATE, DELETE ON `Management`.`credentials` TO
'Management'@'localhost'
GRANT SELECT, UPDATE ON `Management`.`lock` TO 'Management'@'localhost'
GRANT SELECT, INSERT ON `Management`.`audit_log` TO 'Management'@'localhost'
GRANT SELECT, INSERT ON `Management`.`access_log` TO 'Management'@'localhost'
GRANT SELECT, INSERT, UPDATE, DELETE ON `Management`.`distribution` TO
'Management'@'localhost'
Grants for Sudoers@localhost
GRANT USAGE ON *.* TO 'Sudoers'@'localhost' IDENTIFIED BY PASSWORD
'*EF151896427DA84765D2D5557BB39E26F2582200'
GRANT SELECT, INSERT, UPDATE, DELETE ON `Sudoers`.* TO 'Sudoers'@'localhost'
```

Your returned result should **exactly match the above highlighted privileges**. If it does not, consult a Database Administrator - continuing with incorrect privileges could mean that TM does not function correctly, or, worse, TM could be insecure.

The below table summarises the recently set privileges on a per user, per table basis:

Default Database Name	Table Name	Default User	Required Permissions
Management	access_log	Management	SELECT, INSERT
Management	audit_log	Management	SELECT, INSERT
Management	credentials	Management	SELECT, INSERT, UPDATE, DELETE
Management	distribution	Management	SELECT, INSERT, UPDATE, DELETE
Management	lock	Management	SELECT, UPDATE
Sudoers	command_groups	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	commands	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	host_groups	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	hosts	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	lnk_command_groups_to_commands	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	lnk_host_groups_to_hosts	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	lnk_rules_to_command_groups	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	lnk_rules_to_commands	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	lnk_rules_to_host_groups	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	lnk_rules_to_hosts	Sudoers	SELECT, INSERT, UPDATE, DELETE

Sudoers	lnk_rules_to_user_groups	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	lnk_rules_to_users	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	lnk_user_groups_to_user	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	notes	Sudoers	SELECT, INSERT
Sudoers	rules	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	user_groups	Sudoers	SELECT, INSERT, UPDATE, DELETE
Sudoers	users	Sudoers	SELECT, INSERT, UPDATE, DELETE

2.3.8 IPTables Configuration

IPTables may require modification to allow HTTPS connections from clients. The following allows clients to connect to the TM Server on the default HTTPS port, 443. If the TM Server uses a non-standard HTTPS port, you must modify this value to match your configuration. Restart IPTables for the new configuration to take effect. For clients running IPTables, run the following as root on the TM Server:

```
sed -i /etc/sysconfig/iptables -e '/INPUT -j REJECT/i \
\
# Distributed Sudoers Management System HTTPS Exception \
-A INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT \
'
/etc/init.d/iptables restart
```

For clients running Firewalld, run the following as root on the TM Server:

```
firewall-cmd --permanent --zone=public --add-rich-rule="rule family="ipv4" port protocol="tcp" port="443" accept"
systemctl restart firewalld
```

2.3.9 SELinux Configuration

As part of the user account reset mechanism, TM handles password resets via email. The password reset mechanism is covered in more detail in the Account Lockout Reset Process section. To allow TM to send these emails, the following SELinux Boolean needs to be set. If you are not using SELinux, skip this step. As root on the TM Server, run the following:

```
setsebool -P httpd_can_sendmail on
```

For TM Servers running RHEL/CentOS 7 or above, also run the following:

```
setsebool -P httpd_enable_cgi on
semanage fcontext -a -t httpd_sys_script_exec_t "/var/www/html(/.*)?"
restorecon -vRF /var/www/html
```

2.3.10 Common Parameter Configuration

The following describes what each configurable variable does in each section of the Common Configuration file, *common.pl*. You should edit the file with a text editor, such as vi or nano. Only edit the values that need to be explicitly changed and read each section carefully, as some parts of the file should be not edited.

2.3.10.1 Maintenance_Mode

This is a system toggle to turn on or off Maintenance Mode. When Maintenance Mode is on, users are prevented from making system changes, or accessing the system. This is a useful mode to set before upgrading, or during installation. Maintenance Mode is 'On' by default, but should be set to 'Off' after installation or upgrade is finished.

Installation or Upgrade Default:

```
my $Maintenance_Mode = 'On';
```

Running System Default:

```
my $Maintenance_Mode = 'Off';
```

2.3.10.2 System_Name

This is the system's name, used for system identification during login, written to the sudoers file to identify which system owns the sudoers file, is used in password reset emails to identify the source, and other general uses.

Defaults:

```
my $System_Name = 'Distributed Sudoers Management System';
```

2.3.10.3 System_Short_Name

This is the system's shortened name, which is used in short descriptions. It can be the same as the full name in System_Name if you want, but it might get busy on some screens if your system name is long. It's encouraged to keep this short (less than 10 characters).

Defaults:

```
my $System_Short_Name = 'DSMS';
```

2.3.10.4 Verbose

Turns on verbose mode without having to directly trigger this on individual system components. Verbose data is outputted to both command line invoked components and to logs. It is generally useful for light debugging or following a system's activities in more detail.

Defaults:

```
my $Verbose = 0;
```

2.3.10.5 Very_Verbose

Turns on very verbose mode without having to directly trigger this on individual system components. The output from this is extensive and as a result is not suitable for reading on the fly. This should only be used for debugging problems with The Machine itself as the data is not likely to be useful for debugging user input data. By turning on very verbose, verbose is also automatically turned on.

Defaults:

```
my $Very_Verbose = 0;
```

2.3.10.6 Paper_Trail

Turns on the paper trail. This will log all captured parameters including both encrypted and plain text password. It should only be used for debugging and only with fake user details or on test systems. In

case this is accidentally turned on, paper trail also requires that verbose is explicitly turned on to work correctly.

Defaults:

```
my $Paper_Trail = 0;
```

2.3.10.7 System_Log_File

This is the location on the system where the main log file goes. This is where TM will log its internal log outputs.

Defaults:

```
my $System_Log_File = '../Storage/System/System_Log';
```

2.3.10.8 Header

The parameter should not normally be changed. It's a handler for TM to determine where the header file is in relation to its current working location.

Defaults:

```
if (-f 'header.cgi') {$Header = 'header.cgi';} else {$Header = '../header.cgi';}
```

2.3.10.9 Footer

The parameter should not normally be changed. It's a handler for TM to determine where the footer file is in relation to its current working location.

Defaults:

```
if (-f 'footer.cgi') {$Footer = 'footer.cgi';} else {$Footer = '../footer.cgi';}
```

2.3.10.10 DNS_Server

By setting a DNS server here, it will override the operating system's DNS server when doing lookups.

Defaults:

```
my $DNS_Server = '192.168.1.1';
```

2.3.10.11 LDAP_Login

These are the connection parameters for LDAP / Active Directory. If you disable this, the system will use TM's internal authentication mechanism. By default TM will try to connect to the LDAP /Active Directory server using TLS (StartTLS) first, before reverting to plain text if TLS is not available. Even with LDAP authentication turned on, granular TM permissions are still set within TM itself.

Defaults:

```
my $LDAP_Enabled = 'On'; # Set this to 'Off' to disable LDAP/AD authentication

my $LDAP_Server = '192.168.2.1';
my $LDAP_Port = 389;
my $Timeout = 5;
my $LDAP_User_Name_Prefix = 'DOMAIN\\';
my $LDAP_Filter = '(&(objectClass=inetOrgPerson)(memberOf=cn=bla,ou=bla,dc=bla,dc=local))';
my $LDAP_Search_Base = 'ou=User Accounts,dc=bla,dc=local';
```

2.3.10.12 Recovery_Email_Address

This is the email address that TM will appear to send emails from during password recoveries. It may be a legitimate address (such as the system administrator's address) or it could be a blocking address, such as noreply@nwk1.com.

Defaults:

```
my $Recovery_Email_Address = 'noreply@nwk1.com';
```

2.3.10.13 Sudoers_Location

2.3.10.14 This is not necessarily the location of the /etc/sudoers file. This is the path that the system writes the temporary sudoers file to. It could be /etc/sudoers, but you ought to consider the rights that Apache will need to overwrite that file, and the implications of giving Apache those rights. If you want to automate it end to end, you should consider writing a temporary sudoers file, then using a separate root cron job to overwrite /etc/sudoers, which is the recommended procedure, instead of directly writing to it. Of course, if you do not intend on using TM to manage /etc/sudoers on the local machine, then this should NOT be /etc/sudoers. For sudoers locations on Remote Servers, see DB_IP_Allocation

This is your IP Allocation database's connection information.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'IP_Allocation';
my $User = 'IP_Allocation';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.15 DB_Icinga

This is your Icinga database's connection information. This is where your Icinga data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'Icinga';
my $User = 'Icinga';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.16 DB_DNS

This is your DNS database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'DNS';
my $User = 'DNS';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.17 DB_Reverse_Proxy

This is your Reverse Proxy database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'Reverse_Proxy';
my $User = 'Reverse_Proxy';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.18 DB_DShell

This is your D-Shell database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'DShell';
my $User = 'DShell';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.19 Reverse_Proxy_Defaults

These are the default reverse proxy values for entries without custom parameters.

Defaults:

```
my $Transfer_Log = '/var/log/apache/access.log';
my $Error_Log = '/var/log/apache/error.log';
my $SSL_Certificate_File = '/etc/pki/tls/certs/default.cert';
my $SSL_Certificate_Key_File = '/etc/pki/tls/private/default.key';
my $SSL_CA_Certificate_File = '/etc/pki/tls/ca-bundle.pem';
```

2.3.10.20 Redirect_Defaults

These are the default proxy redirect values for entries without custom parameters.

Defaults:

```
my $Transfer_Log = '/var/log/apache/access.log';
my $Error_Log = '/var/log/apache/error.log';
```

DSMS_Distribution_Defaults, or set individual remote sudoers locations through the web panel.

Defaults:

```
my $Sudoers_Location = '/var/www/html/sudoers';
```

2.3.10.21 Sudoers_Storage

This is the directory where replaced sudoers files are stored. You do not need a trailing slash.

Defaults:

```
my $Sudoers_Storage = '/var/www/html/sudoers-storage';
```

2.3.10.22 Sudoers_Owner_ID

For changing the ownership of the sudoers file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

2.3.10.23 Sudoers_Group_ID

For changing the group ownership of the sudoers file after it's created by the DSMS build process, we need to specify a group. It is recommended to run the DSMS build process as a root cron job (as defined in Cron Configuration), but Apache will need to reach this file to display its live contents on the web panel. If you modified Apache based on the Apache Configuration section, the group ownership should usually be 'apache'. However, on some systems, Apache Server doesn't run as the 'apache' user, such as when it runs as 'httpd', so you must specify the appropriate group ownership for TM to read the sudoers file.

Defaults:

```
my $Group = 'apache';
```

2.3.10.24 DNS_Zone_Master_File

This is the zone master file use for defining zones. You should include the full path.

Defaults:

```
my $DNS_Zone_Master_File = '/etc/bind/named.conf.local';
```

2.3.10.25 DNS_Internal_Location

This is the path that the system writes the temporary Internal DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_Internal_Location = '/etc/bind/master-internal';
```

2.3.10.26 DNS_External_Location

This is the path that the system writes the temporary External DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_External_Location = '/etc/bind/master-external';
```

2.3.10.27 DNS_Internal_SOA

This is the SOA data that's written at the top of the Internal DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email = 'postmaster@example.com';
        my $TTL = '86400';              # 1 day
        my $Serial = `date +%s`;        # Epoch
             $Serial =~ s/\n//;
```

```
        my $Refresh = '10800';          # 3 hours
        my $Retry = '3600';             # 1 hour
        my $Expire = '2419200';         # 4 weeks
        my $Minimum = '86400';          # 1 day
        my $NS1 = 'ns1.example.com';
        my $NS2 = 'ns2.example.com';
        my $NS3 = 'ns3.example.com';
```

2.3.10.28 DNS_External_SOA

This is the SOA data that's written at the top of the External DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email = 'postmaster@example.com';
        my $TTL = '86400';              # 1 day
        my $Serial = `date +%s`;        # Epoch
            $Serial =~ s/\n//;
        my $Refresh = '10800';          # 3 hours
        my $Retry = '3600';             # 1 hour
        my $Expire = '2419200';         # 4 weeks
        my $Minimum = '86400';          # 1 day
        my $NS1 = 'ns1.example.com';
        my $NS2 = 'ns2.example.com';
        my $NS3 = 'ns3.example.com';
```

2.3.10.29 DNS_Storage

This is the directory where replaced DNS files are stored. You do not need a trailing slash.

Defaults:

```
 my $DNS_Storage = '../Storage/DNS';
```

2.3.10.30 DNS_Owner_ID

For changing the ownership of the DNS file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
 my $Owner = 'root';
```

2.3.10.31 DNS_Group_ID

For changing the group ownership of the DNS file after it's created, we need to specify a group owner. It is recommended to set this to be group owned by BIND, so the group name is usually 'bind'.

Defaults:

```
 my $Group = 'bind';
```

2.3.10.32 Reverse_Proxy_Location

This is the path that the system writes the temporary reverse proxy files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the reverse proxy config. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Location = '../Storage/tmp/ReverseProxy';
```

2.3.10.33 Proxy_Redirect_Location

This is the path that the system writes the temporary proxy redirect files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the proxy redirect config. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Location = '../Storage/tmp/ ReverseProxy';
```

2.3.10.34 Reverse_Proxy_Storage

This is the directory where replaced reverse proxy files are stored. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Storage = '../Storage/ReverseProxy';
```

2.3.10.35 Proxy_Redirect_Storage

This is the directory where replaced proxy redirect files are stored. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Storage = '../Storage/ReverseProxy';
```

2.3.10.36 DShell_Job_Log_Location

This is the directory where job logs are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_Job_Log_Location = '../Storage/D-Shell/Job-Log';
```

2.3.10.37 DShell_tmp_Location

This is the directory where D-Shell temporary files are are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_tmp_Location = '../Storage/D-Shell/tmp';
```

2.3.10.38 DShell_WaitFor_Timeout

The default time that a *WAITFOR statement will wait before bailing out. Can be overridden manually by issuing *WAITFORnn, where nn is the timeout in seconds. nn can be any number.

Defaults:

```
my $DShell_WaitFor_Timeout = 1800; # 30 minutes
```

2.3.10.39 DB_Management

This is your Management database's connection information. This could be the same database as the database in the DB_Sudoers because the two schemas have different table names to facilitate a

combination. However, the Management data (System Accounts, Access Log, Audit Log, etc) contain sensitive information that normal users should not be allowed access to. This access control should also be applicable to Database Administrators, which is why this data is stored in a separate database by default to simplify access control.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'Management';
my $User = 'Management';
my $Password = ' Password removed from this document, please set a secure and unique one.';
```

2.3.10.40 DB_Sudoers

This is your Sudoers database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'Sudoers';
my $User = 'Sudoers';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.41 DB_IP_Allocation

This is your IP Allocation database's connection information.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'IP_Allocation';
my $User = 'IP_Allocation';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.42 DB_Icinga

This is your Icinga database's connection information. This is where your Icinga data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'Icinga';
my $User = 'Icinga';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.43 DB_DNS

This is your DNS database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'DNS';
my $User = 'DNS';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.44 DB_Reverse_Proxy

This is your Reverse Proxy database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'Reverse_Proxy';
my $User = 'Reverse_Proxy';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.45 DB_DShell

This is your D-Shell database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'DShell';
my $User = 'DShell';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

2.3.10.46 Reverse_Proxy_Defaults

These are the default reverse proxy values for entries without custom parameters.

Defaults:

```
my $Transfer_Log = '/var/log/apache/access.log';
my $Error_Log = '/var/log/apache/error.log';
my $SSL_Certificate_File = '/etc/pki/tls/certs/default.cert';
my $SSL_Certificate_Key_File = '/etc/pki/tls/private/default.key';
my $SSL_CA_Certificate_File = '/etc/pki/tls/ca-bundle.pem';
```

2.3.10.47 Redirect_Defaults

These are the default proxy redirect values for entries without custom parameters.

Defaults:

```
my $Transfer_Log = '/var/log/apache/access.log';
my $Error_Log = '/var/log/apache/error.log';
```

2.3.10.48 DSMS_Distribution_Defaults

These are the default sudoers distribution settings for new hosts. Keep in mind that any active host is automatically tried for sudoers pushes with their distribution settings. Unless you are confident that all new hosts will have the same settings, you might want to set fail-safe defaults here and manually override each host individually on the Distribution Status page.

A good fail-safe strategy would be to set *$Key_Path* to be */dev/null* so that login to the Remote Server becomes impossible. Alternatively, another good method would be to set *$Remote_Sudoers* to */dev/null*, so that you could accurately test remote login, but not affect the existing sudoers file at */etc/sudoers*. Note that if you setup SFTP to use chroot, the sudoers path will be relative to the chroot

jail, so it's likely to be *upload/sudoers*. This is also dependent on your Cron Configuration on the Remote Server.

Defaults:

```
my $Distribution_SFTP_Port = '22'; # Default SFTP port
my $Distribution_User = 'transport'; # Default SFTP user
my $Key_Path = '/root/.ssh/id_rsa'; # Default private key path
my $Timeout = '15'; # Default stalled connection Timeout in seconds
my $Remote_Sudoers = 'upload/sudoers'; # Default sudoers file location on remote systems
```

2.3.10.49 Password_Complexity_Check

Here you can set minimum requirements for password complexity and control whether password complexity is enforced. Take particular care with the special character section if you choose to define a single quote (') as a special character as this may prematurely close the value definition. To define a single quote, you must use the character escape, backslash (\), which should result in the single quote special character definition like this (\'), less the brackets. The space character is pre-defined by default at the end of the string and does not need escaping.

Defaults:

```
my $Enforce_Complexity_Requirements = 'Yes'; # Set to Yes to enforce complexity requir[…]
my $Minimum_Length = 8; # Minimuim password length
my $Minimum_Upper_Case_Characters = 2; # Minimum upper case characters required ([…]
my $Minimum_Lower_Case_Characters = 2; # Minimum lower case characters required ([…]
my $Minimum_Digits = 2; # Minimum digits required (can be 0)
my $Minimum_Special_Characters = 2; # Minimum special characters (can be 0)
my $Special_Characters = '!@#$%^&*()[]{}-_+=/\,.<>" '; # Define special characters (you can […]
```

2.3.10.50 CGI

This contains the CGI Session parameters. The session files are stored in the specified *$Session_Directory*. The *$Session_Expiry* is the time that clients must be inactive before they are logged off automatically. It's unwise to change either of these values whilst the system is in use. Doing so could cause user sessions to expire prematurely and any changes they were working on will probably be lost. Refer to the tables below. The table on the left shows the alias letter definitions; the table on the right gives some example expiry values:

Alias	Definition
s	Seconds
m	Minutes
h	Hours
d	Days
w	Weeks
M	Months
y	Years

Example	Definition
$Session_Expiry = '+1h';	Set the expiry to +1h to expire the session after 1 hour. This is the default.
$Session_Expiry = '+15m';	Set the expiry to +15m to expire the session after 15 minutes.
$Session_Expiry = '+30s';	Set the expiry to +30s to expire the session after 30 seconds.
$Session_Expiry = '+5s';	Set the expiry to +5s if you're Chuck Norris.

Defaults:

```
my $Session_Directory = '/tmp/Sudoers-CGI-Sessions';
my $Session_Expiry = '+1h';
```

2.3.10.51 md5sum

Manually set the path to `md5sum` here, or just leave this as default and the system will try to determine its location through `which md5sum --skip-alias`.

Defaults:

```
my $md5sum = `which md5sum --skip-alias`;
```

2.3.10.52 cut

Manually set the path to `cut` here, or just leave this as default and the system will try to determine its location through `which cut --skip-alias`.

Defaults:

```
my $cut = `which cut --skip-alias`;
```

2.3.10.53 visudo

Manually set the path to `visudo` here, or just leave this as default and the system will try to determine its location through `which visudo --skip-alias`

Defaults:

```
my $visudo = `which visudo --skip-alias`;
```

2.3.10.54 cp

Manually set the path to `cp` here, or just leave this as default and the system will try to determine its location through `which cp --skip-alias`.

Defaults:

```
my $cp = `which cp --skip-alias`;
```

2.3.10.55 ls

Manually set the path to `ls` here, or just leave this as default and the system will try to determine its location through `which ls --skip-alias`.

Defaults:

