
DRLM Documentation

Release 2.1.2

Brain Updaters, S.L.L.

Oct 25, 2018

| | | |
|----------|--|-----------|
| 1 | License | 3 |
| 2 | Contributing | 5 |
| 3 | Contents: | 7 |
| 3.1 | DRLM Quick Start Guide | 7 |
| 3.2 | DRLM Installation | 8 |
| 3.3 | DRLM Client Installation | 23 |
| 3.4 | DRLM Client Recover | 30 |
| 3.5 | Error Reporting Configuration | 35 |
| 3.6 | Network Operations | 40 |
| 3.7 | Client Operations | 43 |
| 3.8 | Backup Operations | 47 |
| 3.9 | Building GRUB2 for diferent platfoms | 53 |
| 3.10 | About DRLM Docs | 55 |
| 4 | Indexes and tables | 59 |

DRLM Docs contains comprehensive documentation on the DRLM (Disaster Recovery Linux Manager). This page describes documentation's licensing, editions, and versions, and describes how to contribute to the DRLM Docs.

For more information on DRLM, see [About DRLM Project](#)¹. To download DRLM, see the downloads page.

¹ <http://drlm.org>

CHAPTER 1

License

This documentation is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International](http://creativecommons.org/licenses/by-nc-sa/4.0/)² (i.e. “CC-BY-NC-SA”) license.

The DRLM Manual is copyright © 2017 Brain Updaters, S.L.L.

² <http://creativecommons.org/licenses/by-nc-sa/4.0/>

CHAPTER 2

Contributing

Please, we encourage you to help us to improve this documentation.

To contribute to documentation the Github interface enables users to report errata or missing sections, discuss improvements and new sections through the issue-tracker at: [DRLM Docs GitHub Issue Tracker](https://github.com/brainupdaters/drlm-docs/issues)³.

Note: This documentation is under constant development. Please be patient. . .

³ <https://github.com/brainupdaters/drlm-docs/issues>

3.1 DRLM Quick Start Guide

3.1.1 DRLM Installation

Follow the steps at [DRLM Installation](#)⁴. (Select your OS)

3.1.2 Add Network to DRLM Server

First of all we must add the network where the ReaR clients are. To do this we have to use the command “drlm addnetwork” with the parameters -i “Network IP” network”, -g “Gateway IP”, -s “Server IP of the network”, -n “Network Name” and -m “Netmask”.

```
$ drlm -vD addnetwork -i 192.168.1.0 -g 192.168.1.1 -s 192.168.1.38 -n BuLan -m 255.  
↪255.255.0
```

3.1.3 Add Client to DRLM Server

Now we can add a ReaR client with the command “drlm addclient” and the parameters -n “Network Name”, -i “ReaR client IP”, -M “ReaR client MAC address” and -c “ReaR client hostname”.

```
$ drlm -vD addclient -n BuLan -i 192.168.1.45 -M 00:13:20:fe:48:16 -c ReaRcli1
```

3.1.4 ReaR Client Installation

Follow the steps at [ReaR Client Installation](#)⁵. (Select your OS)

⁴ <http://docs.drlm.org/en/2.1.2/Install.html#drlm-installation>

⁵ <http://docs.drlm.org/en/2.1.2/ClientConfig.html#rear-client-installation>

3.1.5 Run Client Backup

We are ready to take OS backups!!! At this point we have the DRLM server and ReaR client configured, you just have to run the command “drlm runbackup” with the parameter -c “ReaR client host name”

```
$ drlm -vD runbackup -c ReaRcli1
```

3.1.6 Restore Client Backup

Follow the steps at [DRLM Client Recover](#)⁶.

3.2 DRLM Installation

The purpose of this manual is explain, step by step, the installation and configuration of DRLM. At the end of this guide you should have a fully functional DRLM server.

3.2.1 Debian 8 & Ubuntu 16.04 LTS

Note: On the following steps, is assumed you have a minimal installation of Debian 8 or Ubuntu 16.04.