```
my $ls = `which ls --skip-alias`;
```

2.3.10.56 sudo_grep

Manually set the path to `grep` here, or just leave this as default and the system will try to determine its location through `which grep --skip-alias`.

Why sudo_grep and not grep? - grep is a function of perl, but the function doesn't give the output we need, so we use the TM Server's grep application instead. If the subroutine is named 'grep', and is called through grep(), perl's grep is called, and not the TM Server's grep.

Defaults:

```
my $grep = `which grep --skip-alias`;
```

2.3.10.57 head

Manually set the path to `head` here, or just leave this as default and the system will try to determine its location through `which head --skip-alias`.

Defaults:

```
my $head = `which head --skip-alias`;
```

2.3.10.58 nmap

Manually set the path to `nmap` here, or just leave this as default and the system will try to determine its location through `which nmap --skip-alias`.

Defaults:

```
my $nmap = `which nmap --skip-alias`;
```

2.3.10.59 ps

Manually set the path to `ps` here, or just leave this as default and the system will try to determine its location through `which ps --skip-alias`.

Defaults:

```
my $ps = `which ps --skip-alias`;
```

2.3.10.60 wc

Manually set the path to `wc` here, or just leave this as default and the system will try to determine its location through `which wc --skip-alias`.

Defaults:

```
my $wc = `which wc --skip-alias`;
```

2.3.10.61 git

Manually set the path to `git` here, or just leave this as default and the system will try to determine its location through `which git --skip-alias`.

Defaults:

```
my $git = `which git --skip-alias`;
```

2.3.10.62 Version

This is where TM discovers its version number, which assists with both manual and automated Upgrading, among other things. You should not modify this value.

Defaults:

```
my $Version = '2.0.0';
```

2.3.10.63 Server_Hostname

This is where TM discovers its hostname. This is useful when determining which host you're connected to in High Availability (HA) configurations, which TM fully supports, and is covered in the High Availability and Load Balancing section. You should not modify this value.

Defaults:

```
my $Hostname = `hostname`;
```

2.3.10.64 Random_Alpha_Numeric_Password

This is where TM generates password resets using alpha numeric characters. There are no user changeable values in this section, and any modifications could detriment the security of the system. Do **not** modify this section.

2.3.10.65 System_Logger

This is the system logging function and is used in combination with Paper_Trail. There are no user configurable parameters here.

2.3.10.66 enc

This is the subroutine where encoding for various system components occurs. There are no configurable parameters here.

2.3.10.67 dec

This is the subroutine where decoding for various system components occurs. There are no configurable parameters here.

2.3.10.68 Salt

This is where TM generates password salts using alpha numeric and special characters. There are no user changeable values in this section, and any modifications could detriment the security of the system. Do **not** modify this section.

2.3.11 Cron Configuration

TM has two main components that require a cron job.

The first job is to build the sudoers file at regular intervals, syntax check the new sudoers file, backup any sudoers files that have been changed, and audit the new and old hashes of those files for manual inspection or automated sudoers restoration in case of a future syntax fault.

The second job is to securely distribute the new sudoers files to the Remote Servers and make a report on the success or failure of the transfer back to the main DSMS System.

Both jobs should be run as root to protect the integrity of the Remote Servers. The build process should be run before the distribution process and the distribution process should not start until the build process is complete. The cleanest way of achieving this is to only call the distribution process if the build process completes successfully.

The default is to build the sudoers file every ten minutes, but this is configurable. The /var/www/html is the HTTP web files root path that you defined in Moving the HTTP Files and may differ from the example below.

You should not edit any other part of the line below unless you fully understand the consequences.

As root, on the TM Server, run the following:

```
echo '
```

```
# Distributed Sudoers Management System Build and Distribution Processes
*/10 * * * * root cd /var/www/html/ > /dev/null 2>&1 && ./sudoers-build.pl > /dev/null 2>&1 &&
./distribution.pl > /dev/null 2>&1
' >> /etc/crontab
```

2.3.12   Maintenance Mode

By default, Maintenance Mode is set to 'On' during system installation. This must be turned off before users can connect to the system and the build and distribution system begin working correctly. As root on the TM Server, run the following:

```
cd /var/www/html
sed -i -r "s/Maintenance_Mode = 'On'/Maintenance_Mode = 'Off'/" common.pl
```

2.3.13   Install Complete

You should have now finished TM installation. Try to navigate to the TM Server's IP address with your browser over HTTPS. If you cannot reach the TM Server, contact your system or network administrator. Continue reading the document to setup Remote Servers, or skip to First Time Use if you intend on setting up the remote servers at a later time.

2.4   Remote Server Automated Installation

The current server and desktop kickstart processes already contain an automated DSMS setup and attachment process. For installing to existing systems manually, you can either follow the Remote Server Manual Installation or Rapid Remote Server Deployment processes.

2.5   Remote Server Manual Installation

2.5.1   IPTables Configuration

IPTables may require modification to allow SSH connections from the TM Server. The following allows the TM Server to connect to the Remote Server on the default SSH port, 22. If the Remote Server uses a non-standard SSH port, you must modify this value to match your configuration. You also must modify the TM Server IP in the below text - failure to do so will render all IPTables rules unusable. For clients using IPTables, as root on the Remote Server, run the following:

```
sed -i /etc/sysconfig/iptables -e '/INPUT -j REJECT/i \
\
# Distributed Sudoers Management System SSH Exception \
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -s <TM Server IP> -j ACCEPT \
'
/etc/init.d/iptables restart
```

For clients using Firewalld, as root on the Remote Server, run the following:

```
firewall-cmd --permanent --zone=public --add-rich-rule="rule family="ipv4" source address="<TM Server IP>" port protocol="tcp" port="22" accept"
systemctl restart firewalld
```

2.5.2   Adding a Transport User

For additional security, you are advised to use a dedicated user for transporting the sudoers file to the Remote Server. This is not a requirement, but is highly recommended. By using a dedicated user, you can limit their access to only SFTP connections, thereby removing any risk of the user gaining a shell on the Remote Server. In the below examples, the transport user is named 'transport'. Initially, we set the user's shell as /bin/sh to facilitate the transfer of the TM Server's public key. This will later be

changed to refuse shell logins for the user. On each Remote Server, run the following commands as root, and give the user a secure password:

```
useradd -m -d /home/transport -s /bin/sh transport
passwd transport
Changing password for user transport.
New password: <transport account password>
Retype new password: <transport account password>
passwd: all authentication tokens updated successfully.
```

2.5.3 Adding the TM Server's Public Key

The TM Server uses key authentication to authenticate itself on Remote Servers. The TM Server supports having a different public/private key pair for each server, which can be configured in the Distribution Status management page. Given that the private keys on the TM Server will authenticate us against other servers, it is wise to only allow root to read the DSMS keys. Assuming that you haven't yet created any keys, run the following on the TM Server as root:

```
ssh-keygen -t rsa
```

Accept all defaults unless you want to store the key in a different location. You should get an output similar to this:

```
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
83:8f:5f:c7:4d:ba:70:83:0b:17:b8:c9:dd:b0:56:e3 root@dev-box
```

Once the key set has been created, you need to add the public key of the TM Server to the authorized_keys file on the Remote Server, which, if you called the user transport, should be located at /home/transport/.ssh/authorized_keys. For security, you should make a record of each Remote Server's fingerprint before connecting to it (and therefore adding it as a known host). This step may not be possible for all systems, but where you have local access it is recommended. On the local console on each Remote Server as root, run:

```
ssh-keygen -l -f /etc/ssh/ssh_host_rsa_key.pub
2048 94:3f:80:d5:48:45:92:b1:ea:aa:fe:c9:6e:bb:60:be /etc/ssh/ssh_host_rsa_key.pub (RSA)
```

Whilst */etc/ssh/ssh_host_rsa_key.pub* is noted above as a command variable, in reality this is highly likely to be the location of the host's key on Linux systems. Take note of the returned line and in particular the host's highlighted fingerprint, which, in this case, is *94:3f:80:d5:48:45:92:b1:ea:aa:fe:c9:6e:bb:60:be.*

The following uses the command *ssh-copy-id* which is part of the *openssh-clients* package. If you do not have this package installed on your system, either install it, or copy the TM Server's public key manually to the *authorized_keys* file of the transport user. From the TM Server, run the following as root and modify <Remote Server IP> to read the Remote Server's IP that you wish to connect to whilst paying particular attention that each host's fingerprint matches the one you discovered in the previous step (if it does not match, do not accept the key and contact your System Administrator immediately):

```
ssh-copy-id transport@<Remote Server IP>
The authenticity of host '1.2.3.4 (1.2.3.4)' can't be established.
RSA key fingerprint is 94:3f:80:d5:48:45:92:b1:ea:aa:fe:c9:6e:bb:60:be.
Are you sure you want to continue connecting (yes/no)? yes
```

> Warning: Permanently added '1.2.3.4' (RSA) to the list of known hosts.
> transport@1.2.3.4's password: <transport account password>
> Now try logging into the machine, with "ssh 'transport@1.2.3.4'", and check in:
>
>   .ssh/authorized_keys
>
> to make sure we haven't added extra keys that you weren't expecting.

As advised by the output, you should try to connect to the remote host with, in this case, *ssh 'transport@1.2.3.4'.* You should find yourself logged into the Remote Server as the transport user without being prompted for a password - if you are not, contact your System Administrator as there may be a fault or a non-standard configuration applied to the Remote Server that may need addressing.

If the SSH connection test was successful, run the following on the Remote Server as root:

```
usermod -s /sbin/nologin transport
```

2.5.4   sshd_config Configuration

Depending on your required setup, the /etc/ssh/sshd_config file on each Remote Server may need some modification to force the transport user into a chroot jail, and enforce only SFTP connections from the transport user. Below is an example configuration to be appended to the sshd_config file which should force the transport user to use only the SFTP subsystem. Occasionally, SFTP is already defined as a Subsystem in sshd_config, but it uses options that are not sufficient for our use. On each Remote Server, run the following as root:

```
sed -e '/Subsystem[\s.*|\t.*]sftp/ s/^#*/#/' -i /etc/ssh/sshd_config
```

And check that any existing *Subsystem sftp* lines are now commented (defined by a prefixed hash):

```
grep 'Subsystem.*sftp' /etc/ssh/sshd_config
#Subsystem      sftp      /usr/libexec/openssh/sftp-server
```

On each Remote Server, run the following as root to add the required configuration to sshd_config, and then restart the SSH service to make the changes take effect:

```
echo '
Subsystem sftp internal-sftp

Match User transport, Address <TM Server IP>
   ChrootDirectory /home/transport
   AllowTCPForwarding no
   X11Forwarding no
   ForceCommand internal-sftp' >> /etc/ssh/sshd_config
/etc/init.d/sshd restart
```

2.5.5   Transport Directory Configuration

Because the transport user will be in a chroot jail, it must have very specific permissions set on its home directory, and its home directory must be owned by root. In addition, we must also create a directory that the transport user can upload into as it will no longer have permission to write into its home directory - we'll call this directory 'upload' in these examples, but you can use a different name.

It is advised to create a writable directory as opposed to creating a writeable file in the root of the transport user's home directory, as the DSMS sudoers distribution process first writes a temporary file during the transfer then renames that file once the transfer is complete to ensure file integrity, and to ensure that the cron process that overwrites */etc/sudoers* does not overwrite */etc/sudoers* with a partially written file. In addition, the temporary file has a relatively unpredictable name, and is

therefore difficult to pre-create or allow for with defined exceptions. Therefore, we use a writable directory. As root on each Remote Server, run the following:

```
mkdir /home/transport/upload/
chown -R root:transport /home/transport/
chmod -R 750 /home/transport/
chmod 440 /home/transport/.ssh/authorized_keys
chown transport:root /home/transport/upload/
chmod 320 /home/transport/upload/
```

2.5.6   SELinux Configuration

If your system uses SELinux, we need to explicitly allow the transport user to chroot into their home directory by setting the ssh_chroot_rw_homedirs Boolean from 'off' to 'on':

```
setsebool -P ssh_chroot_rw_homedirs on
```

You should now be able to SFTP with keys to the Remote Server as the transport user and upload files into the 'upload' directory.

2.5.7   Cron Configuration

If you have followed the above recommendations for using a chroot jail, you will need the Remote Server to move the transferred sudoers file from */home/transport/upload/sudoers* to */etc/sudoers*.

The default is to copy the sudoers file every seven minutes, as this is the least likely time frequency to conflict with a current sudoers transfer, because running the move process every minute that's divisible by seven will not run at the same time as any minute divisible by ten in any single hour.

This of course assumes that all transfers will complete in under a minute (based on the 20th (divisible by 10) and 21st (divisible by 7) minute in the hour being one minute apart). However, some transfers may take longer than one minute, so to safeguard against any incomplete sudoers files overwriting the sudoers file at */etc/sudoers* due to a partial transfer, we perform one final *visudo* check against the transferred sudoers file. If the *visudo* check fails, it will return an exit code of 1, and the */etc/sudoers* file will not be overwritten; however if the *visudo* check passes, it will return an exit code of 0, and the */etc/sudoers* file will be updated with the newest sudoers file from the DSMS System.

You can modify the time that the cron job runs, and you may need to adjust the transport user to match the user you defined in Adding a Transport User, and the transport directory you defined in Transport Directory Configuration, but in this example we'll use the defaults which are 'transport' and 'upload' respectively.

You should not edit any other part of the line below unless you fully understand the consequences.

As root, on the Remote Server, run the following:

```
echo '
# Distributed Sudoers Management System File Relocation
*/7 * * * * root cd /home/transport/upload/ > /dev/null 2>&1 && `which visudo --skip-alias` -c -f
sudoers > /dev/null 2>&1 && `which cp --skip-alias` sudoers /etc/sudoers > /dev/null 2>&1  &&
`which curl --skip-alias` -k "https://<DSMS IP>/checkin.pl?Host_Name=`hostname`" > /dev/null
2>&1
' >> /etc/crontab
```

2.5.8   Testing the Connection

You should now be able to reach the Remote Server from the TM Server using SFTP. From the TM Server, substitute the below user and IP address with the Remote Server values and run the following as root; you should get a SFTP prompt on the Remote Server:

```
sftp transport@1.2.3.4
Connecting to 1.2.3.4...
sftp>
```

2.6   Rapid Remote Server Deployment

The below script is a rapid deployment script, which could be included in a Remote Server's kickstart file, or quickly applied to many servers in seconds manually through a traditional remote shell, or with a remote command execution system, such as Ansible. Before deploying this script, you should change the highlighted values to match your system. Once deployed, the Remote System should be fully configured to receive the CDSF.

```
### IPTables Additions (For RHEL 6 and below, SLES 11 and below, uncomment) ###
#sed -i /etc/sysconfig/iptables -e '/INPUT -j REJECT/i \
#\
## Distributed Sudoers Management System SSH Exception \
#-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -s <DSMS_IP> -j ACCEPT \
#'
#/etc/init.d/iptables restart
### / IPTables Additions ###


### Firewalld command. (For RHEL 7 and above, SLES 12 and above) ###
#firewall-cmd --permanent --zone=public --add-rich-rule="rule family="ipv4" source
#address="<DSMS_IP>" port protocol="tcp" port="22" accept"
### / Firewalld command ###


### Transport User and Public Key Addition ###
useradd -m -d /home/transport -s /sbin/nologin transport
mkdir -p /home/transport/.ssh
echo 'ssh-rsa <ROOT'S_PUBLIC_KEY>' >> /home/transport/.ssh/authorized_keys
### / Transport User and Public Key Addition ###


### Configuration of sshd_config chroot ###
sed -e '/Subsystem[\s.*|\t.*]sftp/ s/^#*/#/' -i /etc/ssh/sshd_config
echo '
Subsystem sftp internal-sftp
Match User transport, Address <DSMS_IP>
   ChrootDirectory /home/transport
   AllowTCPForwarding no
   X11Forwarding no
   ForceCommand internal-sftp' >> /etc/ssh/sshd_config
/etc/init.d/sshd restart
### / Configuration of sshd_config chroot ###


### Transport User chroot Configuration ###
mkdir /home/transport/upload/
chown -R root:transport /home/transport/
chmod -R 750 /home/transport/
chmod 440 /home/transport/.ssh/authorized_keys
chown transport:root /home/transport/upload/
chmod 320 /home/transport/upload/
### / Transport User chroot Configuration ###


### SELinux chroot Boolean Configuration ###
setsebool -P ssh_chroot_rw_homedirs on
### / SELinux chroot Boolean Configuration ###
```

```
### Crontab Configuration ###
echo '
# Distributed Sudoers Management System File Relocation
*/7 * * * * root cd /home/transport/upload/ > /dev/null 2>&1 && `which visudo --skip-alias` -c -f
sudoers > /dev/null 2>&1 && `which cp --skip-alias` sudoers /etc/sudoers > /dev/null 2>&1 &&
`which curl --skip-alias` -k "https:// <DSMS IP>/checkin.pl?Host_Name=`hostname`" > /dev/null
2>&1
' >> /etc/crontab
### / Crontab Configuration ###

IFACE=`ip route | grep ^default | head -1 | cut -f 5 -d ' '`;
IPADDR=`ip addr  | grep $IFACE | grep "inet " | sed -e 's/\/.*$//' -e 's/^.*inet //'`;
LOWER_FQDN=`echo $FQDN | tr '[:upper:]' '[:lower:]'`;

### Register with DSMS ###
curl -k "https:// <DSMS_IP>/register.pl?Host_Name_Add=$LOWER_FQDN&IP_Add=$IPADDR"
### / Register with DSMS ###
```

3   Upgrading

3.1   Determine your Current Version

The current version can be determined from two main places. You can determine the version from the main system Web Panel, or from the command line. You must make a note of the version number as part of the upgrade process.

3.1.1   From the Web Panel

The version is displayed in the top left corner of the web panel on each page, along with the system's hostname, your username, and the Logout link. In the 3.1.1a example, the system version is 1.5.0.

DSMS version 1.5.0 on dev-box | Welcome Ben Schofield [Logout]

3.1.1a - Web Panel Version

3.1.2   From the Command Line

If you cannot access the Web Panel (or are in Maintenance Mode) but do have at least read access to the DSMS files, run the following command in the root of the DSMS HTTP directory. This is usually /var/www/html/, but could differ depending on your configuration:

```
cd /var/www/html
grep '$Version\s=' common.pl | sed -r "s/.*'(.*)'.*/\1/"
```

After running that command, you should receive an output like:

```
1.5.0
```

In this case, 1.5.0 is your version number.

3.2   Understanding the Existing Environment

Before you begin the upgrade process, it is wise to gather information about the existing environment first to fully understand the upgrade requirements. See the table below, which lists the system defaults. If you installed the system exactly as described in TM Server Manual Installation and Remote Server Manual Installation, or used the automated installation method of each and accepted the proposed defaults, your installation should exactly match the details in the below table.

Item	Default Installation Value/Location
Installation Path **(for files *.cgi, common.pl)**	/var/www/html
Build Script Location	/var/www/html/sudoers-build.pl
Distribution Script Location	/var/www/html/distribution.pl
Sudoers Build Location	/var/www/html/sudoers
Cron Sudoers Build and Distribution Frequency **(TM Server)**	10 minutes
Cron Sudoers Relocation Frequency **(Remote Server(s))**	7 minutes

Management Database Name	Management
Database Management Username	Management
Management User Password	<MANAGEMENT MYSQL PASSWORD>
Sudoers Database Name	Sudoers
Database Sudoers Username	Sudoers
Sudoers User Password	< SUDOERS MYSQL PASSWORD>
SSH Transport Username	transport
Identity Key Location	/root/.ssh/id_rsa
Sudoers Transfer Timeout	15 seconds
Remote Sudoers Path (chroot dependent)	upload/sudoers

If your system uses any none default settings, you must make a note of the differences and modify the upgrade process where applicable to ensure that your system remains functional after the upgrade.

3.3 Maintenance Mode On

You are advised to prevent users from making changes to the system until the upgrade process is complete. There is a built-in Maintenance Mode, which prevents users from accessing the system, and also prevents the Build and Distribution systems from making any modifications to any files or databases.

To enable Maintenance Mode, as root, run the following on the TM Server:

```
cd /var/www/html
sed -i -r "s/Maintenance_Mode = 'Off'/Maintenance_Mode = 'On'/" common.pl
```

3.4 Automated Upgrade

An automated upgrade process is currently being built and tested. Please use the Manual Upgrade section.

3.5 Manual Upgrade

3.5.1 Backup Configuration Files

You should backup the configuration file before making any system changes, so that, in addition to the record of differences you made in Understanding the Existing Environment, you have a copy of the previous configuration.