Install requirements

```
$ apt-get install openssh-client openssl gawk nfs-kernel-server rpcbind isc-dhcp-  
↪server tftpd-hpa apache2 qemu-utils sqlite3 lsb-release
```

Get DRLM

You can obtain the DRLM package building it from the source code or downloading from www.drlm.org website

Build DEB package from Source

```
$ apt-get install git build-essential debhelper  
$ git clone https://github.com/brainupdaters/drlm  
$ cd drlm  
$ make deb
```

Download DEB package From DRLM Web

```
$ wget http://www.drlm.org/downloads/drlm_2.1.2_all.deb
```

Install DRLM package

The DEB package can be installed as follows (on Debian, Ubuntu)

Execute the next command:

⁶ <http://drlm-docs.readthedocs.org/en/2.1.2/Restore.html>

```
$ dpkg -i drlm_2.1.2_all.deb
```

DRLM Components Configuration

This section covers configuration of:

- GRUB
- TFTP Service
- NFS Service
- DHCP Service
- HTTP Service

Configuring loop limits

The default configuration allows up to eight active loop devices. If more than eight file-based guests or loop devices are needed the number of loop devices configured can be adjusted adding the parameter `max_loop=1024` in the `/etc/default/grub` file as follows:

```
...  
GRUB_CMDLINE_LINUX="quiet max_loop=1024" ##UPDATE THIS LINE  
...
```

```
$ grub-mkconfig -o /boot/grub/grub.cfg
```

TFTP

You have to update the destination folder in the `/etc/default/tftpd-hpa` configuration file as follows

```
# /etc/default/tftpd-hpa  
TFTP_USERNAME="tftp"  
TFTP_DIRECTORY="/var/lib/drlm/store"  
TFTP_ADDRESS="0.0.0.0:69"  
TFTP_OPTIONS="--secure"
```

NFS

We don't have to configure the `/etc/exports` file, the file is automatically maintained by DRLM.

DHCP

Same as `/etc/exports` file, configuration of `/etc/dhcp/dhcpd.conf` file is not required, the file is automatically maintained by DRLM.

HTTP

```
$ a2enmod ssl
$ a2enmod rewrite
$ a2enmod cgi
```

Edit `/etc/apache2/apache2.conf` file

```
# Include the DRLM Configuration:
Include /usr/share/drlm/conf/HTTP/https.conf
```

```
$ rm /etc/apache2/sites-enabled/*
```

Edit `/etc/apache2/ports.conf` file

```
#Listen 80
```

Restart & check services

```
$ systemctl restart tftpd-hpa.service
$ systemctl status tftpd-hpa.service

$ systemctl restart rpcbind.service
$ systemctl status rpcbind.service

$ systemctl restart apache2.service
$ systemctl status apache2.service
```

Note: DHCP and NFS servers are not running because there is no config yet! no worries they will be reloaded automatically after first DRLM client will be added.

3.2.2 Debian 7 & Ubuntu 14.04 LTS

Note: On the following steps, is assumed you have a minimal installation of Debian 7 or Ubuntu 14.04.

Install requirements

```
$ apt-get install openssh-client openssl wget gzip tar gawk sed grep coreutils util-
↳linux nfs-kernel-server rpcbind isc-dhcp-server tftpd-hpa apache2 qemu-utils
↳sqlite3 lsb-release
```

Get DRLM

You can obtain the DRLM package building it from the source code or downloading from www.drlm.org website

Build DEB package from Source

```
$ apt-get install git build-essential debhelper
$ git clone https://github.com/brainupdaters/drlm
$ cd drlm
$ make deb
```

Download DEB package From DRLM Web

```
$ wget http://www.drlm.org/downloads/drlm_2.1.2_all.deb
```

Install DRLM package

The DEB package can be installed as follows (on Debian, Ubuntu)

Execute the next command:

```
$ dpkg -i drlm_2.1.2_all.deb
```

DRLM Components Configuration

This section covers configuration of:

- GRUB
- TFTP Service
- NFS Service
- DHCP Service
- HTTP Service

Configuring loop limits

The default configuration allows up to eight active loop devices. If more than eight file-based guests or loop devices are needed the number of loop devices configured can be adjusted adding the parameter `max_loop=1024` in the `/etc/default/grub` file as follows:

```
...
GRUB_CMDLINE_LINUX="quiet max_loop=1024" ##UPDATE THIS LINE
...
```

```
$ grub-mkconfig -o /boot/grub/grub.cfg
```