```
cd /var/www/html
cp common.pl common.pl-`date +%Y-%m-%d`
chmod 000 common.pl-`date +%Y-%m-%d`
```

3.5.2   Backup the Databases

You should backup both databases before making any system changes, to ensure that you have a copy of all data should anything go wrong. The following will require you to know the MySQL root password. You may need to modify the dump queries below to match your database names. As root, on the TM Server, run the following:

```
mysqldump -u root -p Management > /root/Management-Backup-`date +%Y-%m-%d`.sql
Enter password: <YOUR MYSQL ROOT PASSWORD>
mysqldump -u root -p Sudoers > /root/Sudoers-Backup-`date +%Y-%m-%d`.sql
Enter password: <YOUR MYSQL ROOT PASSWORD>
```

3.5.3   Extracting the Package

Upload the latest DSMS package to the /tmp directory on the TM Server through any means you wish. As root on the TM Server, run the following:

```
cd /tmp
tar -xzf sudoers-release-1.10.0.tar.gz
```

3.5.4   File Integrity Checking

Before upgrading TM files, we must check them for integrity to ensure they're not corrupt or incomplete. As root on the TM Server, run the following:

```
cd /tmp/sudoers
md5sum -c checksums
```

You should inspect every returned line for any failures. A successful checksum returns an OK message for each matching file which looks like this:

```
./HTTP/index.cgi: OK
```

A checksum failure looks like this:

```
./HTTP/index.cgi: FAILED
```

If you identify any failed checksums, source a new sudoers-release-1.10.0.tar.gz file and begin the upgrade process again from Backup Configuration Files.

3.5.5   Changelog Review

Carefully review the changelog for any new requirements, such as new perl modules, new permission requirements, new Linux packages or new minimum versions of any of the aforementioned. Install the missing requirements before continuing. Alternatively, determine your missing requirements from the System Requirements section of this documentation.

You must also note the version number differences between your current version and the latest version detailed in the changelog. This is important, as you are required to upgrade the database in sequence.

As root on the TM Server, run the following to view the changelog:

```
cat /tmp/sudoers/changelog
```

You may wish to pipe the output to 'more' or 'less' if you have a small scroll back limit on your terminal.

3.5.6   Moving the HTTP Files

After all the files checksum correctly, we need to move the files into the HTTP root directory. The HTTP root directory is usually *var/www/html*, but if you have a different HTTP root directory, you should use the directory that's appropriate to your system. As root on the TM Server, run the following:

```
mv /tmp/sudoers/HTTP/ /var/www/html
```

3.5.7   Determining Configuration Differences

Your upgraded system may come with new defaults configured in common.pl, or entirely new options. To quickly discover the differences between your upgraded system and your existing system, compare the new and old common.pl files:

```
cd /var/www/html
diff common.pl common.pl-`date +%Y-%m-%d`
```

Manually update the new common.pl file with your configuration settings, if they differ.

3.5.8   File Permissions

To improve security, we need to assign as restrictive permissions to the files as possible. You may need to substitute some of the values with your custom changes if required. If you do not run SELinux, omit the last line. As root on the TM Server, run the following:

```
cd /var/www/html
touch sudoers
mkdir sudoers-storage/
chown root:apache *.cgi
chmod 650 *.cgi
chown root:apache common.pl register.pl checkin.pl
chmod 650 common.pl register.pl checkin.pl
chown root:root sudoers-build.pl distribution.pl
chmod 100 sudoers-build.pl distribution.pl
chown root:apache environmental-defaults sudoers
chmod 640 environmental-defaults sudoers
chown -R root:apache format.css favicon.ico resources/
chmod -R 440 format.css favicon.ico resources/
chown root:apache resources/ resources/imgs/ resources/imgs/buttons/
chmod 550 resources/ resources/imgs/ resources/imgs/buttons/
chown -R root:root sudoers-storage/
chmod -R 700 sudoers-storage/
semanage fcontext -a -t httpd_sys_script_exec_t "/var/www/html(/.*)?"
restorecon -vRF /var/www/html
```

3.5.9   Database Upgrade

New releases often come with database changes. Instead of applying the full schema to your existing database and potentially losing data, the DSMS package comes with pre-built SQL files specifically for upgrading. You are highly advised to **upgrade to each version in sequence**; for instance, if upgrading from version 1.4 to 1.6, you must first run the SQL upgrade from 1.4 to 1.5, and then run the SQL upgrade from 1.5 to 1.6 - **not doing so could cause inconsistent data and your DSMS System may cease to function**.

The following will require you to know the MySQL root password. The following also assumes that you use the default database names, Management and Sudoers - if you do not, first change your database names within the Upgrade.sql file for each version change. Inspect the folders, and edit the version number in the command below if necessary. For each version step, as root on the TM Server, run the following:

```
cd /tmp/sudoers/Configs/Upgrade/1.5.0\ to\ 1.6.0/
mysql -u root -p < Upgrade.sql
```

3.6   Maintenance Mode Off

After completing the upgrade process, Maintenance Mode must be turned off before users can use the system again.

To disable Maintenance Mode, as root, run the following on the TM Server:

```
cd /var/www/html
sed -i -r "s/Maintenance_Mode = 'On'/Maintenance_Mode = 'Off'/" common.pl
```

4   First Time Use

4.1   Logging In

Using your browser, navigate to your system's URL. If you are not already logged in, the system will redirect you to the login.cgi page. You should always use a secure connection to TM as recommended as part of the installation process.

For first time login, you will need to use the account and password that you specified as part of the installation process. If you used the TM Server Manual Installation process, one account is configured for you already with the following credentials:

- User Name: admin
- Password: 123

Note: Immediately change this password after logging in before entering any other data.

To change your account's password after logging in, consult the Changing Your Password section.

It is also recommended that you create a different administration account with a less obvious username, and then delete the 'admin' account. Account administration is explained in Account Management.

TM contains a mechanism that detects hijacked browser cookies by way of recording the client's IP address along with the session data on the server. If you find that your session is prematurely expiring before the session expiry time defined in the CGI section, you may want to consider that your IP address is changing by way of a short DHCP lease and TM is forcefully terminating your session.

4.2   Logging Out

Located at the top left of every page, to the right of your user name, is a 'Logout' link. You can see an example of this in 3.3.1a. When clicking the 'Logout' link, your session is immediately destroyed and you are forced back to the Login page.

4.3   Determining the System Version

The System's version can be determined from two main places. You can determine the version from the main system Web Panel, or from the command line.

4.3.1   From the Web Panel

The version is displayed in the top left corner of the web panel on each page, along with the system's hostname, your username, and the Logout link. In the 4.3.1a example, the system version is 1.5.0, the hostname is dev-box, and the username is Ben Schofield. If you find a fault with

4.3.1a - Web Panel Version

the system or experience difficulties, you should provide this system information along with a description of the fault.

4.3.2   From the Command Line

If you cannot access the Web Panel but do have at least read access to the DSMS files, run the following command in the root of the DSMS HTTP directory. This is usually */var/www/html/*, but could differ depending on your configuration:

```
grep '$Version\s=' common.pl | sed -r "s/.*'(.*)'.*/\1/"
```

After running that command, you should receive an output like:

```
1.5.0
```

In this case, 1.5.0 is your version number.

4.4   Navigation

Navigation is done by using the six menus at the top of every page. Each menu is titled to describe the page's general function that it links to. Some menu buttons contain sub-menus when hovered with your mouse cursor. For instance, the 'Groups' dropdown menu includes Host Groups, User Groups and Command Groups within. For an example of this, see 4.4a.

Home	Groups	Hosts	Sudo Users	Commands	Rules
	Host Groups				
	User Groups				
	Command Groups				
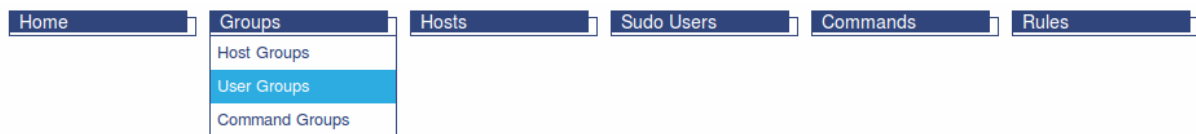
4.4a - Navigation

The Home menu also contains general system tools, such as the Change Password utility and the system Changelog, as well as system management functions in a separate sub-menu titled Management. Management functions are covered in the System Management Use section.

4.5   Common Interface Indications

Throughout the system, there are a set of commonly used icons and colours in use to indicate different functions and conditions. This is to assist you in quickly recognising certain functions or conditions across different pages.

4.5.1   Common Icons
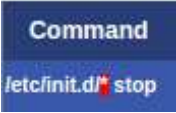
TM contains a set of common icons to assist you in quickly identifying what functions can be applied to an item. See the following table for a breakdown of each icon's meaning.

Icon	Name	Description
	Edit	This icon, which is also a button, denotes that the current item can be edited. To edit the item's parameters, click this button.
	Delete	This icon, which is also a button, denotes that the current item can be deleted. To delete this item, click this button. This button also serves as a 'Close Window' function for popup windows, and appears in the extreme top right of the window.
	Notes	This icon, which is also a button, denotes that the current item can have notes assigned to it. This is a dynamic icon, as it also includes the number of current notes associated with the item. In this case, this item has 3 notes that can be read.
	Approve	This icon, which is also a button, denotes that the current item can be approved. To approve this item, click this button.
	Cannot Approve	This icon denotes that the current item can be approved, but not by the user that you're currently logged in as. To Approve this item, you must login as an Approver that is not also the current user.

	View	This icon, which is also a button, denotes that the current item can be viewed in more detail, but you will be taken to a different page to see that additional detail. To view the item in more detail, click this button.
	On	This is a general use icon to represent something that is on or an allowed action. For instance, if you see this icon next to a permission on the Rule configuration page, it means that the permission statement is true, and you can perform the action described.
	Off	This is a general use icon to represent something that is off or a disallowed action. For instance, if you see this icon next to a permission on the Rule configuration page, it means that the permission statement is false, and you cannot perform the action described.
	Links	This icon, which is also a button, denotes that the current item may be linked to other components in the system. For instance, a Host Group may be linked to some Hosts and some Rules; to view which components that this item is associated with, click this button.

4.5.2   Common Colour Indications

TM contains a set of common colours to assist you in quickly identifying an item's status. See the following table for a breakdown of each icon's meaning.

Example	Colour	Description
Active / Yes	**Green**	Items coloured green generally represent active or approved items. In this example, the green highlights that the current item is Active.
Approved / No	**Red**	Items coloured red generally represent inactive or unapproved items. Red items also represent item errors. In this example, the red highlights that the current item is not Approved.
Run As Tags / root PASSWD, EXEC	**Orange**	Items coloured orange generally represent warnings, such as for item modifications in the Audit Log, or highlight unsafe values, such as running a command as root. In this example, the orange cell of Run As is warning that the current item is to be Run As root, which is potentially unsafe. In the second example, the orange of the EXEC tag highlights that the EXEC option is potentially unsafe.
Expires / 2014-10-05	**Grey**	Items coloured grey generally represent expired items. In this example, the grey cell of Expires signifies that this item expired on 2014-10-05.
Rule Name / ApacheControl	**Yellow Highlight**	Items highlighted yellow are highlighted as a result of matching a Search string. This is applicable to both Global Search and Local Search types. In this example, the search string was 'apache' which matched the Rule name 'ApacheControl' and highlighted the matching section of the string.
Command / /etc/init.d/* stop	**Red Highlight**	Items highlighted red are often done so because they could represent a dangerous component. In this example, the asterisk of the Command is highlighted because it could be used dangerously, or as otherwise intended.
Last Modified Last Approved 2014-10-13 17:04:34 2014-10-13 17:06:26	**Dual Colour**	To reduce the required horizontal space for some lines, similar items may be combined and represented by different colours to maintain distinguishability. In this example, Last Modified and Last Approved are combined because they are similar datasets. To maintain distinguishability between the two values, the Last Approved values are coloured differently from the Last Modified values.

4.6 Changing Your Password

If you wish to change your DSMS System password, you can do so through the Change Password page. The Change Password page is available at /password-change.cgi, or though the menus by navigation to Home -> Change Password. To change your password, you will be required to know your old password. Passwords are salted with a random per user string, and then hashed, before being stored. Depending on your configuration, your password may be subject to complexity requirements.

4.7 Search

The system has two search functions; a Global Search which searches the system for a user provided string, or a Local Search, which searches for a specific type of item in the current page.

4.7.1 Global Search

The Global Search function can be found at the top of every page to the top right of the navigation menu, an example of which is shown in 4.7.1a. You can search the system for Hosts, Users, Commands, Groups and Rules that match your input string. To search, type your search query in

4.7.1a - Global Search Input

the input box and hit Enter. Search results are displayed in a separate overlay window and are categorised based on their function. For example, when searching for 'http' using the Global Search, both Command Groups and Commands are returned as matching the string, as displayed under the

'Category' column. The search function is case insensitive for convenience.

Items matching the search string are highlighted in place by a yellow background. Notice that for the three Commands returned in the Search Results example in 4.7.1b,

4.7.1b - Global Search Results

the Global Search tool matched both the 'HTTP' part of the Command's name, and the 'http' part of the attached command in brackets. The Global Search tool also includes Status and View columns.

The Status column describes the current status of the applicable item. Notice that results 1, 3 and 4 are Active, while result 2 is Inactive. Also notice that result 4, while Active, has also Expired, which means that it will not be included in the final sudoers file until the Expiry condition is either reset or removed. Incidently, this column also highlights the general colour rules discussed in Common Interface Indications to determine the condition of the current item.

The View column contains a View button for each item, which, when pressed, will take you to the item's page to view more details about it, where you can also perform further actions, such as editing or deleting the item.

4.7.2 Local Search

The Local Search function can usually be found in the grey settings panel above tables. The local search is for filtering results of the table that you're currently viewing, which is why it's often labelled 'Filter' to differentiate it more easily from the Global Search tool. To search, type your search query in the input box and hit Enter. Items matching the search string are highlighted in place by a yellow background. In the example in 4.7.2a, the search string was 'restart' which returned two commands containing the word restart - this search function is also case insensitive. Results that do not match the input string

4.7.2a - Local Search

are stripped from the table. To clear the filter, either visit the page again, or submit a blank search in the input box.

5 System Management

5.1 Account Management

5.1.1 Viewing System Accounts

To view System Accounts, navigate to the /account-management.cgi page directly, or through the menu at Home -> Management -> Account Management.

System Accounts are used to add Hosts, Users and Commands, as well as configure Groups and build Rules. Each DSMS user should have their own System Account, and System Accounts should not be shared between users - doing so undermines the DSMS System's auditing and access control systems. There is no limit to the number of System Accounts on the DSMS System.

User Name	Email Address	Last Login	Last Active	Admin	Approver	Requires Approval	Lockout	Last Modified	Modified By	Edit	Delete
Ben Schofield	benoch@datacom.co.nz	2014-10-14 12:48:37	2014-10-14 12:58:50	Yes	Yes	Yes	No	2014-10-14 12:58:50	admin		

5.1.1a - System Account Details

The System Accounts present on TM are displayed on the Account Management page, along with some information relating to each account, such as the Account's email address, the last time the Account was logged into, the last time there was activity on the Account, the last time the Account was modified and who was the last user to edit the Account. Note that the last Account modification time is also updated when a user changes their own password. An example account is shown in 5.1.1a.

All Account changes are logged to the Audit Log.

5.1.2 System Account Permissions

Each System Account has a permission set that dictates what they are able to do within the DSMS System. See the table below for a list of permissions and a description of the rights associated with each permission.

Permission	Options	Default	Description
Administrator Privileges	**Yes/No/Read-only**	**No**	If this option is set to Yes, the Account becomes a System Administrator. Accounts with System Administrator Privileges can create, edit and delete System Accounts, as well as edit the privileges of any account including their own. Administrator accounts should be treated as 'root' accounts and should not be used for general system use. An Administrator account also has permission to view the Distribution Status page, the System Status page, the Access Log page and Audit Log page. A Read-only Administrator can view Administrative pages except the Account Management page, but cannot make any changes.
Can Approve Rule Changes	**Yes/No**	**No**	If this option is set to Yes, the Account becomes an Approver. Approvers can approve new and edited Rules created or edited by other users.

Requires Rule Change Approval	**Yes/No**	**Yes**	If this option is set to No, this Account is exempt from the restrictions preventing users from approving their own Rules.
Locked Out	**Yes/No**	**No**	If this option is set to Yes, this Account becomes locked out and cannot be logged into. This is a useful option for temporarily disabling a System Account if a user will not be using TM for an extended period. This option will switch automatically to Yes if five incorrect passwords are tried against the Account.

5.1.3 Creating System Accounts

To create a System Account, click the 'Add Account' button in the top centre of the screen (see 5.1.3a). A new window titled 'Add New Account' will appear with form elements requesting new account details. When filling in the requested details, you must be accurate - User Name and Password combinations are used for login, and Email is used for account reset.
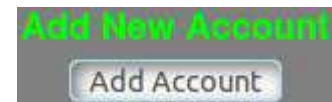
5.1.3a - Add New Account

The username 'System' is a reserved System Account and cannot be created.

To create a System Account, you must be a System Administrator.

5.1.4 Editing System Accounts

System Accounts can be edited by using either the edit dropdown menu shown in 5.1.4a, or by using the Edit icon next to the Account's details on the Account Management page.

5.1.4a - Edit an Account

To edit a System Account, you must be a System Administrator.

5.1.5 Deleting System Accounts

System Accounts can be deleted by using the Delete icon next to the Account's details on the Account Management page. Deleted Accounts cannot be recovered.

To delete a System Account, you must be a System Administrator.

5.2   Distribution Status

5.2.1   Viewing Distribution Status

To view the Distribution Status of the sudoers file, navigate to the /distribution-status.cgi page directly, or through the menu at Home -> Management -> Distribution Status.

Host ID	Host (IP)	User	Key Path	Timeout	Remote Sudoers Path	Status Message	Status	Status Received	Last Modified	Modified By	Edit
1	wlgprdapp013 (127.0.0.1)	transport	/root/.ssh/id_rsa	15	upload/sudoers	OK: upload/sudoers written successfully to wlgprdapp013 (127.0.0.1). Sudoers MD5: 184f5dc5d940b393ea0a015301b40a9e	OK	2014-10-13 23:15:02	2014-10-09 16:53:31	Ben Schofield	
2	wlgprdapp02 (127.0.0.2)	transport	/root/.ssh/id_rsa	15	upload/sudoers	Connection Failed: Connection to remote server stalled Hints: 1) Check that the key fingerprint is stored in known_hosts 2) Check for a route to the remote host 3) Check that your 15 second Timeout value is high enough	Error	2014-10-13 23:14:58	2014-10-09 16:35:15	Ben Schofield	

5.2.1a - Distribution Status

The Distribution Status page is used as a live indicator about the distribution status of the sudoers file to each host. Every new host is automatically included in the sudoers distribution system, so no extra user input is required. 5.2.1a shows two hosts to which the sudoers file is distributed. See the table below for an explanation of each column.

5.2.2   Default Distribution Parameters

Column Name	Description
Host ID	The Host ID is synonymous with the Host ID in the Sudoers Hosts table and is a unique identifier for each host.
Host Name (IP)	This column shows the Remote Server's host name, followed by the Remote Server's IP address in brackets. When the Distribution System attempts to reach a remote host, it uses the IP address to connect to avoid any reliance on DNS. The Host Name and IP address are synonymous with the host name and IP address in the Sudoers Hosts table.
User	This is the user that creates the SFTP connection to the Remote Server. It should be a user account on the Remote Server, and ideally be restricted to the SFTP sub system and in a chroot jail for security.
Key Path	This is the path to the identity key file used to authenticate the SFTP user on the Remote Server.
Timeout	This is the amount of time in seconds that the SFTP connection process waits for a response from the Remote Server before aborting the transfer.
Remote Sudoers Path	This is the path to where the sudoers file is put onto the Remote Server, before the Remote Server's cron task collects the file and moves it to /etc/sudoers. This path should be defined as a full path on the Remote Server, unless the Remote User SFTPs into a chroot jail, in which case it should be a path relative to the chroot jail directory with the preceding slash removed (i.e. *upload/sudoers* instead of */home/transport/upload/sudoers*).
Status Message	The Status Message is a description of the result of the last transfer attempt. Successful transfers are noted with an 'OK' while transfers that fail to connect display 'Connection Failed' and transfers that connect successfully but fail to write the sudoers file display 'Push Failed'. Each status message is followed by further details of the transfer, such as hints about what might've gone wrong if the transfer failed.
Status	This is a quick status message, which either displays as 'OK' if the sudoers file has been successfully written to the Remote Server or 'Error' if the transfer failed. Any hosts reporting an error should be diagnosed with the Diagnosing Failed Transfers section.
Last Modified	This details when the host's individual transfer parameters were changed. This is not a record of host changes - those are noted in the Sudoers Hosts table - but does record when this host's distribution system parameters were last modified.
Modified By	This details who changed this host's individual transfer parameters. This is not a record of host changes - those are noted in the Sudoers Hosts table - but does record who modified this host's distribution system parameters.

5.2.2.1   When new hosts are added to the DSMS System, they inherit default distribution parameters. These are defined in the DB_IP_Allocation

5.2.2a - Distribution Defaults

This is your IP Allocation database's connection information.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'IP_Allocation';
my $User = 'IP_Allocation';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

5.2.2.2   DB_Icinga

This is your Icinga database's connection information. This is where your Icinga data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'Icinga';
my $User = 'Icinga';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

5.2.2.3   DB_DNS

This is your DNS database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'DNS';
my $User = 'DNS';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

5.2.2.4   DB_Reverse_Proxy

This is your Reverse Proxy database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'Reverse_Proxy';
my $User = 'Reverse_Proxy';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

5.2.2.5   DB_DShell

This is your D-Shell database's connection information. This is where your sudoers data is stored.

Defaults:

```
my $Host = 'localhost';
my $Port = '3306';
my $DB = 'DShell';
my $User = 'DShell';
my $Password = 'Password removed from this document, please set a secure and unique one.';
```

5.2.2.6   Reverse_Proxy_Defaults

These are the default reverse proxy values for entries without custom parameters.

Defaults:

```
my $Transfer_Log = '/var/log/apache/access.log';
my $Error_Log = '/var/log/apache/error.log';
my $SSL_Certificate_File = '/etc/pki/tls/certs/default.cert';
my $SSL_Certificate_Key_File = '/etc/pki/tls/private/default.key';
my $SSL_CA_Certificate_File = '/etc/pki/tls/ca-bundle.pem';
```

5.2.2.7   Redirect_Defaults

These are the default proxy redirect values for entries without custom parameters.

Defaults:

```
my $Transfer_Log = '/var/log/apache/access.log';
my $Error_Log = '/var/log/apache/error.log';
```

DSMS_Distribution_Defaults section. The current distribution defaults assigned to new hosts are show at the top of the Distribution Status page; an example of the distribution defaults section is shown in 5.2.2a.

5.2.3   Assigning Individual Host Parameters

You can assign individual connection parameters to any host, which override the default distribution parameters. To assign individual parameters, either choose the host from the drop down menu under 'Edit Host Parameters' (shown in 5.2.3a) or click the Edit button next to the individual host.

5.2.3a - Edit Host Parameters

Once you've chosen a host, the Edit Host Parameters window will appear with details about the host's configuration at the top, and below that input boxes to set the host's distribution parameters. The host's current values are displayed in each relevant input box - replace these values with the new parameters and click the 'Edit Host Parameters' button to save the changes. You should notice that you host will have been updated with the new parameters in the Distribution Status page.

5.2.4   Diagnosing Failed Transfers

A transfer could fail for a number of reasons. Failed transfers are best diagnosed by a Linux System Administrator or Engineer. Please consult the table below to try to identify common problems and how to resolve them.

Error	Diagnosis and Resolution
Connection Failed: Connection to remote server stalled	This is the most common failure type for new Remote Servers. There are three main problems that cause this fault: 1) Remote Host's fingerprint is not stored in the known_hosts file. This can be resolved manually, by applying the remote server's key to the known_hosts file. The distribution process is usually run as root, so the correct known_hosts file is usually /root/.ssh/known_hosts. Alternatively, attempt to manually create an SSH connection to the Remote Server from the TM Server and accept the key. 2) There is no route to the Remote Server. This is usually a network or firewall issue. Firstly, check iptables

on the TM Server and the Remote Server for correct configuration and adjust as necessary - TM uses SSH/SFTP exclusively to transfer the sudoers file between hosts. Also, attempt to manually create an SSH connection to the Remote Server from the TM Server taking note of any errors. If the TM Server and Remote Server's configurations are correct but the connection still fails, consult the network team for network diagnosis.

3) **The connection timed out before it could be completed.**
Often, this is due to a low SSH timeout set for the host, as some Remote Servers under load take too long to respond. The most straightforward test is to increase the timeout to a high value, such as 300 seconds. If this does not resolve the issue, the Remote System may be genuinely unreachable, and you should consult the network team for network diagnosis.

Connection Failed: Connection to remote server is broken	This error generally suggests that the Remote Server is reachable, which suggests a configuration problem as the cause of this error. There are four main problems that cause this fault: 1) The transport user has an incorrect username. Check that the SFTP user TM is trying to SFTP to the Remote Server with exists on the Remote Server. 2) The IP address is incorrectly formatted. An incorrectly formatted IP address will cause an immediate failure in the connection attempt. Check that your IP address recorded for the Remote Server is correct. 3) Key identity file not found. Check that the identity file exists, the path to the key file is correctly recorded for the host and that the key file is correctly formatted. Attempt a manual connection to the Remote Server by using the key path. 4) Insufficient permissions to read key identity file. Check that the ownership and permissions on the identity file are sufficient for the distribution process to read. The distribution process is most often run as root, however this is not always the case, so suitable file permissions are required for the system to function correctly.
Push Failed: Permission denied	This error is usually due to insufficient write permissions on the Remote Server for the Remote Sudoers file path. Check that the transport user can write to the Remote Sudoers file location.
Push Failed: Couldn't open remote file 'upload/sudoers(123).tmp': No such file	This error message could vary slightly depending on your Remote Sudoers Path setting. As part of the transfer process, the distribution system first writes a temporary file to the Remote Server which is then moved to the final Remote Sudoers Path when it has finished transferring. This is to prevent the Remote Server from collecting a partially transferred sudoers file and overwriting */etc/sudoers* with it. The error produced is often as a result of an incorrect Remote Sudoers Path location. Check that the Remote Sudoers Path specified is correct. This should be the full system path (e.g. */home/transport/upload/sudoers*) except in cases where the Remote Server uses a chroot for the transport user, in which case the path

should be relative to the chroot jail and not contain a prefixed slash (e.g. *upload/sudoers*).

5.3   System Status

5.3.1   Viewing the System Status

To view the DSMS System's Status, navigate to the /system-status.cgi page directly, or through the menu at Home -> Management -> System Status.

The System Status is live snapshot of the DSMS System's current condition. It is a convenience place to view the system's configuration exactly as defined in the *common.pl* file. In addition, it also contains details on the live Build and Distribution status, including whether either process is currently running, when the start and finish times were of the last run process of each, and a calculation that displays how long each process took to complete.

The System Status page can help diagnose system problems beyond the static contents of the *common.pl* file alone, as it displays the dynamic values by the live running process that would not otherwise be possible to see. For example, the System Status page displays the self-determined path to various system applications, and displays live outputs of randomly calculated password strings and salt values to demonstrate that these systems are functioning correctly.

5.4   Access Log

5.4.1   Viewing the Access Log

To view the DSMS System's Access Log, navigate to the /access-log.cgi page directly, or through the menu at Home -> Management -> Access Log.

The Access Log is a log of all DSMS System activity. The Access Log is reverse time sorted, so the most recent actions are listed at the top. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this log entry. It serves no informational value, other than as a reference point when highlighting an entry in the log.
IP	This is the IP of the client system that made the request.
Hostname	This is the hostname of the client system that made the request. This field may be blank if the client's IP address and hostname are not registered in a local DNS server. You will also need to have HostnameLookups set to On in httpd.conf (or you system's equivalent) for this field to be populated.
User Agent	This is the client's browser's version identifier. This information is useful when diagnosing abnormal system behaviour for one particular client, as it may highlight an incompatible browser.
Script	This is the file that the client executed on the system. Generally speaking, this is the page that the client was using when the log entry was made. This information is useful when discovering diagnostic information about system faults.
Referer	This is the file from which the user came. In other words, it was the file executed before the 'Script' file. It helps to build up a picture of how a user

	moved through the system, and may be useful information when discovering diagnostic information about system faults. Trivia: A misspelling of the word 'referrer' which made it into the HTTP standard accidentally because it wasn't recognised as an erroneous spelling by spell-checking software; or a person with a dictionary, evidently. Don't look at it for too long, it stings the eyes.
Query	The Query is the key value pair data that a client's browser sends to the server during some interactions. This information is useful when discovering diagnostic information about system faults.
Method	This details the method used to retrieve the current page. There are two possible values; GET and POST. GET is the most common and is used for all operations other than form posts, where POST is used.
HTTPS	The HTTPS flag highlights if a client is not using HTTPS to communicate with the server. All communication with TM should be done over HTTPS, where its usage is shown as 'On'. If this shows as 'Off' then the client's browser is not using HTTPS to communicate with the DSMS System. If the default DSMS System Installation procedure was used, all connections should be over HTTPS.
User Name	The User Name that the client was logged in as at the time of the operation.
Time	The time the operation occurred. This time is the TM Server's local time.

5.4.2   Filtering the Access Log

The Access Log can be filtered to show only the items that you'd like to view. There are two filter types, a User Name filter, and a general text Local Search.

The User Name filter shows all actions by a specific user and removes all other entries generated by other users.

The Local Search filter searches strings in the following fields:

- ID
- IP
- Hostname
- User Agent
- Script
- Referer
- Query
- Request Method
- User Name
- Time

Searches are case-insensitive and the two search methods can be combined together for more granular results. Matching patterns are highlighted according to the Common Colour Indications defaults.

5.5   Audit Log

5.5.1 Viewing the Audit Log

To view the DSMS System's Audit Log, navigate to the /audit-log.cgi page directly, or through the menu at Home -> Management -> Audit Log. Example entries in an Audit Log are displayed in 5.4.1a.

ID	Category	Method	Action	Time	User
344	Rules	Approve	Ben Schofield Approved Rule [Rule ID 3].	2014-10-13 16:56:12	Ben Schofield
343	Rules	Approve	Ben Schofield Approved ApacheControl [Rule ID 1].	2014-10-13 16:56:10	Ben Schofield
342	Hosts	Modify	Ben Schofield modified Host ID 8. The new entry is recorded as wlgprddb04 (127.0.1.4), set Active and does not expire.	2014-10-13 16:55:55	Ben Schofield
341	Rules	Revoke	Ben Schofield modified Host ID 8, which caused the revocation of 1 Rules to protect the integrity of remote systems.	2014-10-13 16:55:55	Ben Schofield
339	Distribution	Delete	Ben Schofield deleted wlgprdapp03 (127.0.0.31) [Host ID 3] from the sudoers distribution system.	2014-10-13 16:55:28	Ben Schofield

5.4.1a - Audit Log

The Audit Log is a detailed log of DSMS System changes. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this log entry. It serves no informational value, other than as a reference point when highlighting an entry in the log.
Category	The Category defines which item set the audit entry refers to. In the five examples in 5.4.1a, the first, second and fourth entries relate to the Rule set, the third relates to the Hosts set and the fifth relates to the Distribution System. Changes to these items are individually categorised to enable more efficient searching and sorting.
Method	The Method summaries the type of Action that was performed against the item. In the first two entries in 5.4.1a, the Rules were Approved. In the third entry, the Host was Modified. In the fourth entry, the Rule's Approval was revoked due to the modified Host in the third entry, and in the fifth entry a deleted Host was removed from the Distribution system.
Action	The Action is a detailed description of the change, in plain English. Actions relating to items always carry the item's ID, to make searching for an item's history as simple as searching for 'Item ID x', e.g. 'Rule ID 3'.
Time	The time the action was audited.
User	The user that performed the auditable action.

5.5.2 Filtering the Audit Log

The Audit Log can be filtered to show only the items that you'd like to view. There are four filter types, a User Name filter, a Category filter, a Method filter, and a general text Local Search.

The User Name filter shows all actions by a specific user and removes all other entries generated by other users.

The Category filter shows only specific items, such as Hosts or Rules.

The Method filter shows only actions that match a particular type, such as items that were added or removed.

The Local Search filter searches strings in the following fields:

- ID
- Category
- Method
- Action

- Time
- User

Searches are case-insensitive and the four search methods can be combined together for more granular results. Matching patterns are highlighted according to the Common Colour Indications defaults.

6 IP System

6 IP System

7   D-Shell System

7.1   Key Storage

There are two authentication systems available in D-Shell for remote hosts - password and public key. Either are provided, along with a username, when triggering a D-Shell Job. When authenticating with a password, a username is provided by the user. When authenticating with a key, the username is associated with the key itself and stored in the database.

The value of SSH keys is extremely high, so the storage of them is something that's taken very seriously with The Machine.

7.1.1   Submitting a Key

Keys are submitted through the 'My Account' page. There are five fields; four are mandatory:

Name	Required	Notes
Key Name	**Yes**	This is the name that you wish to identify your key with. It does not have to be unique and there are no special requirements. This is the name that will display when you select a key to authenticate against a host with.
Key Password	**Yes**	This is not the key's passphrase (if one exists for the key) but is the password that will be used to encrypt the key with while at rest. This should be a strong password. If you lose the password, you will not be able to use the key. It is not possible to store a key in plain text.
Key Username	**Yes**	This is the SSH username that is associated with this key for the Remote Host(s).
Has Passphrase	**No**	This informs The Machine that this key is already encrypted with a passphrase. The Machine will prompt you for this passphrase when you trigger a Job in addition to the password unlock string. Having a passphrase on a key is highly recommended for security. The existence of a passphrase on a key does not affect how it's encrypted by The Machine – it's always encrypted.
Private Key	**Yes**	The contents of the private key, in text. The key can be any OpenSSH compatible key (such as RSA, DSA, ECDSA, etc), provided that the SSH server on The Machine and the SSH server on the Remote Server support the key.

7.1.2   Deleting a Key

Deleting a key is straightforward. In your 'My Account' page, press the Delete button next to the key you wish to delete. A deleted key cannot be recovered.

7.1.3   Setting a Default Key

You can choose a key to be your default key when
triggering a Job and it will be pre-selected in the key selection dropdown. To set a key as default, in your 'My Account' page, press the Default button next to the key you wish to set as default. A defaulted key shows a green circle next to it while other keys have a grey circle.

7.1.4 Key Storage

When a key is submitted, the process of storage is defined in the table below.

Step	Technology	Explanation
1	**Password Padding**	When you submit a key with your key lock password, the key lock password is salted by random strings until the length of the password is 256 characters. This is then used as the key for the encryption processes.
2	**DES Encrypt Key**	The salted password is salted again by another random length for a second time and a hash is created of the password. The hash is then used to encrypt the key with DES encryption.
3	**AES (Rijndael) Encrypt DES File**	The salted password is salted again by a different random length string and a hash is created of the password. The DES file created in step 2 is encrypted again, this time with AES, with the newly hashed value. This double layered encryption is used to mitigate against the possibility that if either encryption technology is broken in the future, the key will still remain safe.

7.2 Processing System

7.2.1 Graphical Interface

7.2.1.1 Job Receiver

The Job Receiver has no graphical counterpart. You can interact with the Job Receiver by running a Command Set which will contact the Job Receiver on your behalf.

7.2.1.2 Job Processor

The Job Processor has no graphical counterpart. You can interact with the Job Processor by running a Command Set, running a queued Job, or otherwise controlling a Job. These components will contact the Job Processor on your behalf.

7.2.2 Command Line

7.2.2.1 Job Receiver

Jobs are always received by the system via command line. These can come from various sources, including the Command Sets graphical page and triggered from other scripts (through automation). The table below shows what parameters you can pass to the Job Receiver.

Parameter	Expected Value	Required	Explanation
-c --command-set	**Integer**	**Yes**	The database ID of the command set that you're queuing or triggering against a Remote Server.
-H --hosts	**(Multiple) Integer**	**Yes**	The space separated database IDs of the hosts you wish to run the Command Set against.

-u --username	String	No	The username that you wish to trigger this Job as. Note that if you pass a username to the Job Receiver, the Job will be queued *and* triggered, so processing with begin immediately.
-P --password	String	No	Passes the encoded/encrypted password. Used in combination with username. Note that, unlike the Job Processor, you cannot pass this as a plaintext string as there is no good reason for it.
-k --key	Integer	No	The key that you wish to trigger this Job with. Note that if you pass a key to the Job Receiver, the Job will be queued *and* triggered, so processing with begin immediately.
-f --failure	Integer	No	Use this parameter to specify the on-failure behaviour of commands. If a command has a non-zero exit status on a Remote Server, the entire Job will halt if you set on-failure to 1. If you set on-failure to 0, the Job will continue processing even if the remote system shows signs of failures. If you do not explicitly define this parameter, it will default to 1 – halt a job on command failure.
The parameters below this line are not shown in the help output of the Job Receiver to prevent you doing stuff without reading the consequences here first.			
-X	String	No	Use this parameter to pass details about who triggered the Job to determine ownership. It doesn't have to be a username, but it might help to easily identify your Jobs in the system.
-L --lock	String	No*	Passes the encoded/encrypted password. Used in combination with username. Note that, unlike the Job Processor, you cannot pass this as a plaintext string as there is no good reason for it. Required if a key parameter is passed.
-K --passphrase	String	No*	Passes the encoded/encrypted password. Used in combination with username. Note that, unlike the Job Processor, you cannot pass this as a plaintext string as there is no good reason for it. Required if a key parameter is passed and the key is encrypted with a passphrase.

7.2.2.2   Job Processor

The Job Processor is the component that connects to Remote Servers, executes commands, collates outputs, and determines actions to take based on system tags, outputs, errors, connectivity and other control flags.

It is the main logic component of the D-Shell system and is not directly driven by the user. Instead, it is driven by other components, such as the Job Receiver, Job Controller, and other Job Processors for multi-dependency Jobs. As it is not meant to be driven by the user, the parameters that you pass to it are not particularly straightforward. Nevertheless, it can be useful to directly run it for debugging or super verbosity.

The table below shows what parameters you can pass to the Job Processor.

Parameter	Expected Value	Required	Explanation
-j **--job**	**Integer**	**Yes**	The database ID of the job that you're triggering.
-p **--parent**	**Integer**	**Yes**	If this is a dependency of another process, the database ID of that process is passed here. Outputs for this execution will be associated with the parent job.
-c **--command-set**	**Integer**	**Yes**	The database ID of the command set that you're triggering against a Remote Server.
-H **--host**	**Integer**	**No***	The database ID of the host for which the job will be run against. If this process is a dependency of another, the host ID is required, otherwise it is resolved as part of the Job discovery process when gathering data from the database.
-d **--dependency-chain**	**Integer**	**No***	Used by the Job Processor to keep track of spawned sub-processes for logging purposes. The number increments starting from 0 for the parent, 1 for the first child, 2 for the second and so on. If no dependency chain value is passed it is assumed to be 0.
-u **--user**	**String**	**No***	The username used to login to the Remote Server. This is required if you aren't using key authentication. If you are using key authentication, the username is discovered during the discovery process on the database.
-k **--key**	**Integer**	**No**	The database ID of the key that you want to use to authenticate against the Remote Server.
-v **--verbose**	*null*	**No**	Turns on verbose output and logging. Does not take a value as passing it assumes verbose on; not passing it assumes verbose off.
-V **--very-verbose**	*null*	**No**	Turns on very verbose output and logging. Does not take a value as passing it assumes very verbose on; not passing it assumes very verbose off.
--override	*null*	**No**	When passed, allows you to re-run a Job that's already completed, is currently running, or is currently starting up. The system will otherwise reject jobs that are triggered twice without an override.
The parameters below this line are not shown in the help output of the Job Processor to prevent you doing stuff without reading the consequences here first.			
-P **--password**	**String**	**No**	This parameter is used to pass an encoded/encrypted password to the Job Processor. When combined with the -D / --nodec parameter, you can pass a password in plaintext. It's a really stupid idea to pass a plaintext password as a parameter, because it'll show up in any process list. If the Job Processor sees that you haven't passed a password (here) or passed a key, it will detect that you need to pass a password and create a prompt for it. The prompt will look like this: *Password for <your passed user>:*

Parameter	Expected Value	Required	Explanation
			If you don't know how to reliably pipe a plaintext string into a script then you can pass the password as a parameter. But you shouldn't.
-D --nodec	*null*	No	This turns off decode/decryption on passwords, lock strings and passphrases. Essentially, when The Machine passes credentials around to different components, they are first encoded/encrypted to mask themselves from the system - as well as is technically possible, at least. As it's fairly difficult to encrypt your password, passphrase, and lock string in your head, you can pass this parameter to tell the system not to try to decrypt the passwords that you pass it, therefore enabling you to pass them in plaintext. It's a pretty stupid thing to do on a shared system though, so only do this during debugging, and close the blinds to hide the shame.
-L --lock	String	No*	Use this parameter to pass your encoded/encrypted lock string to decrypt the key when it's collected from the database. When combined with the -D / --nodec parameter, you can pass a lock string in plaintext. This parameter is required if you pass a key ID.
-k --passphrase	String	No*	Use this parameter to pass your encoded/encrypted passphrase to decrypt the key when it's used to create an SSH connection. When combined with the -D / --nodec parameter, you can pass a passphrase in plaintext. This parameter is required if you pass a key ID and your key is encrypted with a passphrase.

7.3 Command Sets

7.3.1 Dependencies

7.3.2 History and Version Control

7.3.3 Triggering Jobs

7.4 Snapshot Control

7.4.1.1 *Graphical Interface*

The Snapshot system has no graphical counterpart. You can interact with Snapshots by running a Command Set that contains snapshot tags, as detailed in the Tagging System. Snapshots will then be controlled by the Job Processor on your behalf.

7.4.1.2 *Command Line*

Parameter	Expected Value	Required	Explanation
-t --threads	Integer	No	The snapshot processing system is multithreaded. You can specify the number of threads to use when for searching for a VM. More threads will find the VM faster as it will contact

			more hosts at once, but more threads will also increase the load on each VM's host.
-H **--hosts**	**String**	**Yes***	Not required if hosts are passed as IDs. Can also be combined with host IDs parameter. NEEDS TO HAVE CSV FIXED.
-i **--host-ids**	**Integer**	**Yes***	Not required if hosts are passed as hostnames. Can also be combined with hostnames parameter. NEEDS TO HAVE CSV FIXED.
-c **--count**	*null*	**No**	Count the total number of snapshots the hosts defined and show which of those were taken by The Machine. Counting snapshots makes no changes. If no other snapshot operations are defined (snapshot, remove, erase) count is used by default.
-s **--snapshot**	*null*	**No**	Attempts to take a memory snapshot of the hosts defined. If the memory snapshot fails due to the presence of independent disks, The Machine will instead try a regular snapshot.
-r **--remove**	*null*	**No**	Removes all snapshots taken by The Machine for the list of hosts defined.
-e **--erase**	*null*	**No**	Removes *all* snapshots for the hosts defined – not just those taken by The Machine.
-v **--verbose**	*null*	**No**	Turns on verbose output. Does not take a value as passing it assumes verbose on; not passing it assumes verbose off.
-V **--very-verbose**	*null*	**No**	Turns on very verbose output. Does not take a value as passing it assumes very verbose on; not passing it assumes very verbose off.

7.5   Jobs

7.5.1   Job Control

7.5.1.1   *Graphical Interface*

7.5.1.2   *Command Line*

7.5.2   Job Log

7.5.2.1   *Graphical Interface*

7.5.2.2   *Command Line*

../Storage/D-Shell/Job-Log/<Job ID>-<Dependency ID>[-Transactions]

Storage/D-Shell/Job-Log/151-0

Storage/D-Shell/Job-Log/151-0-Transactions

Storage/D-Shell/Job-Log/151-1

Storage/D-Shell/Job-Log/151-1-Transactions

7.5.3   Job Process and Exit Codes

D-Shell has several codes that are assigned to pending, running, complete and failed jobs. These behave like regular system exit codes for normal applications and will be returned by the Job Processor, and sent to the database to record the status of each job.

Job Code	Status	Command Notice	Notes
0	Job Complete	None, Job Complete.	Job has finished successfully.
1	Running	*Currently running command is displayed*	Displays the command currently being processed by the remote system.
2	Paused	None, Processing Paused.	Job has been manually paused. When resumed, the job will continue from the last executed command.
3	Killed	This job was killed manually.	A user killed the job.
4	Pending	None, Job Pending.	This job has been queued but is not set to run.
5	Error	Job Failed! Connection timeout, network or host resolution problems are the most likely causes. Try running it manually.	Usually caused by a bad hostname or IP, routing problems, or firewall restrictions.
6	Error	Job Failed! Bad credentials are the most likely cause. Try running it manually.	The credentials that you supplied do not seem to work on the host. Try them manually.
7	Error	Job Failed! Bailed out on unmatched WAITFOR. Check the log for what appeared.	The WAITFOR timeout was met before your WAITFOR tag was matched. Perhaps the timeout was too short, you mistyped the WAITFOR string or the string never appeared on the console – check the log to see what did appear.
8	Error	Execution Failed! User Name not caught.	Usually only occurs when submitting jobs directly into the Job Processor as the frontend sanity checks these inputs.
9	Error	Execution Failed! Password not caught.	Usually only occurs when submitting jobs directly into the Job Processor as the frontend sanity checks these inputs.

10	**Starting**	*Nothing is displayed, connection is being established*	Job is starting. SSH connection is being established with the remote host.
11	**Killed**	On failure set to kill. Failure condition met - job killed.	Job was killed by the system as an error occurred on the Remote Server.
12	**Error**	Lost the remote prompt. Command timeout, SSH connection died or the Job was terminated by the system.	Usually occurs if the host unexpectedly drops the connection, such as if it was rebooted or the SSH session was killed prematurely.
13	**Error**	Died during startup - possible SSH known_hosts mismatch.	Server was reachable, TCP was available on the SSH port specified, but The Machine could not connect. Usually indicates that the underlying SSH application refused to connect – often due to mismatched fingerprints.
14	**Error**	Server didn't come back after a controlled reboot.	Host is either taking exceptionally long to respond after a reboot or the host has hung. Occasionally caused by very long disk checks on the Remote Server.
15	**Error**	Failed to decrypt SSH key. Wrong key unlock password?	You specified the wrong passphrase to decrypt your SSH key, or the key is corrupt.
16	**Error**	You cannot specify both interactive and key credentials, pick one.	You specified both key authentication and password based authentication. You can't eat all the cake by yourself.
17	**Error**	Fingerprint mismatch with database. Clear or modify the recorded fingerprint for the host.	The fingerprint of the host does not match the fingerprint recorded in the database. Either the fingerprint on the host has changed or the host is not the one you expected.
99	**Error**	My head fell off. I don't know why.	The Job Processor determined that there was a fault but wasn't able to decide how to handle it, so it exited.
Unhandled Exception	**Error**	Unhandled exit code. This is not supposed to happen.	No Job Code was discovered so the Job Controller was unable to determine the Job's status. This should never happen normally - if it does, somebody may have manually changed the values in the database, the connection to the database may have been lost mid-transaction, or The Machine may have consumed a Cheerio/Saveloy sausage. Those things are nasty.

7.6   Tagging System

D-Shell has a tagging system that extends the capabilities of regular scripts. All tags should be created on a new line without spaces before the tag. The only exception is the *REBOOT tag which can be used anywhere, even within a conditional statement.

7.6.1   *VSNAPSHOT

The *VSNAPSHOT tag enables you to control VMWare snapshots from within Jobs. It currently takes four options; COUNT, TAKE, REMOVE and REMOVEALL. The successful execution of VMWare snapshot operations is dependent on VMWare shell access being sufficiently granted to TM, and that the system being snapshotted is on that VMWare system.

7.6.1.1   *VSNAPSHOT COUNT

This tag counts the current number of snapshots that exist for the system on which the Job is run. It will output the current total number of snapshots the system has, the current number of snapshots that were taken by TM that the system has, and the VMWare node where the system is currently running. You use this tag as is – it take no further options.

7.6.1.2   *VSNAPSHOT TAKE

This tag takes a snapshot of the system on which the Job is running. TM will first try to take a full memory snapshot even on system with independently attached disks. If this fails, then TM will take a regular snapshot that does not include memory. You use this tag as is – it take no further options.

7.6.1.3   *VSNAPSHOT REMOVE

This tag removes all snapshots taken by TM of the system on which the Job is running. It does not remove any other snapshots, including those taken with VMWare access tools or by other applications, such as backup systems. You use this tag as is – it take no further options.

7.6.1.4   *VSNAPSHOT REMOVEALL

This tag removes all snapshots of the system on which the Job is running. You use this tag as is – it take no further options.

7.6.2   *PAUSE

The *PAUSE tag is useful for creating an artificial wait on TM when processing a Job. It is similar to the *sleep* command, however the delay is created on the TM side of the connection. This is useful when working with heavily loaded systems or where you want to ensure that the load of the remote system remains as low as possible, as otherwise TM will immediately send the next command as soon as the remote prompt is seen like what would happen in a regular script. If you wish to create a delay on the remote system, you should issue a regular *sleep* command. To use this tag correctly, you should follow it with a space and the number of seconds you wish for TM to pause for before sending the next command; e.g.:

- ***PAUSE** 10   # Waits for 10 seconds
- ***PAUSE** 120   # Waits for 2 minutes

There is no limit on the number of seconds that you can specify, however do be mindful that the remote system may terminate the SSH connection (due to inactivity) if the wait is significant.

7.6.3   *SEND

The *SEND tag is used mostly in combination with the *WAITFOR tag, although it can be used by itself. It simulates the behaviour of a user throwing a set of commands at a system at once without waiting for the remote prompt to appear. The remote system will still behave normally and process each command in turn, but TM will not wait for the remote prompt to appear before sending the next command. This can be used to great affect when combined with *WAITFOR as TM will no longer watch for the remote prompt but instead watch for the string listed under *WAITFOR. This is covered in more detail in the next section. The output of the *SEND command is also ignored, and no exit code is sought, so it is particularly useful when sending a block of text where each line doesn't need to be checked for an exit code – a fine example is when using cat to push a multiline string to a file; this is also covered below. To use this tag correctly, you should follow it with a space and the string you wish TM to send. Consider the following scenarios when issuing the example commands to a remote system:

7.6.3.1   Command processing without using *SEND

Commands submitted as:

command 1
command 2

The process executes as:

1. TM sends *command 1* and waits for the remote prompt to appear, signalling that the command has finished processing.
2. Remote prompt appears, TM captures the output of the command.
3. TM then checks the exit code of the previous command (and acts accordingly, depending on the conditions set in On Failure).
4. TM reports both the output of the command and the exit code to the database.

5. TM sends *command 2* and waits for the remote prompt to appear, signalling that the command has finished processing.
6. Remote prompt appears, TM captures the output of the command.
7. TM then checks the exit code of the previous command (and acts accordingly, depending on the conditions set in On Failure).
8. TM reports both the output of the command and the exit code to the database.

7.6.3.2   Command processing when using *SEND

Commands submitted as:

SEND command 1
SEND command 2

The process executes as:

1. TM sends *command 1* and does not wait for the remote prompt or discover the exit status. No output is recorded.
2. TM sends *command 2* and does not wait for the remote prompt or discover the exit status. No output is recorded.

7.6.3.3   Working with multiline commands

Occasionally it's useful to use multiline commands instead of several single lines, such as when pushing a multiline block of text into a file. This is best used with *SEND because each line may not need to be checked for successful execution, or it may not be possible at all.

An example of using *SEND to deal with multiline commands:

```
*SEND cat << _EOF_ > /tmp/a-file.txt
*SEND Hello,
*SEND This is the story of
*SEND a multiline file.
*SEND It was a short story.
*SEND _EOF_
```

And another example with formatting:

```
*SEND cat << _EOF_ > /etc/httpd/conf.d/data.nwk1.com.conf
*SEND <VirtualHost *:80>
*SEND    ServerName          data.nwk1.com
*SEND
```

```
*SEND   <IfModule mod_rewrite.c>
*SEND      RewriteEngine   On
*SEND      RewriteCond          %{HTTPS} off
*SEND      RewriteRule          (.*)   https://%{HTTP_HOST}%{REQUEST_URI} [R=301,L]
*SEND   </IfModule>
*SEND   <IfModule !mod_rewrite.c>
*SEND      Redirect             /        https:// data.nwk1.com
*SEND   </IfModule>
*SEND </VirtualHost>
*SEND _EOF_
```

7.6.4   *WAITFOR

When combined with *SEND, *WAITFOR is a very powerful tool as it enables you to drive the command line as though a person was operating it. For instance, actions that would otherwise not be achievable (or easily achievable) with a script, like interactive prompts, are now possible. *WAITFOR will also match against regular expressions that you pass it. To use this tag correctly, you should follow it with a space and the string you wish TM to watch for.

7.6.4.1   *WAITFOR Examples

Command	Description
*WAITFOR password:	Will wait for a typical password prompt before sending a password.
*WAITFOR Is this ok?	Waits for the prompt 'Is this OK?' to appear on the console before sending your response.
*WAITFOR mysql>	Matches the MySQL prompt.
*WAITFOR \d{4}-\d\d-\d\d	Matches a date string like 2016-08-02, but will also match 1234-56-78. Beware of the ~~dragon~~ regex.

7.6.4.2   *WAITFOR##

*WAITFOR has a default timeout that's set globally. You can override this on an individual basis by specifying a timeout as part of the *WAITFOR tag (before the space that separates the tag and the string). For example, if you want to wait for a time of as much as 120 seconds for the output 'Hello', you would issue a wait that looked like this:

*WAITFOR120 Hello

7.6.4.3   Combining *SEND and *WAITFOR

Combining *SEND and *WAITFOR allows you to drive the prompt as though it was interactive. Underneath, D-Shell uses Expect as well as some additional trickery and regex, so if you're familiar with Expect the concept is the same. By default, and unlike Expect, D-Shell uses greedy matching to make it much easier to use than Expect, which is notoriously difficult to get right.

The example below uses *SEND and *WAITFOR together to create a certificate signing request. Yes yes, I know that you can pass all parameters on one line for this, however this is a common operation so this serves as a sensible example:

```
*SEND openssl req -nodes -newkey rsa:2048 -keyout magic-cert.key -out magic-cert.csr
*WAITFOR Country Name
*SEND NZ
*WAITFOR State or Province Name
*SEND Wellington
*WAITFOR Locality Name
*SEND Wellington
```

```
*WAITFOR Organization Name
*SEND Magic Org
*WAITFOR Organizational Unit Name
*SEND Magic Unit Thing
*WAITFOR Common Name
*SEND magic-server.nwk1.com
*WAITFOR Email Address
*SEND ben@nwk1.com
*WAITFOR A challenge password
*SEND
*WAITFOR An optional company name
*SEND
```

And here's another example that cannot be done by a regular script, have options echoed into it, or otherwise done reliably without treating it interactively. So here's how to do it, interactively:

```
*SEND mysql_secure_installation
*WAITFOR Enter current password
*SEND
*WAITFOR Set root password
*SEND y
*WAITFOR New password
*SEND ThePassword
*WAITFOR Re-enter new password
*SEND ThePassword
*WAITFOR Remove anonymous users
*SEND y
*WAITFOR Disallow root login remotely
*SEND y
*WAITFOR Remove test database and access to it
*SEND y
*WAITFOR Reload privilege tables now
*SEND y
```

7.6.5   *REBOOT

The *REBOOT tag is unlike other tags in that it can be used anywhere within a script instead of only at the beginning of a line. For instance, you can use it inside of a conditional statement so that a reboot is only performed *if* the tag is met during processing.

The *REBOOT tag is both a reboot instruction for the Remote Server and a notification to the Job Processor that the system is likely to reboot unexpectedly – therefore making the reboot expected. Because the tag can be used inside conditional statements (like *if* or *case* blocks) where the outcome is determined by the Remote Server, the *REBOOT tag acts as an advisory to D-Shell that it *may* lose the connection. D-Shell therefore watches the connection and, if the connection fails, watches for the server's SSH service to return and then attempts to re-establish the connection. Once the connection is re-established, D-Shell continues processing the rest of the script.

*REBOOT should always be used in place of standard 'reboot', 'shutdown -r' or similar commands so that D-Shell is notified that a lost connection is expected and therefore doesn't assume a connection error if the Remote Server connection were to disappear unexpectedly. Like all other tags, this tag is stripped before it is sent to the Remote Server; in its place, 'shutdown -r 1' is sent. D-Shell then watches the console output stream for a moment for signs of a reboot in progress. If the Remote Server appears to be rebooting, D-Shell starts the connection recovery process and begins to poll the server and establishes a new connection. If the system does not appear to be rebooting, D-Shell continues with other commands as normal.

The *REBOOT tag can be used by itself as simply as:

```
*REBOOT
```

The above would make D-Shell aware of a possible reboot, and send 'shutdown -r 1' to the Remote Server. The Remote Server will then reboot, and D-Shell will correctly handle the re-connection.

Here's an example of where a *REBOOT tag would be useful in a conditional context. The command 'needs-restarting' checks for services that need to be restarted; it is usually run after a system is updated (similar commands are 'zypper ps' and 'checkrestart'). When run, if the output is empty, the system does not need rebooting and the command echo's a statement. If the output wasn't empty, a reboot is triggered:

```
if [[ `needs-restarting | wc -l` == 0 ]]; then echo 'Does not need rebooting.'; else *REBOOT; fi
```

As D-Shell did not know if the Remote Server needed rebooting (because the reboot was conditional on services on the Remote Server requiring a restart) it instead prepared itself for the *possibility* of a reboot and therefore handles both scenarios – reboot or no reboot – correctly.

7.6.6   Stripping Tags

To strip the tags from a script built for The Machine and turn it back into a regular shell script, run the following regular expressions against the script.

Tag	Regex	Replaced with	Notes
*VSNAPSHOT	s/^*VSNAPSHOT.*//g	*NULL*	There is no shell equivalent, so it is stripped.
*PAUSE	s/^*PAUSE/sleep/g or s/^*PAUSE.*//g	sleep or *NULL*	Not really a shell equivalent since the pause occurs on The Machine, but sleep partly replicates the behaviour on the remote server (in most circumstances). You have two options – the first replaces *PAUSE with sleep, the second strips *PAUSE entirely.
*SEND	s/^*SEND\s//g	Rest of string	Replaced with raw shell.
*WAITFOR	s/^*WAITFOR.*//g	*NULL*	There is no shell equivalent, so it is stripped.
*REBOOT	s/*REBOOT/reboot/g	shutdown -r	Tag replaced with 'shutdown -r'

8   DNS System

9 Reverse Proxy System

9.1

Comma separate server name / alias.

10 DSMS System

10.1 Currently Distributed Sudoers File

10.1.1 Sudoers Build Structure

The sudoers file is built with a common structure every time to make for easier reading, as each item is grouped by component type, the component order is always the same, and each item is always recorded alphabetically.

10.1.1.1 Sectional Markings

Each section is clearly separated by sectional markings, which detail where one section begins and the next section ends. 6.1.1.1a shows an example of a sectional note, which highlights that the items below it are Rules.

6.1.1.1a - Sectional Markings

10.1.1.2 Environmental Defaults

Environmental Defaults, set in the Setting Environmental Defaults section, are listed at the top of the sudoers file. All currently active environmental defaults are highlighted in green, as illustrated in 6.1.1.2a.

6.1.1.2a - An Environmental Default

10.1.1.3 Host Groups

Host Groups are defined as sudo aliases after the Environmental Defaults, and each host name belonging to the group is added, along with its IP address. Host Groups are coloured orange. 6.1.1.3a shows an example of this.

6.1.1.3a - An example Host Group in a sudoers file

All aliases are capitalised automatically, as this is a requirement of sudo. All aliases are also prefixed with their type (in this case, HOST_GROUP_) to make following the flow of the file easier, and to avoid clashes between equally named groups (such as Host Groups and Command Groups with the same name).

10.1.1.4 User Groups

User Groups are defined as sudo aliases after the Host Groups, and each user name belonging to the group is added. System groups are also given their own sudoers user alias for consistency and for easier tracking by the internal DSMS System group name. User Groups are coloured purple.

All aliases are capitalised automatically, as this is a requirement of sudo. All aliases are also prefixed with their type (in this case, USER_GROUP_) to make following the flow of the file easier, and to avoid clashes between equally named groups (such as User Groups and Host Groups with the same name).

10.1.1.5 Command Groups

Command Groups are defined as sudo aliases after the User Groups, and each Command belonging to the group is added *as an alias of the Command*. The reason for this is that commands themselves already have a command alias, which is defined in the Commands section. Commands need to be referred to by an alias, as they can often contain spaces and other characters that cannot be used as a sudo reference. Command Groups and Commands are coloured yellow.

All aliases are capitalised automatically, as this is a requirement of sudo. All aliases are also prefixed with their type (in this case, COMMAND_GROUP_) to make following the flow of the file easier, and to avoid clashes between equally named groups (such as Command Groups and User Groups with the same name).

10.1.1.6 Commands

Commands are defined as sudo aliases after the Command Groups. Commands are uniquely configured relative to Hosts and Users, in that commands must themselves be referred to with an alias. Commands need to be referred to by an alias, as they can often contain spaces and other characters that cannot be used as a sudo reference. Command Groups and Commands are coloured yellow.

All aliases are capitalised automatically, as this is a requirement of sudo. All aliases are also prefixed with their type (in this case, COMMAND_) to make following the flow of the file easier, and to avoid clashes between equally named groups (such as Commands and Command Groups with the same name).

10.1.1.7 Rules

In TM Rules are made up of four components:

- A final Host Alias of all existing Host Aliases attached to this Rule. The name of this final Host Alias follows the standard name of Host_Rule_Group_<Rule_ID>, where <Rule_ID> is the unique database ID assigned to the Rule.

- A final User Alias of all existing User Aliases attached to this Rule. The name of this final User Alias follows the standard name of User_Rule_Group_<Rule_ID>, where <Rule_ID> is the unique database ID assigned to the Rule.

- A final Command Alias of all existing Command Aliases attached to this Rule. The name of this final Command Alias follows the standard name of Command_Rule_Group_<Rule_ID>, where <Rule_ID> is the unique database ID assigned to the Rule.

- The Rule Line, which contains a combination of the final Host, User and Command Aliases, plus a 'Run As' user and option tags.

You can see an example Rule in 6.1.1.7a.