TFTP

You have to update the destination folder in the `/etc/default/tftpd-hpa` configuration file as follows

```
# /etc/default/tftpd-hpa
TFTP_USERNAME="tftp"
TFTP_DIRECTORY="/var/lib/drlm/store"
TFTP_ADDRESS="0.0.0.0:69"
TFTP_OPTIONS="--secure"
```

NFS

We don't have to configure the `/etc/exports` file, the file is automatically maintained by DRLM.

DHCP

Same as `/etc/exports` file, configuration of `/etc/dhcp/dhcpd.conf` file is not required, the file is automatically maintained by DRLM.

HTTP

```
$ a2enmod ssl
$ a2enmod rewrite
$ a2enmod cgi
```

Edit `/etc/apache2/apache2.conf` file

```
# Include the DRLM Configuration:
Include /usr/share/drlm/conf/HTTP/https.conf
```

```
$ rm /etc/apache2/sites-enabled/*
```

Edit `/etc/apache2/ports.conf` file

```
#NameVirtualHost *:80
#Listen 80
```

Restart & check services

```
$ service tftpd-hpa restart
$ service tftpd-hpa status
in.tftpd is running.
$ service rpcbind restart
$ service rpcbind status
rpcbind is running.
$ service apache2 restart
$ service apache2 status
Apache2 is running (pid 2023).
```

Note: DHCP and NFS servers are not running because there is no config yet! no worries they will be reloaded automatically after first DRLM client will be added.

3.2.3 CentOS 7 & RHEL 7

Note: On the following steps, is assumed you have a minimal installation of CentOS or RHEL 7.

Warning: SELinux has been disabled

```
$ cat /etc/sysconfig/selinux

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

```
$ setenforce 0
```

Note: It is not a requirement to disable SELinux, but to work with DRLM Server must be properly configured. We have disabled this feature for easier installation.

Install requirements

```
$ yum -y install openssh-clients openssl wget gzip tar gawk sed grep coreutils util-
↳linux rpcbind dhcp tftp-server httpd xinetd nfs-utils nfs4-acl-tools mod_ssl qemu-
↳img sqlite redhat-lsb-core
```

Get DRLM

Build RPM package from Source

```
$ yum install git rpm-build
$ git clone https://github.com/brainupdaters/drlm
$ cd drlm
$ make rpm
```

Download RPM package From DRLM Web

```
$ wget http://www.drlm.org/downloads/drlm-2.1.2-1git.el6.noarch.rpm
```

Install DRLM package

The RPM package can be installed as follows (on Redhat, CentOS)

Execute the next command:

```
$ rpm -ivh drlm-2.1.2-1git.el7.centos.noarch.rpm
```

DRLM Components Configuration

This section covers configuration of:

- GRUB
- TFTP Service
- NFS Service
- DHCP Service
- HTTP Service

Configuring loop limits

The default configuration allows up to eight active loop devices. If more than eight file-based guests or loop devices are needed the number of loop devices configured can be adjusted adding the parameter `max_loop=1024` in the `/etc/default/grub` file as follows:

```
...  
GRUB_CMDLINE_LINUX="..... max_loop=1024" ##UPDATE THIS LINE ADDING MAX_LOOP=1024_  
↪PARAMETER  
...
```

```
$ grub2-mkconfig -o /boot/grub2/grub.cfg
```

TFTP

You have to update the `/etc/xinetd.d/tftp` configuration file as follows:

```
service tftp  
{  
    socket_type = dgram  
    protocol = udp  
    wait = yes  
    user = root  
    server = /usr/sbin/in.tftpd  
    server_args = -s /var/lib/drlm/store  
    disable = no  
    per_source = 11  
    cps = 100 2  
    flags = IPv4  
}
```

NFS

We don't have to configure the `/etc/exports` file, the file is automatically maintained by DRLM.

DHCP

Same as `/etc/exports` file, configuration of `/etc/dhcp/dhcpd.conf` file is not required, the file is automatically maintained by DRLM.

HTTP

Disable the default Virtual Host and configure the server to work with SSL.

We have to edit de `/etc/httpd/conf.d/ssl.conf`, comment or delete the Virtual host and include the DRLM http default configuration at the end of it.

```
Coment from here --->
##
## SSL Virtual Host Context
##
```

At the end of the file **and** insert:

```
# Include the DRLM Configuration:
Include /usr/share/drlm/conf/HTTP/https.conf
```

Then we have to coment the 80 port service commenting or deleting the next lines in `/etc/httpd/conf/httpd.conf` file.

```
#Listen 80

#ServerAdmin root@localhost

#DocumentRoot "/var/www/html"

#<Directory />
#   Options FollowSymLinks
#   AllowOverride None
#</Directory>

#<Directory "/var/www/html">
#   Options Indexes FollowSymLinks
#   AllowOverride None
#   Order allow,deny
#   Allow from all
#</Directory>

#ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"

#<Directory "/var/www/cgi-bin">
#   AllowOverride None
#   Options None
#   Order allow,deny
#   Allow from all
#</Directory>
```

To finish we have to comment the `ErrorLog` and `CustomLog` lines in `/usr/share/drlm/conf/HTTP/https.conf` file.

```
#   ErrorLog ${APACHE_LOG_DIR}/error.log
#   CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined
```

Restart & check services

```
$ systemctl enable xinetd.service
$ systemctl restart xinetd.service

$ systemctl enable rpcbind.service
$ systemctl restart rpcbind.service

$ systemctl enable httpd.service
$ systemctl restart httpd.service
```

Note: DHCP and NFS servers are not running because there is no config yet! no worries they will be reloaded automatically after first DRLM client will be added.

3.2.4 CentOS 6 & RHEL 6

Note: On the following steps, is assumed you have a minimal installation of CentOS or RHEL 6.

Warning: Iptables and SELinux has been disabled

```
$ cat /etc/sysconfig/selinux

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

```
$ setenforce 0
```

Note: It is not a requirement to disable SELinux and Iptables, but to work with DRLM Server must be properly configured. We have disabled these features for easier installation.

Iptables

```
$ chkconfig iptables off
$ service iptables stop
```

Install requirements

```
$ yum -y install openssh-clients openssl wget gzip tar gawk sed grep coreutils util-
↳linux rpcbind dhcp tftp-server httpd xinetd nfs-utils nfs4-acl-tools mod_ssl qemu-
↳img sqlite redhat-lsb-core
```

Get DRLM

Build RPM package from Source

```
$ yum install git rpm-build
$ git clone https://github.com/brainupdaters/drlm
$ cd drlm
$ make rpm
```

Download RPM package From DRLM Web

```
$ wget http://www.drlm.org/downloads/drlm-2.1.2-1git.el7.centos.noarch.rpm
```

Install DRLM package

The RPM package can be installed as follows (on RHEL, CentOS)

Execute the next command:

```
$ rpm -ivh drlm-2.1.2-1git.el6.noarch.rpm
```

DRLM Components Configuration

This section covers configuration of:

- GRUB
- TFTP Service
- NFS Service
- DHCP Service
- HTTP Service

Configuring loop limits

The default configuration allows up to eight active loop devices. If more than eight clients are needed, the number of loop devices configured can be adjusted adding the parameter *max_loop=1024* in the */etc/grub.conf* file as follows:

```
title Red Hat Enterprise Linux (2.6.32-358.el6.x86_64)
root (hd0,0)
kernel /vmlinuz-2.6.32-358.el6.x86_64 ro root=/dev/mapper/vgroot-lvroot rd_NO_LUKS_
↳LANG=en_US.UTF-8 KEYBOARDTYPE=pc KEYTABLE=es rd_NO_MD rd_LVM_LV=vgroot/lvswap_
↳SYSFONT=latacyrheb-sun16 crashkernel=auto rd_LVM_LV=vgroot/lvroot rd_NO_DM rhgb_
↳quiet max_loop=1024
initrd /initramfs-2.6.32-358.el6.x86_64.img
```

TFTP

You have to update the `/etc/xinetd.d/tftp` configuration file as follows:

```
service tftp
{
    socket_type = dgram
    protocol = udp
    wait = yes
    user = root
    server = /usr/sbin/in.tftpd
    server_args = -s /var/lib/drlm/store
    disable = no
    per_source = 11
    cps = 100 2
    flags = IPv4
}
```

NFS

We don't have to configure the `/etc/exports` file, the file is automatically maintained by DRLM.