```
## ApacheControl (ID: 1), does not expire, last modified 2014-10-15 15:31:22 by Ben Schofield, last approved 2014-10-15 15:31:29 by Ben Schofield
Host_Alias   HOST_RULE_GROUP_1 = HOST_GROUP_APPLICATIONSERVERS
User_Alias   USER_RULE_GROUP_1 = USER_GROUP_UNIXADMINISTRATORS
Cmnd_Alias   COMMAND_RULE_GROUP_1 = COMMAND_GROUP_APACHECOMMANDS
USER_RULE_GROUP_1   HOST_RULE_GROUP_1 = (root) PASSWD:EXEC: COMMAND_RULE_GROUP_1
```

6.1.1.7a - An example Rule

Note that the Host, User and Command Aliases follow the colour defaults for each component. Safe options are shown as a light blue (e.g. PASSWD) and unsafe options are shown in red (e.g. EXEC). Any command to be run as 'root', such as the command above, has the run as component coloured red.

All Rules that are set Active and have been Approved are included in the final sudoers file - even incomplete Rules. A Rule is defined as incomplete if it lacks one or more of the following requirements:

- At least one Host Group or at least one Host

- At least one User Group or at least one User

- At least one Command Group or at least one Command

Rules that are determined incomplete by TM are highlighted with red hash tags before and after for clarity. As far as the sudo application is concerned, these lines are comments and are not read as a legitimate configuration. An example of an incomplete Rule is shown in 6.1.1.7b.

```
########
######## MySQLControl (ID: 2) was not written because the rule is not complete. It lacks defined Hosts, Users or Commands. ########
########
```

6.1.1.7b - An example of an incomplete Rule

10.1.2 Viewing the Currently Distributed Sudoers File (Web Panel)

To view the CDSF, navigate to the /index.cgi page directly, or through the menu at Home. The CDSF displayed represents the latest compiled sudoers file that is actively distributed to Remote Server - it does not necessarily represent the latest configuration. The latest configuration is compiled into the sudoers file when all Rules have been approved.

Above the CDSF, there are two parameters displayed, the CDSF's build timestamp and the MD5 checksum of the CDSF.

The timestamp details when the CDSF was last compiled by the DSMS System. A new CDSF is compiled when there are changes to the system, and those changes are eligible to be included in the CDSF file. Examples of items that are not eligible to be included in the CDSF file are Inactive items, Expired items, or incomplete Rules.

The MD5 is a checksum of the CDSF to ensure data integrity, auditability and legacy sudoers backup tracking. It is also used to identify which sudoers file was last distributed to each Remote Server, which can be viewed by DSMS Administrators on the Distribution Status page.

10.1.3 Viewing the Currently Distributed Sudoers File (Command Line)

As root on the TM Server, run the following to view the CDSF; you may need to modify the path to match your HTTP root path:

```
cat /var/www/html/sudoers
```

You may wish to pipe the output to 'more' or 'less' if you have a small scroll back limit on your terminal.

10.2 Legacy Sudoers File Storage

10.2.1 Replaced Sudoers Files

As each new CDSF is built, a copy of it is stored in the Sudoers Storage directory, which is */var/www/html/sudoers-storage* by default but can be configured as defined in the Sudoers_Storage section. The newly built sudoers file's name is defined as sudoers_<MD5SUM>, where <MD5SUM> is the MD5 checksum of the newly built CDSF. Each copy is an exact clone of the CDSF, and is therefore plain text and can be read by any text viewer.

10.2.2 Broken Sudoers Files

Theoretically, TM cannot generate syntactically incorrect sudoers file because it is programmed to write only syntactically valid sudoers file. However, if the sudo specification changes and TM is not updated, or TM becomes internally corrupt, or the new sudoers file becomes corrupt on disk before it has been fully deployed, there is a safety mechanism that prevents a syntactically incorrect file from being deployed.

If a syntactically incorrect sudoers file is detected, it is stored in the Sudoers Storage directory as broken_<MD5SUM>, where <MD5SUM> is the MD5 checksum of the broken sudoers file. A 'Deployment Failed' message is also sent to the Audit Log with a description of the failure.

A second safety mechanism is also used on the Remote Server to prevent a syntactically incorrect or corrupt CDSF from overwriting */etc/sudoers* in case the CDSF was corrupt during transfer, or corrupt on the Remote Server's disk.

10.3 Sudoers File Deployment

10.3.1 Sudoers Build Process

The CDSF build mechanism runs regularly, if cron is configured correctly according to the Cron Configuration section of TM installation. The Sudoers Build file is called *sudoers-build.pl* and usually runs as root. The Sudoers Build file requires the *common.pl* file from the HTTP root directory in order to run correctly, as it uses the common configuration file to determine the database's connection parameters, the default sudoers file name and location, the default sudoers storage location, the DSMS System's name, the DSMS System's current version, the final ownership and group ownership of the CDSF, and the defined locations of the 'md5sum', 'cut', 'visudo', 'cp', 'ls', 'grep' and 'head' server applications.

The Build Process first checks to see if TM is in Maintenance Mode, and abandons the sudoers file build immediately if the system is observing Maintenance Mode to prevent any unexpected changes from taking place.

The Build Process then compiles the CDSF using the configuration in the database. Once the compile is complete, the Build Process checks that the file is syntactically correct and will also detect if the CDSF is corrupt. If the CDSF is syntactically correct, a copy is made as described in the Replaced Sudoers Files section, otherwise the steps in the Broken Sudoers Files section are invoked.

An audit of the event is stored in the Audit Log. If no changes were made to the CDSF since the last time the file was built (i.e. the checksums match), a new Audit Log entry is not made. This is to prevent the Audit Log from filling up with unnecessary copies of the same successful (or unsuccessful) build report.

The Build Process works independently of the Web Panel and does not require Apache to be running.

If the Build Process exits cleanly; that is, if it completes in full and finds no errors in the CDSF, it returns an exit code of 0, which, if the Cron Configuration is correct, will then allow Distribution Process to begin.

10.3.2 CDSF Distribution Process

The Distribution Process mechanism runs, if cron is configured correctly according to the Cron Configuration section of TM installation, immediately after the Sudoers Build Process. The Distribution file is called *distribution.pl* and usually runs as root. The Distribution file requires the *common.pl* file from the HTTP root directory in order to run correctly, as it uses the common configuration file to determine the database's connection parameters, the default sudoers file name and location, the default sudoers storage location, and the defined locations of the 'md5sum' and 'cut' server applications.

The Distribution Process determines the hosts to distribute the CDSF to from the database, as well as the distribution parameters for each host: the transport username, the identity key path, the connection timeout value and the remote sudoers storage path. All transfers use the SFTP subsystem of SSH. It is advised to allow only SFTP subsystem connections for the transport user.

Each part of the connection and transfer process is verified against success qualifiers, such as a maximum response time, a routable Remote Server or a writable remote directory. If any part of the connection or transfer process does not meet these minimum qualifiers, the connection or transfer is assumed dead or failed and an error is recorded in the Distribution Status page. Common error messages are described in the Diagnosing Failed Transfers section. Successful transfers are also recorded in the Distribution Status page with an MD5 checksum of the successfully deployed CDSF.

10.3.2.1 CDSF Distribution with chroot

Distribution with chroot is advised, according to the sshd_config Configuration and Transport Directory Configuration sections. For chroot jail CDSF deployments, the remote sudoers storage path must be relative to the chroot jail, with the prefixed slash removed.

10.3.2.2 CDSF Distribution without chroot

Where a chroot jail is not supported or not configured on a Remote Server, the remote sudoers storage location must be a full system path.

10.3.3 Remote Server CDSF Collection

After the CDSF file has successfully reached the Remote Server, the Remote Server collects the file and moves it to */etc/sudoers.* This is usually done by a cron job, as described in Cron Configuration of the Remote Server Manual Installation section. The reason that this is performed by the Remote Server's cron process is because */etc/sudoers* file requires root privileges to write to it. Giving the CDSF's transport user root or root equivalent privileges on each Remote Server was decidedly unideal.

A second safety mechanism is also used on the Remote Server to prevent a syntactically incorrect or corrupt CDSF from overwriting */etc/sudoers* in case the CDSF was corrupt during transfer, or corrupt on the Remote Server's disk. A corrupt or syntactically incorrect CDSF will not replace the existing */etc/sudoers* file on the Remote Server if cron was configured correctly according to the Cron Configuration section.

10.4 Hosts

10.4.1 Viewing Hosts

To view Hosts, navigate to the /sudoers-hosts.cgi page directly, or through the menu at Hosts.

The Local Search filter searches strings in the following fields:

- ID
- Hostname
- IP
- Expiry

Searches are case-insensitive. Matching patterns are highlighted according to the Common Colour Indications defaults. Row ordering is by Host Name ascending.

The Hosts table contains a list of sudoers hosts. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this Host entry. It serves no informational value, other than as a reference point when highlighting a Host.
Host Name	The Host Name is the name of the host in this sudoers entry. It should match the host's name exactly as it will be read by sudo. If it does not match exactly, Rules designed for this host may not work. You should consider using a FQDN.
IP Address	The IP Address is the IP address of the host in this sudoers entry. It should match the host's external IP address exactly as it will be read by sudo and also forms part of the connection string in the CDSF Distribution Process. If it does not match exactly, Rules destined for this host may not work.
Expires	The Expires entry details on what date the Host will expire. When a Host expires, it is removed from the CDSF. Expired Hosts are highlighted in grey. Hosts that do not expire show an expiry of 'Never'.

Active	The Active column describes whether or not the Host is eligible for CDSF inclusion. Inactive Hosts are removed from the CDSF, but retain Rule and Group memberships.
Last Modified	This column defines when this Host entry was last modified.
Modified By	This column defines who this Host was last modified by.

10.4.2 Adding Hosts

To add a Host to the DSMS System, click the 'Add New Host' button in the top centre of the screen (see 6.4.2a). A new window titled 'Add New Host' will appear with form elements requesting new host details. When filling in the requested details, you must be accurate - Host Names are used by sudo, and IP addresses are used by the Distribution System. Host Names and IPs must be unique and POSIX compliant. Hosts with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.
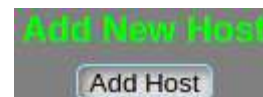
6.4.2a - Add New Host

New Hosts are added into the Distribution System automatically. All Host additions are logged to the Audit Log.

10.4.3 Editing Hosts

Hosts can be edited by using either the edit dropdown menu shown in 6.4.3a, or by using the Edit icon next to the Host's details on the Hosts page. Host Names and IPs must be unique and POSIX compliant. Hosts with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.
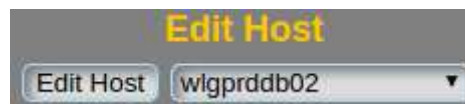
6.4.3a - Edit a Host

All Host edits are logged to the Audit Log. Any Rule that this Host modified will immediately lose its Approved status to prevent potential system abuse. You will then become the user that last modified any automatically Unapproved Rule to prevent you from re-Approving that Rule change without a second Approver's oversight.

10.4.4 Deleting Hosts

Hosts can be deleted by using the Delete icon next to the Host's details on the Hosts page. Deleted Hosts cannot be recovered.

All Host deletes are logged to the Audit Log.

10.4.5 Viewing Host Notes

Notes can be assigned against Hosts to assist with tracking changes, or to attribute changes to a Change Order or a Work Order number. To view notes, click the 'Notes' button next to the item who's notes you want to view. A window will display with the notes relevant to that particular item.

10.4.6 Adding Host Notes

Whilst viewing notes, you can add a new note to an item by adding your note to the text box above the notes table and clicking Submit New Note. The new note is added immediately, and displayed in the notes table.

10.5 Host Groups

10.5.1 Viewing Host Groups

To view Host Groups, navigate to the /sudoers-host-groups.cgi page directly, or through the menu at Groups -> Host Groups.

The Local Search filter searches strings in the following fields:

- ID
- Group Name
- Expiry

Searches are case-insensitive. Matching patterns are highlighted according to the Common Colour Indications defaults. Row ordering is by Group Name ascending.

The Host Groups table contains a list of grouped Hosts. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this Group entry. It serves no informational value, other than as a reference point when highlighting a Group.
Group Name	The Group Name is the name of the group in this sudoers entry.
Connected Hosts	The Connected Hosts column details which Hosts are connected to this Group.
Expires	The Expires entry details on what date the Group will expire. When a Group expires, it is removed from the CDSF. Expired Groups are highlighted in grey. Groups that do not expire show an expiry of 'Never'.
Active	The Active column describes whether or not the Group is eligible for CDSF inclusion. Inactive Groups are removed from the CDSF, but retain Rule memberships and Connected Hosts.
Last Modified	This column defines when this Group entry was last modified.
Modified By	This column defines who this Group was last modified by.

10.5.2 Adding Host Groups

To add a Group to the DSMS System, click the 'Add New Group' button in the top centre of the screen (see 6.5.2a). A new window titled 'Add New Group' will appear with form elements requesting new group details. Group Names must be unique and contain only a-z, A-Z, 0-9 and _ characters. Groups with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.


6.5.2a - Add New Group

10.5.3 Attaching Hosts to New Host Groups

To add Hosts to this Host Group, select each applicable Host from the dropdown menu on the 'Add New Group' page; see 6.5.3a for an example. For each new Host selection, the new Host will appear in the Attached Hosts category, accompanied by its IP address. Attached Active Hosts are coloured green, Inactive Hosts are coloured red, and expired Hosts are
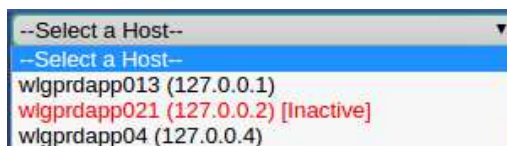
6.5.3a - Adding Hosts to a Host Group

coloured grey in keeping with the DSMS System's Common Colour Indications. For an example of this, see 6.5.3b.

All Group additions are logged to the Audit Log.

6.5.3b - Hosts attached to a Host Group

10.5.4   Editing Host Groups

Host Groups can be edited by using either the edit dropdown menu shown in 6.5.4a, or by using the Edit icon next to the Group's details on the Host Groups page. Group Names must be unique and contain only a-z, A-Z, 0-9 and _ characters. Groups with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

6.5.4a - Editing a Host Group

All Group edits are logged to the Audit Log. Any Rule that this Group modified will immediately lose its Approved status to prevent potential system abuse. You will then become the user that last modified any automatically Unapproved Rule to prevent you from re-Approving that Rule change without a second Approver's oversight.

10.5.5   Attaching Hosts to Existing Host Groups

To add Hosts to an existing Host Group, select each applicable Host from the dropdown menu on the 'Edit Group' page. For each new Host addition, the new Host will appear in the New Hosts category, accompanied by its IP address. Hosts already attached to this Group will appear under the Existing Hosts category. Attached Active Hosts are coloured green, Inactive Hosts are coloured red, and expired Hosts are coloured grey in keeping with the DSMS System's Common Colour Indications. For an example of this, see 6.5.5a.

6.5.5a - Existing and newly attached Hosts in a Host Group

10.5.6   Deleting Attached Hosts from the Group

To delete a Host from a Host Group, view the Group on the Host Groups page and click the [Remove] tag next to each item you want removed from the Group. See an example of the [Remove] tag in 6.5.6a. There is no action confirmation for removing items from a Group - they are instantly dropped.

6.5.6a - Remove an attached Host from the Group

10.5.7   Deleting Host Groups

Groups can be deleted by using the Delete icon next to the Group's details on the Host Groups page. Deleted Groups cannot be recovered.

All Group deletes are logged to the Audit Log.

10.5.8   Viewing Host Group Notes

Notes can be assigned against Host Groups to assist with tracking changes, or to attribute changes to a Change Order or a Work Order number. To view notes, click the 'Notes' button next to the item who's notes you want to view. A window will display with the notes relevant to that particular item.

10.5.9   Adding Host Group Notes

Whilst viewing notes, you can add a new note to an item by adding your note to the text box above the notes table and clicking Submit New Note. The new note is added immediately, and displayed in the notes table.

10.6 Users

10.6.1 Viewing Users

To view Users, navigate to the /sudoers-users.cgi page directly, or through the menu at Sudo Users.

The Local Search filter searches strings in the following fields:

- ID
- User Name
- Expiry

Searches are case-insensitive. Matching patterns are highlighted according to the Common Colour Indications defaults. Row ordering is by User Name ascending.

The Users table contains a list of sudo users. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this User entry. It serves no informational value, other than as a reference point when highlighting a User.
User Name	The User Name is the name of the user in this sudoers entry. It should match the user's name exactly as it will be read by sudo. If it does not match exactly, Rules designed for this user may not work.
Expires	The Expires entry details on what date the user will expire. When a User expires, it is removed from the CDSF. Expired Users are highlighted in grey. Users that do not expire show an expiry of 'Never'.
Active	The Active column describes whether or not the User is eligible for CDSF inclusion. Inactive Users are removed from the CDSF, but retain Rule and Group memberships.
Last Modified	This column defines when this User entry was last modified.
Modified By	This column defines who this User was last modified by.

10.6.2 Adding Users

To add a User to the DSMS System, click the 'Add New User' button in the top centre of the screen. A new window titled 'Add New User' will appear with form elements requesting new user details. When filling in the requested details, you must be accurate - User Names are used by sudo. User Names must be unique and POSIX compliant. Users with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

All User additions are logged to the Audit Log.

10.6.3 Editing Users

Users can be edited by using either the edit dropdown menu in the top right of the screen, or by using the Edit icon next to the User's details on the Sudo Users page. User Names must be unique and POSIX compliant. Users with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

All User edits are logged to the Audit Log.

10.6.4 Deleting Users

Users can be deleted by using the Delete icon next to the User's details on the Sudo Users page. Deleted Users cannot be recovered.

All Sudo User deletes are logged to the Audit Log. Any Rule that this User modified will immediately lose its Approved status to prevent potential system abuse. You will then become the user that last modified any automatically Unapproved Rule to prevent you from re-Approving that Rule change without a second Approver's oversight.

10.6.5 Viewing User Notes

Notes can be assigned against Users to assist with tracking changes, or to attribute changes to a Change Order or a Work Order number. To view notes, click the 'Notes' button next to the item who's notes you want to view. A window will display with the notes relevant to that particular item.

10.6.6 Adding User Notes

Whilst viewing notes, you can add a new note to an item by adding your note to the text box above the notes table and clicking Submit New Note. The new note is added immediately, and displayed in the notes table.

10.7 User Groups

10.7.1 User Group Types

There are two types of User Groups: DSMS User Groups and System Groups. By default, User Groups are DSMS Groups, which means that the group definitions are made up from the configuration in TM and remain as groups on Remote Servers regardless of the Remote Server's configuration. System Groups inherit user membership from the Remote Server's */etc/group* file, and membership may differ for each Remote Server, so Connected Users are therefore not displayed for System Groups.

10.7.2 Viewing User Groups

To view User Groups, navigate to the /sudoers-user-groups.cgi page directly, or through the menu at Groups -> User Groups.

The Local Search filter searches strings in the following fields:

- ID
- Group Name
- Expiry

Searches are case-insensitive. Matching patterns are highlighted according to the Common Colour Indications defaults. Row ordering is by Group Name ascending.

The User Groups table contains a list of grouped Users. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this Group entry. It serves no informational value, other than as a reference point when highlighting a Group.
Group Name	The Group Name is the name of the group in this sudoers entry. The User Group Type is highlighted by the Group Name's prefix: Sudoers Groups have no prefix, which System Groups have a % prefix.

Connected Users	The Connected Users column details which Users are connected to this Group. System Groups do not have any Connected Users as Users in System Groups are inherited from the Remote Server.
Expires	The Expires entry details on what date the Group will expire. When a Group expires, it is removed from the CDSF. Expired Groups are highlighted in grey. Groups that do not expire show an expiry of 'Never'.
Active	The Active column describes whether or not the Group is eligible for CDSF inclusion. Inactive Groups are removed from the CDSF, but retain Rule memberships and Connected Users.
Last Modified	This column defines when this Group entry was last modified.
Modified By	This column defines who this Group was last modified by.

10.7.3 Adding User Groups

To add a Group to the DSMS System, click the 'Add New Group' button in the top centre of the screen (see 6.7.3a). A new window titled 'Add New Group' will appear with form elements requesting new group details. Group Names must be unique and contain only a-z, A-Z, 0-9 and _ characters. Groups with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

6.7.3a - Add New Group

To change the group's type from DSMS Group to System Group, check the System Group checkbox. System Groups cannot have Users allocated to them in TM - any users previously allocated before the checkbox was checked will be removed. The checkbox is shown in 6.7.3b.

6.7.3b - Assigning System Group status to a User Group

All Group additions are logged to the Audit Log.

10.7.4 Attaching Users to New User Groups

To add Users to this User Group, assuming that you have not specified this Group as a System Group, select each applicable User from the dropdown menu on the 'Add New Group' page. For each new User selection, the new User will appear in the Attached Users category, accompanied by its IP address. Attached Active Users are coloured green, Inactive Users are coloured red, and expired Users are coloured grey in keeping with the DSMS System's Common Colour Indications.

10.7.5 Editing User Groups

User Groups can be edited by using either the edit dropdown menu in the top right of the page, or by using the Edit icon next to the Group's details on the User Groups page. Group Names must be unique and contain only a-z, A-Z, 0-9 and _ characters. Groups with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

To change the group's type from DSMS Group to System Group, check the System Group checkbox. System Groups cannot have Users allocated to them in TM - any users previously allocated before the checkbox was checked will be removed.

All Group edits are logged to the Audit Log. Any Rule that this Group modified will immediately lose its Approved status to prevent potential system abuse. You will then become the user that last modified any automatically Unapproved Rule to prevent you from re-Approving that Rule change without a second Approver's oversight.

10.7.6 Attaching Users to Existing User Groups

To add Users to an existing User Group, assuming that you have not specified this Group as a System Group, select each applicable User from the dropdown menu on the 'Edit Group' page. For each new User addition, the new User will appear in the New Users category. Users already attached to this Group will appear under the Existing Users category. Attached Active Users are coloured green, Inactive Users are coloured red, and expired Users are coloured grey in keeping with the DSMS System's Common Colour Indications.

10.7.7 Deleting Attached Users from the Group

To delete a User from a User Group, view the Group on the User Groups page and click the [Remove] tag next to each item you want removed from the Group. See an example of the [Remove] tag in 6.7.7a. There is no action confirmation for removing items from a Group - they are instantly dropped.

[Remove]

6.7.7a - Remove an attached User from the Group

10.7.8 Deleting User Groups

Groups can be deleted by using the Delete icon next to the Group's details on the User Groups page. Deleted Groups cannot be recovered.

All Group deletes are logged to the Audit Log.

10.7.9 Viewing User Group Notes

Notes can be assigned against User Groups to assist with tracking changes, or to attribute changes to a Change Order or a Work Order number. To view notes, click the 'Notes' button next to the item who's notes you want to view. A window will display with the notes relevant to that particular item.

10.7.10 Adding User Group Notes

Whilst viewing notes, you can add a new note to an item by adding your note to the text box above the notes table and clicking Submit New Note. The new note is added immediately, and displayed in the notes table.

10.8 Commands

10.8.1 Viewing Commands

To view Commands, navigate to the /sudoers-commands.cgi page directly, or through the menu at Commands.

The Local Search filter searches strings in the following fields:

- ID
- Command Alias
- Command
- Expiry

Searches are case-insensitive. Matching patterns are highlighted according to the Common Colour Indications defaults. Row ordering is by Command Alias ascending.

The Commands table contains a list of sudo commands. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this Command entry. It serves no informational value, other than as a reference point when highlighting a Command.
Command Alias	The Command Alias is the name of the command in this sudoers entry.
Command	The Command is the command in this sudoers entry. Its first character should always be a slash (/), as sudo requires absolute paths for commands specified in sudoers.
Expires	The Expires entry details on what date the Command will expire. When a Command expires, it is removed from the CDSF. Expired Commands are highlighted in grey. Commands that do not expire show an expiry of 'Never'.
Active	The Active column describes whether or not the Command is eligible for CDSF inclusion. Inactive users are removed from the CDSF, but retain Rule and Group memberships.
Last Modified	This column defines when this Command entry was last modified.
Modified By	This column defines who this Command was last modified by.

10.8.2 Adding Commands

To add a Command to the DSMS System, click the 'Add New Command' button in the top centre of the screen. A new window titled 'Add New Command' will appear with form elements requesting new user details. When filling in the requested details, you must be accurate. Command Aliases must be unique. Do not use spaces in Command Aliases, they will be stripped. Commands with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

All Command additions are logged to the Audit Log.

10.8.3 Editing Commands

Commands can be edited by using either the edit dropdown menu in the top right of the screen, or by using the Edit icon next to the Command's details on the Commands page. Command Aliases must be unique. Do not use spaces in Command Aliases, they will be stripped. Commands with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

All Command edits are logged to the Audit Log. Any Rule that this Command modified will immediately lose its Approved status to prevent potential system abuse. You will then become the user that last modified any automatically Unapproved Rule to prevent you from re-Approving that Rule change without a second Approver's oversight.

10.8.4 Deleting Commands

Commands can be deleted by using the Delete icon next to the Command's details on the Commands page. Deleted Commands cannot be recovered.

All Command deletes are logged to the Audit Log.

10.8.5 Viewing Command Notes

Notes can be assigned against Commands to assist with tracking changes, or to attribute changes to a Change Order or a Work Order number. To view notes, click the 'Notes' button next to the item who's notes you want to view. A window will display with the notes relevant to that particular item.

10.8.6 Adding Command Notes

Whilst viewing notes, you can add a new note to an item by adding your note to the text box above the notes table and clicking Submit New Note. The new note is added immediately, and displayed in the notes table.

10.9 Command Groups

10.9.1 Viewing Command Groups

To view Command Groups, navigate to the /sudoers-command-groups.cgi page directly, or through the menu at Groups -> Command Groups.

The Local Search filter searches strings in the following fields:

- ID
- Group Name
- Expiry

Searches are case-insensitive. Matching patterns are highlighted according to the Common Colour Indications defaults. Row ordering is by Group Name ascending.

The Command Groups table contains a list of grouped Commands. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this Group entry. It serves no informational value, other than as a reference point when highlighting a Group.
Group Name	The Group Name is the name of the group in this sudoers entry.
Connected Commands	The Connected Commands column details which Command Aliases are connected to this Group, with their associated Command in brackets.
Expires	The Expires entry details on what date the Group will expire. When a Group expires, it is removed from the CDSF. Expired Groups are highlighted in grey. Groups that do not expire show an expiry of 'Never'.
Active	The Active column describes whether or not the Group is eligible for CDSF inclusion. Inactive Groups are removed from the CDSF, but retain Rule memberships and Connected Commands.
Last Modified	This column defines when this Group entry was last modified.
Modified By	This column defines who this Group was last modified by.

10.9.2   Adding Command Groups

To add a Group to the DSMS System, click the 'Add New Group' button in the top centre of the screen (see 6.9.2a). A new window titled 'Add New Group' will appear with form elements requesting new group details. Group Names must be unique and contain only a-z, A-Z, 0-9 and _ characters. Groups with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

6.9.2a - Add New Group

10.9.3   Attaching Commands to New Command Groups

To add Commands to this Command Group, select each applicable Command from the dropdown menu on the 'Add New Group' page. For each new Command selection, the new Command Alias will appear in the Attached Commands category, accompanied by its full Command. Attached Active Commands are coloured green, Inactive Commands are coloured red, and expired Commands are coloured grey in keeping with the DSMS System's Common Colour Indications.

All Group additions are logged to the Audit Log.

10.9.4   Editing Command Groups

Command Groups can be edited by using either the edit dropdown menu in the top right, or by using the Edit icon next to the Group's details on the Command Groups page. Group Names must be unique and contain only a-z, A-Z, 0-9 and _ characters. Groups with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

All Group edits are logged to the Audit Log. Any Rule that this Group modified will immediately lose its Approved status to prevent potential system abuse. You will then become the user that last modified any automatically Unapproved Rule to prevent you from re-Approving that Rule change without a second Approver's oversight.

10.9.5   Attaching Commands to Existing Command Groups

To add Commands to an existing Command Group, select each applicable Command from the dropdown menu on the 'Edit Group' page. For each new Command addition, the new Command Alias will appear in the New Commands category, accompanied by its full Command. Commands already attached to this Group will appear under the Existing Commands category. Attached Active Commands are coloured green, Inactive Commands are coloured red, and expired Commands are coloured grey in keeping with the DSMS System's Common Colour Indications.

10.9.6   Deleting Attached Commands from the Group

To delete a Command from a Command Group, view the Group on the Command Groups page and click the [Remove] tag next to each item you want removed from the Group. See an example of the [Remove] tag in 6.9.6a. There is no action confirmation for removing items from a Group - they are instantly dropped.

6.9.6a - Remove an attached Command from the Group

10.9.7   Deleting Command Groups

Groups can be deleted by using the Delete icon next to the Group's details on the Command Groups page. Deleted Groups cannot be recovered.

All Group deletes are logged to the Audit Log.

10.9.8   Viewing Command Group Notes

Notes can be assigned against Command Groups to assist with tracking changes, or to attribute changes to a Change Order or a Work Order number. To view notes, click the 'Notes' button next to

the item who's notes you want to view. A window will display with the notes relevant to that particular item.

10.9.9 Adding Command Group Notes

Whilst viewing notes, you can add a new note to an item by adding your note to the text box above the notes table and clicking Submit New Note. The new note is added immediately, and displayed in the notes table.

10.10 Rules

10.10.1 Viewing Rules

To view Rules, navigate to the /sudoers-rules.cgi page directly, or through the menu at Rules.

The Local Search filter searches strings in the following fields:

- ID
- Rule Name
- Run As
- Expiry

Searches are case-insensitive. Matching patterns are highlighted according to the Common Colour Indications defaults. Row ordering is by Rule Name ascending.

The Rules table contains a list of sudoers rules. See the table below for an explanation of each column.

Column Name	Description
ID	The ID is a unique identifier for this Group entry. It serves no informational value, other than as a reference point when highlighting a Group.
Rule Name	The Rule Name is the name of the rule in this sudoers entry.
Attached Host Groups	The Attached Host Groups column details which Host Groups are connected to this Rule.
Attached Hosts	The Attached Hosts column details which Hosts are connected to this Rule.
Attached User Groups	The Attached User Groups column details which User Groups are connected to this Rule.
Attached Users	The Attached Users column details which Users are connected to this Rule.
Attached Command Groups	The Attached Command Groups column details which Command Groups are connected to this Rule.
Attached Commands	The Attached Commands column details which Commands are connected to this Rule.
Run As	The Run As column details what user this Rule will be run as on the Remote Server.

Tags	The Tags column details which options are present on this Rule.
Expires	The Expires entry details on what date the Rule will expire. When a Rule expires, it is removed from the CDSF. Expired Rule are highlighted in grey. Rules that do not expire show an expiry of 'Never'.
Active	The Active column describes whether or not the Rule is eligible for CDSF inclusion. Inactive Rules are removed from the CDSF, but retain all of their connected items, tags and other configuration.
Approved	The Approved column details whether or not this Rule has been approved for inclusion in the CDSF.
Last Modified Last Approved	This column defines when this Rule entry was last modified and beneath it in the same cell, when this Rule was last approved.
Modified By Approved By	This column defines who this Rule was last modified by and beneath it in the same cell, who this Rule was last approved by.

10.10.2 Adding Rules

To add a Rule to the DSMS System, click the 'Add New Rule' button in the top centre of the screen (see 6.10.2a). A new window titled 'Add New Rule' will appear with form elements requesting new Rule details Rule Names must be unique and contain only a-z, A-Z, 0-9 and _ characters. Rules with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

6.10.2a - Add New Rule

To add Hosts, Users, Commands or any Groups to this Rule, select each applicable item from their respective dropdown menus on the 'Add New Rule' page; 6.10.2b shows an example of this. For each new item selection, the new item will appear in the relevant attachment category, accompanied by any other useful data associated with that item, such as IP addresses or commands, as illustrated in 6.10.2c. Attached Active items are coloured green, Inactive items are coloured red, and expired items are coloured grey in keeping with the DSMS System's Common Colour Indications.

6.10.2b - Attaching items to a new Rule

From this window, you are also able to specify the user that this Rule should Run As, and which options that this Rule should run with. The safe options are highlighted in green and are selected by default. The Run As and Options are displayed in 6.10.2d.

The PASSWD option means that the user must specify a password when running

6.10.2c - Items attached to a new Rule

the sudo command. Conversely, the NOPASSWD option means that a password isn't required and that the user can run the command without providing a password. The latter is sometimes useful for scripted situations.

Some commands may require you to turn on the less safe options, in particular, commands that invoke other applications or commands that break out to a shell (such as service control commands) may need to be run with EXEC permissions. A safe approach is to always set NOEXEC for every command, then enable it as necessary for commands that require it.

6.10.2d - Setting Run As and Options

All Rule additions are logged to the Audit Log.

10.10.3 Editing Rules

Rules can be edited by using either the edit dropdown menu in the top right, or by using the Edit icon next to the Rule's details on the Rules page. Rule Names must be unique and contain only a-z, A-Z, 0-9 and _ characters. Rules with an expiry set are automatically removed from the CDSF at 23:59:59 (or the next CDSF refresh thereafter) on the day of expiry. Expired entries are functionally equivalent to inactive entries. The expiry date entry format is YYYY-MM-DD.

As with adding a new Rule, to add Hosts, Users, Commands or Groups to an existing Rule, select each applicable item from the dropdown menu on the 'Edit Rule' page. For each new item added, the new item will appear in the relevant New category. Items already attached to this Rule will appear under the relevant Existing category. An example of this can be seen in 6.10.3a. Attached Active items are coloured green, Inactive items are coloured red, and expired items are coloured grey in keeping with the DSMS System's Common Colour Indications.

6.10.3a - Existing and New Rule Items

From this window, you are also able to specify the user that this Rule should Run As, and which options that this Rule should run with. The safe options are highlighted in green and are selected by default.

The PASSWD option means that the user must specify a password when running the sudo command. Conversely, the NOPASSWD option means that a password isn't required and that the user can run the command without providing a password. The latter is sometimes useful for scripted situations.

Some commands may require you to turn on the less safe options, in particular, commands that invoke other applications or commands that break out to a shell (such as service control commands) may need to be run with EXEC permissions. A safe approach is to always set NOEXEC for every command, then enable it as necessary for commands that require it.

All Rule edits are logged to the Audit Log.

10.10.4 Deleting Attached Items from a Rule

To delete an item from a Rule, view the Rule on the Rules page and click the [Remove] tag next to each item you want removed from the Rule. See an example of the [Remove] tag in 6.10.4a. There is no action confirmation for removing items from a Rule - they are instantly dropped.

6.10.4a - Remove an attached item from the Rule

10.10.5 Deleting Rules

Rules can be deleted by using the Delete icon next to the Rule's details on the Rules page. Deleted Rules cannot be recovered.

All Rule deletes are logged to the Audit Log.

10.10.6 Approving Rules

Any new or edited Rule must be approved by a second person before it is included in the CDSF. This is to prevent users from creating their own Rules and using those new Rules to elevate their privileges on Remote Systems. You must have Approver privileges to approve Rules.

6.10.6a - Approve Rule

Already approved Rules can be reapproved to facilitate a later manual audit of Rules.
By reapproving existing but still good Rules, the approval time of that Rule is updated, and you can then filter out and purge legacy Rules with an old approval date.

To approve a Rule, press the 'Approve Rule' button next to a Rule. See an example 'Approve Rule' button in 6.10.6a.

10.10.7 Rule Approval Auto-Revocation

Any changes to a Rule, or any changes to any items attached to a Rule, will cause the Rule to lose its Approval and Distribution will be locked until the Rule is approved. This is to prevent potential system abuse by users, for instance, changing a Command that's attached to an already Approved Rule and then being able to run that new Command on a Remote Server. You will then become the user that last modified any automatically Unapproved Rule to prevent you from re-Approving that Rule change without a second Approver's oversight.

10.10.8 Viewing Rule Notes

Notes can be assigned against Rules to assist with tracking changes, or to attribute changes to a Change Order or a Work Order number. To view notes, click the 'Notes' button next to the item who's notes you want to view. A window will display with the notes relevant to that particular item.

10.10.9 Adding Rule Notes

Whilst viewing notes, you can add a new note to an item by adding your note to the text box above the notes table and clicking Submit New Note. The new note is added immediately, and displayed in the notes table.

11 System Maintenance and References

11.1 Setting Environmental Defaults

As part of sudo, you can set environmental defaults that set a default set of requirements for all sudo actions. These values are usually only set once and seldom change, so TM uses an environmental defaults file, called '*environmental-defaults*', in the HTTP root directory that is used each time a new CDSF is built. The defaults file, by default, is exactly the same as the default configuration that's included with a standard RHEL 6 installation. The environmental defaults are always written at the top of the CDSF file. The '*environmental-defaults*' file should be edited by a System Administrator. To edit the file, as root on the TM Server, run the following:

```
cd /var/www/html
vi environmental-defaults
```

11.2 DSMS System Account Lockout

11.2.1 Conditions for Lockout

A DSMS System Account can be locked out in two ways. The first method is by an Administrator through the Account Management page.

The second method is through an automatic lockout feature invoked by five consecutive incorrect login attempts. The failed lockout counter resets after a successful login within the five login attempt threshold; the lockout counter is not time based and will not reset with the passing of time alone. If an Account is locked out, the user will not be able to login until the account is unlocked.

11.2.2 Account Lockout Reset Process

As with conditions for lockout, there are two ways to unlock an account. The first method is by an Administrator through the Account Management page. This method does not require the user's password to be reset for the Account to be unlocked.

The second unlock method does not require an Administrator's input. If the user's Account is locked out due to failed password attempts, on the failure of the final attempt TM send a password reset key to the user's registered email address. This key is randomly generated, and is used to unlock the Account. As part of the unlock process, the user is required to set a new password.

11.3 Recovering from a Crashed Build and Distribution Process

If the TM Server crashes whilst the Sudoers Build Process or CDSF Distribution Process is in progress, the automated build process may not recover automatically and may require administrative intervention. To prevent two build processes running simultaneously and writing to the same sudoers file at the same time, there is a build locking mechanism that prevents duplicate build processes from launching. Similar to the build lock is a distribution lock, which has the same effect for a distribution process. If either process is engaged, a new process of *either* type will not start to prevent distributing a partially built CDSF file.

11.3.1.1   This locking process is not handled by a local lock file for the simple reason that doing so would prevent a High Availability cluster from knowing that a build process is underway on another host. Therefore, the locking mechanism is handled in lock table of the Sudoers_Owner_ID

For changing the ownership of the sudoers file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

11.3.1.2 Sudoers_Group_ID

For changing the group ownership of the sudoers file after it's created by the DSMS build process, we need to specify a group. It is recommended to run the DSMS build process as a root cron job (as defined in Cron Configuration), but Apache will need to reach this file to display its live contents on the web panel. If you modified Apache based on the Apache Configuration section, the group ownership should usually be 'apache'. However, on some systems, Apache Server doesn't run as the 'apache' user, such as when it runs as 'httpd', so you must specify the appropriate group ownership for TM to read the sudoers file.

Defaults:

```
my $Group = 'apache';
```

11.3.1.3 DNS_Zone_Master_File

This is the zone master file use for defining zones. You should include the full path.

Defaults:

```
my $DNS_Zone_Master_File = '/etc/bind/named.conf.local';
```

11.3.1.4 DNS_Internal_Location

This is the path that the system writes the temporary Internal DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_Internal_Location = '/etc/bind/master-internal';
```

11.3.1.5 DNS_External_Location

This is the path that the system writes the temporary External DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_External_Location = '/etc/bind/master-external';
```

11.3.1.6 DNS_Internal_SOA

This is the SOA data that's written at the top of the Internal DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email = 'postmaster@example.com';
        my $TTL = '86400';              # 1 day
        my $Serial = `date +%s`;        # Epoch
                $Serial =~ s/\n//;
        my $Refresh = '10800';          # 3 hours
        my $Retry = '3600';             # 1 hour
        my $Expire = '2419200';         # 4 weeks
        my $Minimum = '86400';          # 1 day
        my $NS1 = 'ns1.example.com';
```

```
        my $NS2 = 'ns2.example.com';
        my $NS3 = 'ns3.example.com';
```

11.3.1.7 DNS_External_SOA

This is the SOA data that's written at the top of the External DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email = 'postmaster@example.com';
        my $TTL = '86400';              # 1 day
        my $Serial = `date +%s`;        # Epoch
                $Serial =~ s/\n//;
        my $Refresh = '10800';          # 3 hours
        my $Retry = '3600';             # 1 hour
        my $Expire = '2419200';         # 4 weeks
        my $Minimum = '86400';          # 1 day
        my $NS1 = 'ns1.example.com';
        my $NS2 = 'ns2.example.com';
        my $NS3 = 'ns3.example.com';
```

11.3.1.8 DNS_Storage

This is the directory where replaced DNS files are stored. You do not need a trailing slash.

Defaults:

```
my $DNS_Storage = '../Storage/DNS';
```

11.3.1.9 DNS_Owner_ID

For changing the ownership of the DNS file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

11.3.1.10 DNS_Group_ID

For changing the group ownership of the DNS file after it's created, we need to specify a group owner. It is recommended to set this to be group owned by BIND, so the group name is usually 'bind'.

Defaults:

```
my $Group = 'bind';
```

11.3.1.11 Reverse_Proxy_Location

This is the path that the system writes the temporary reverse proxy files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the reverse proxy config. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Location = '../Storage/tmp/ReverseProxy';
```

11.3.1.12 Proxy_Redirect_Location

This is the path that the system writes the temporary proxy redirect files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the proxy redirect config. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Location = '../Storage/tmp/ ReverseProxy';
```

11.3.1.13 Reverse_Proxy_Storage

This is the directory where replaced reverse proxy files are stored. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Storage = '../Storage/ReverseProxy';
```

11.3.1.14 Proxy_Redirect_Storage

This is the directory where replaced proxy redirect files are stored. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Storage = '../Storage/ReverseProxy';
```

11.3.1.15 DShell_Job_Log_Location

This is the directory where job logs are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_Job_Log_Location = '../Storage/D-Shell/Job-Log';
```

11.3.1.16 DShell_tmp_Location

This is the directory where D-Shell temporary files are are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_tmp_Location = '../Storage/D-Shell/tmp';
```

11.3.1.17 DShell_WaitFor_Timeout

The default time that a *WAITFOR statement will wait before bailing out. Can be overridden manually by issuing *WAITFORnn, where nn is the timeout in seconds. nn can be any number.

Defaults:

```
my $DShell_WaitFor_Timeout = 1800; # 30 minutes
```

DB_Management database.

When the build process begins, it first checks that another build process is not already in progress by checking for an active lock in the database. If there is no active lock, the build process sets an active lock, builds the sudoers file, and then releases the lock. The distribution process works in exactly the same way. If the TM Server were to crash before either process' lock was released, every subsequent process that checks for an existing running process would not create or distribute a new sudoers file, despite no other process actually existing on the system because the database lock was not released. If this situation occurs, run the following command on the TM Server:

```
echo ' UPDATE `Management`.`lock` SET `sudoers-build` = '0', `sudoers-distribution` = '0';' | mysql
-u root -p
Enter password: <YOUR MYSQL ROOT PASSWORD>
```

The lock should be released, and the Sudoers Build Process and CDSF Distribution Process should run autonomously again.

11.4 DSMS System Changelog Discovery

11.4.1.1 Viewing the Changelog (Web Panel)

TM Changelog can be viewed by navigating to the /changelog.cgi page directly, or through the menu at Home -> System Changelog.

11.4.1.2 Viewing the Changelog (Package)

TM Changelog is included with every package distributed. It is a text file and can be viewed with any text viewer. The Changelog file is called 'changelog'.

11.5 DSMS System Backups

The following table describes the items that should be backed up, and the recommended frequency of the backup.

Item	Backup Recommendation
Sudoers_Owner_ID **For changing the ownership of the sudoers file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.** **Defaults:**	This database is critical to the system's function as it contains all management data, including the System Account details and the Audit Log, and should be backed up at least daily. It's also highly advisable to have this database write to a transaction log between backups.

```
my $Owner = 'root';
```

11.5.1.1 Sudoers_Group_ID

For changing the group ownership of the sudoers file after it's created by the DSMS build process, we need to specify a group. It is recommended to run the DSMS build process as a root cron job (as defined in Cron Configuration), but Apache will need to reach this file to display its live contents on the web panel. If you modified Apache based on the Apache Configuration section, the group ownership should usually be 'apache'. However, on some systems, Apache Server doesn't run as the 'apache' user, such as when it runs as 'httpd', so you must specify the appropriate group ownership for TM to read the sudoers file.

Defaults:

```
my $Group = 'apache';
```

11.5.1.2 DNS_Zone_Master_File

This is the zone master file use for defining zones. You should include the full path.

Defaults:

```
my $DNS_Zone_Master_File = '/etc/bind/named.conf.local';
```

11.5.1.3 DNS_Internal_Location

This is the path that the system writes the temporary Internal DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_Internal_Location =
'/etc/bind/master-internal';
```

11.5.1.4   DNS_External_Location

This is the path that the system writes the temporary External DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_External_Location
= '/etc/bind/master-external';
```

11.5.1.5   DNS_Internal_SOA

This is the SOA data that's written at the top of the Internal DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email =
'postmaster@example.com';
        my $TTL = '86400';
                # 1 day
        my $Serial = `date
+%s`;   # Epoch
                $Serial =~
s/\n//;
        my $Refresh =
'10800';                # 3
hours
        my $Retry = '3600';
                # 1 hour
        my $Expire =
'2419200';       # 4 weeks
        my $Minimum =
'86400';        # 1 day
        my $NS1 =
'ns1.example.com';
        my $NS2 =
'ns2.example.com';
```

```
        my $NS3 =
'ns3.example.com';
```

11.5.1.6   DNS_External_SOA

This is the SOA data that's written at the top of the External DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email =
'postmaster@example.com';
        my $TTL = '86400';
                # 1 day
        my $Serial = `date
+%s`;   # Epoch
                $Serial =~
s/\n//;
        my $Refresh =
'10800';                    # 3
hours
        my $Retry = '3600';
                # 1 hour
        my $Expire =
'2419200';       # 4 weeks
        my $Minimum =
'86400';         # 1 day
        my $NS1 =
'ns1.example.com';
        my $NS2 =
'ns2.example.com';
        my $NS3 =
'ns3.example.com';
```

11.5.1.7   DNS_Storage

This is the directory where replaced DNS files are stored. You do not need a trailing slash.

Defaults:

```
my $DNS_Storage =
'../Storage/DNS';
```

11.5.1.8   DNS_Owner_ID

For changing the ownership of the DNS file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

11.5.1.9 DNS_Group_ID

For changing the group ownership of the DNS file after it's created, we need to specify a group owner. It is recommended to set this to be group owned by BIND, so the group name is usually 'bind'.

Defaults:

```
my $Group = 'bind';
```

11.5.1.10 Reverse_Proxy_Location

This is the path that the system writes the temporary reverse proxy files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the reverse proxy config. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Location
=
'../Storage/tmp/ReverseProxy';
```

11.5.1.11 Proxy_Redirect_Location

This is the path that the system writes the temporary proxy redirect files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the proxy redirect config. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Location
= '../Storage/tmp/
ReverseProxy';
```

11.5.1.12 Reverse_Proxy_Storage

This is the directory where replaced reverse proxy files

are stored. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Storage
= '../Storage/ReverseProxy';
```

11.5.1.13 Proxy_Redirect_Storage

This is the directory where replaced proxy redirect files are stored. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Storage
= '../Storage/ReverseProxy';
```

11.5.1.14 DShell_Job_Log_Location

This is the directory where job logs are stored. You do not need a trailing slash.

Defaults:

```
my
$DShell_Job_Log_Location =
'../Storage/D-Shell/Job-Log';
```

11.5.1.15 DShell_tmp_Location

This is the directory where D-Shell temporary files are are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_tmp_Location =
'../Storage/D-Shell/tmp';
```

11.5.1.16 DShell_WaitFor_Timeout

The default time that a *WAITFOR statement will wait before bailing out. Can be overridden manually by issuing *WAITFORnn, where nn is the timeout in seconds. nn can be any number.

Defaults:

```
my $DShell_WaitFor_Timeout
= 1800; # 30 minutes
```

DB_Management Database	
DB_Sudoers Database	This database is critical to the system's function as it contains all of the data relevant to the building of a complete sudoers file, and should be backed up at least daily. It's also highly advisable to have this database write to a transaction log between backups.
Common Parameter Configuration	The data in this file should remain relatively static. The system does not change this file, so it does not require ongoing backup, but should be backed up each time an administrator changes its contents, including after the system is first deployed.
Environmental Defaults File	The data in this file should remain relatively static. The system does not change this file, so it does not require ongoing backup, but should be backed up each time an administrator changes its contents, including after the system is first deployed.
Remaining Files	The remaining files do not require backing up. They should not be modified in any way and any restore can be done directly from the .tar.gz file where the checksum process can also be used to ensure file integrity.

When restoring the Sudoers_Owner_ID

For changing the ownership of the sudoers file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

11.5.1.17 Sudoers_Group_ID

For changing the group ownership of the sudoers file after it's created by the DSMS build process, we need to specify a group. It is recommended to run the DSMS build process as a root cron job (as defined in Cron Configuration), but Apache will need to reach this file to display its live contents on the web panel. If you modified Apache based on the Apache Configuration section, the group ownership should usually be 'apache'. However, on some systems, Apache Server doesn't run as the 'apache' user, such as when it runs as 'httpd', so you must specify the appropriate group ownership for TM to read the sudoers file.

Defaults:

```
my $Group = 'apache';
```

11.5.1.18 DNS_Zone_Master_File

This is the zone master file use for defining zones. You should include the full path.

Defaults:

```
my $DNS_Zone_Master_File = '/etc/bind/named.conf.local';
```

11.5.1.19 DNS_Internal_Location

This is the path that the system writes the temporary Internal DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_Internal_Location = '/etc/bind/master-internal';
```

11.5.1.20 DNS_External_Location

This is the path that the system writes the temporary External DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_External_Location = '/etc/bind/master-external';
```

11.5.1.21 DNS_Internal_SOA

This is the SOA data that's written at the top of the Internal DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email = 'postmaster@example.com';
        my $TTL = '86400';              # 1 day
        my $Serial = `date +%s`;       # Epoch
              $Serial =~ s/\n//;
        my $Refresh = '10800';         # 3 hours
        my $Retry = '3600';            # 1 hour
        my $Expire = '2419200';        # 4 weeks
        my $Minimum = '86400';         # 1 day
        my $NS1 = 'ns1.example.com';
        my $NS2 = 'ns2.example.com';
        my $NS3 = 'ns3.example.com';
```

11.5.1.22 DNS_External_SOA

This is the SOA data that's written at the top of the External DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email = 'postmaster@example.com';
        my $TTL = '86400';              # 1 day
        my $Serial = `date +%s`;       # Epoch
              $Serial =~ s/\n//;
        my $Refresh = '10800';         # 3 hours
        my $Retry = '3600';            # 1 hour
        my $Expire = '2419200';        # 4 weeks
        my $Minimum = '86400';         # 1 day
        my $NS1 = 'ns1.example.com';
        my $NS2 = 'ns2.example.com';
        my $NS3 = 'ns3.example.com';
```

11.5.1.23 DNS_Storage

This is the directory where replaced DNS files are stored. You do not need a trailing slash.

Defaults:

```
my $DNS_Storage = '../Storage/DNS';
```

11.5.1.24 DNS_Owner_ID

For changing the ownership of the DNS file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

11.5.1.25 DNS_Group_ID

For changing the group ownership of the DNS file after it's created, we need to specify a group owner. It is recommended to set this to be group owned by BIND, so the group name is usually 'bind'.

Defaults:

```
my $Group = 'bind';
```

11.5.1.26 Reverse_Proxy_Location

This is the path that the system writes the temporary reverse proxy files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the reverse proxy config. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Location = '../Storage/tmp/ReverseProxy';
```

11.5.1.27 Proxy_Redirect_Location

This is the path that the system writes the temporary proxy redirect files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the proxy redirect config. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Location = '../Storage/tmp/ ReverseProxy';
```

11.5.1.28 Reverse_Proxy_Storage

This is the directory where replaced reverse proxy files are stored. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Storage = '../Storage/ReverseProxy';
```

11.5.1.29 Proxy_Redirect_Storage

This is the directory where replaced proxy redirect files are stored. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Storage = '../Storage/ReverseProxy';
```

11.5.1.30 DShell_Job_Log_Location

This is the directory where job logs are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_Job_Log_Location = '../Storage/D-Shell/Job-Log';
```

11.5.1.31 DShell_tmp_Location

This is the directory where D-Shell temporary files are are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_tmp_Location = '../Storage/D-Shell/tmp';
```

11.5.1.32 DShell_WaitFor_Timeout

The default time that a *WAITFOR statement will wait before bailing out. Can be overridden manually by issuing *WAITFORnn, where nn is the timeout in seconds. nn can be any number.

Defaults:

```
my $DShell_WaitFor_Timeout = 1800; # 30 minutes
```

DB_Management database, be mindful that the sudoers build process lock may be set. This can be manually resolved by following the instructions in the Recovering from a Crashed Build and Distribution Process section.

11.6 High Availability and Load Balancing

TM is capable of a highly available configuration in Master/Slave, Master/Master or clustered setup, and it is relatively simple to achieve this. The following points should be considered when pairing or clustering the DSMS System.

11.6.1 NTP Configuration

Network Time Protocol must be configured at working on all servers in the cluster.

11.6.2 MySQL Configuration

11.6.2.1 If using a Master/Master or clustered setup, the auto-increment value should be offset for each server by the number of systems in your cluster to prevent duplicate data identities. You are also advised to setup and use a separate replication user. Both the Sudoers_Owner_ID

For changing the ownership of the sudoers file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

11.6.2.2 Sudoers_Group_ID

For changing the group ownership of the sudoers file after it's created by the DSMS build process, we need to specify a group. It is recommended to run the DSMS build process as a root cron job (as defined in Cron Configuration), but Apache will need to reach this file to display its live contents on the web panel. If you modified Apache based on the Apache Configuration section, the group ownership should usually be 'apache'. However, on some systems, Apache Server doesn't run as the 'apache' user, such as when it runs as 'httpd', so you must specify the appropriate group ownership for TM to read the sudoers file.

Defaults:

```
my $Group = 'apache';
```

11.6.2.3 DNS_Zone_Master_File

This is the zone master file use for defining zones. You should include the full path.

Defaults:

```
my $DNS_Zone_Master_File = '/etc/bind/named.conf.local';
```

11.6.2.4 DNS_Internal_Location

This is the path that the system writes the temporary Internal DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_Internal_Location = '/etc/bind/master-internal';
```

11.6.2.5 DNS_External_Location

This is the path that the system writes the temporary External DNS files to, before it is picked up by cron. If this server is the master DNS server, this path could be the path to the DNS config.

Defaults:

```
my $DNS_External_Location = '/etc/bind/master-external';
```

11.6.2.6 DNS_Internal_SOA

This is the SOA data that's written at the top of the Internal DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email = 'postmaster@example.com';
        my $TTL = '86400';              # 1 day
        my $Serial = `date +%s`;        # Epoch
              $Serial =~ s/\n//;
        my $Refresh = '10800';          # 3 hours
        my $Retry = '3600';             # 1 hour
        my $Expire = '2419200';         # 4 weeks
        my $Minimum = '86400';          # 1 day
        my $NS1 = 'ns1.example.com';
        my $NS2 = 'ns2.example.com';
        my $NS3 = 'ns3.example.com';
```

11.6.2.7 DNS_External_SOA

This is the SOA data that's written at the top of the External DNS file. It's recommended to keep the serial as default.

Defaults:

```
        my $Email = 'postmaster@example.com';
        my $TTL = '86400';              # 1 day
        my $Serial = `date +%s`;        # Epoch
```

```
        $Serial =~ s/\n//;
my $Refresh = '10800';          # 3 hours
my $Retry = '3600';             # 1 hour
my $Expire = '2419200';         # 4 weeks
my $Minimum = '86400';          # 1 day
my $NS1 = 'ns1.example.com';
my $NS2 = 'ns2.example.com';
my $NS3 = 'ns3.example.com';
```

11.6.2.8 DNS_Storage

This is the directory where replaced DNS files are stored. You do not need a trailing slash.

Defaults:

```
my $DNS_Storage = '../Storage/DNS';
```

11.6.2.9 DNS_Owner_ID

For changing the ownership of the DNS file after it's created, we need to specify an owner. It is recommended to keep this as the default, which is 'root'.

Defaults:

```
my $Owner = 'root';
```

11.6.2.10 DNS_Group_ID

For changing the group ownership of the DNS file after it's created, we need to specify a group owner. It is recommended to set this to be group owned by BIND, so the group name is usually 'bind'.

Defaults:

```
my $Group = 'bind';
```

11.6.2.11 Reverse_Proxy_Location

This is the path that the system writes the temporary reverse proxy files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the reverse proxy config. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Location = '../Storage/tmp/ReverseProxy';
```

11.6.2.12 Proxy_Redirect_Location

This is the path that the system writes the temporary proxy redirect files to, before it is picked up by cron. If this server is the master reverse proxy server, this path could be the path to the proxy redirect config. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Location = '../Storage/tmp/ ReverseProxy';
```

11.6.2.13 Reverse_Proxy_Storage

This is the directory where replaced reverse proxy files are stored. You do not need a trailing slash.

Defaults:

```
my $Reverse_Proxy_Storage = '../Storage/ReverseProxy';
```

11.6.2.14 Proxy_Redirect_Storage

This is the directory where replaced proxy redirect files are stored. You do not need a trailing slash.

Defaults:

```
my $Proxy_Redirect_Storage = '../Storage/ReverseProxy';
```

11.6.2.15 DShell_Job_Log_Location

This is the directory where job logs are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_Job_Log_Location = '../Storage/D-Shell/Job-Log';
```

11.6.2.16 DShell_tmp_Location

This is the directory where D-Shell temporary files are are stored. You do not need a trailing slash.

Defaults:

```
my $DShell_tmp_Location = '../Storage/D-Shell/tmp';
```

11.6.2.17 DShell_WaitFor_Timeout

The default time that a *WAITFOR statement will wait before bailing out. Can be overridden manually by issuing *WAITFORnn, where nn is the timeout in seconds. nn can be any number.

Defaults:

```
my $DShell_WaitFor_Timeout = 1800; # 30 minutes
```

DB_Management and DB_Sudoers databases should be replicated - only replicating one will cause problems. The configuration of this is straight-forward, but is out of the scope of this document.

11.6.3 Cron Configuration

The sudoers build and distribution processes are initiated using cron. The build process is locked in the database to prevent two build processes running at the same time. However, a race condition is possible where the build process may begin before the database lock data has been replicated to all systems. If this happens, two or more servers will build the CDSF and distribute them together. Due to the way the SFTP transfer occurs, and due to the final syntax validation performed on the Remote Server, the two build process should not conflict with each other and cause an incorrectly written final sudoers file on the Remote Server. However, it is still advised to have only one TM Server perform the build and distribution of the CDSF, not least for tracking network errors.

11.6.4 Public Key Configuration

Each TM Server's public key must be in each Remote Server's *authorized_keys* file for the transport user, or all TM Servers must share the same private key, for the distribution system to function correctly in a HA setup.

11.6.5 Load Balancer Configuration

TM supports Active/Passive and Active/Active load balancing however, due to the way TM handles CGI session data locally, you should use only load balancers which support sticky sessions (session

affinity) for handling web traffic. If there is sufficient interest, session handling can be moved into MySQL - please use the System Feedback section and file a request.

11.7 Note System Indexing

The following table is for note system fault diagnostic reference only:

Note Code	Type
00	Default database catchall code. Any note entries with a type ID of 00 represents a fault in identifying or recording the item type.
01	Hosts.
02	Host Groups.
03	Users.
04	User Groups.
05	Commands.
06	Command Groups.
07	Rules.

11.8 System Feedback

11.8.1 Reporting Faults

Please first report system faults to your local Service Desk as your system may require some modification to work with The Machine. Refer them to the requirements in this document.

If the fault is with TM itself, please report this fault in as much detail as possible to Ben Schofield (ben@nwk1.com).

11.8.2 Requesting Features

Please send all requests, with as much detail as possible, to Ben Schofield (ben@nwk1.com).