DHCP

Same as `/etc/exports` file, configuration of `/etc/dhcp/dhcpd.conf` file is not required, the file is automatically maintained by DRLM.

HTTP

Disable the default Virtual Host and configure the server to work with SSL.

We have to edit the `/etc/httpd/conf.d/ssl.conf`, comment or delete the Virtual host and include the DRLM http default configuration at the end of it.

```
Coment from here --->
##
## SSL Virtual Host Context
##
```

At the end of the file **and** insert:

```
# Include the DRLM Configuration:
Include /usr/share/drlm/conf/HTTP/https.conf
```

Then we have to comment the 80 port service commenting or deleting the next lines in `/etc/httpd/conf/httpd.conf` file.

```
#Listen 80

#ServerAdmin root@localhost

#DocumentRoot "/var/www/html"

#<Directory />
```

(continues on next page)

(continued from previous page)

```
# Options FollowSymLinks
# AllowOverride None
#</Directory>

#<Directory "/var/www/html">
# Options Indexes FollowSymLinks
# AllowOverride None
# Order allow,deny
# Allow from all
#</Directory>

#ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"

#<Directory "/var/www/cgi-bin">
# AllowOverride None
# Options None
# Order allow,deny
# Allow from all
#</Directory>
```

To finish we have to comment the ErrorLog and CustomLog lines in `/usr/share/drlm/conf/HTTP/https.conf` file.

```
# ErrorLog ${APACHE_LOG_DIR}/error.log
# CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined
```

Restart & check services

```
$ service xinetd restart
$ service xinetd status
xinetd (pid 5307) is running...
$ service rpcbind restart
$ service rpcbind status
rpcbind (pid 5097) is running...
$ service httpd restart
$ service httpd status
httpd (pid 5413) is running...
```

Note: DHCP and NFS servers are not running because there is no config yet! no worries they will be reloaded automatically after first DRLM client will be added.

3.2.5 SLES 12 SP1

Note: On the following steps, is assumed you have a minimal SLES 12 SP1

Install requirements

```
$ zypper in openssl wget gzip tar gawk sed grep coreutils util-linux nfs-kernel-  
→server rpcbind dhcp-server sqlite3 apache2 openssl qemu-tools tftp xinetd lsb-  
→release
```

Get DRLM

You can obtain the DRLM package building it from the source code or downloading from www.drlm.org website

Build RPM package from Source

```
$ zypper install git-core rpm-build  
$ git clone https://github.com/brainupdaters/drlm  
$ cd drlm  
$ make rpm
```

You can obtain the RPM DRLM package from www.drlm.org website

Download RPM package From DRLM Web

```
$ wget http://www.drlm.org/downloads/drlm-2.1.2-1git.noarch.rpm
```

Install DRLM package

The RPM package can be installed as follows (on SLES 12 SP1)

Execute the next command:

```
$ zypper in drlm-2.1.2-1git.noarch.rpm
```

DRLM Components Configuration

This section covers configuration of:

- GRUB
- TFTP Service
- NFS Service
- DHCP Service
- HTTP Service

Configuring loop limits

The default configuration allows up to eight active loop devices. If more than eight file-based guests or loop devices are needed the number of loop devices configured can be adjusted adding the parameter `max_loop=1024` in the `/etc/default/grub` file as follows:


```
...
GRUB_CMDLINE_LINUX="quiet max_loop=1024" ##UPDATE THIS LINE
...
```

```
$ grub2-mkconfig -o /boot/grub2/grub.cfg
```

TFTP

You have to update the `/etc/xinetd.d/tftp` configuration file as follows:

```
service tftp
{
    socket_type = dgram
    protocol = udp
    wait = yes
    user = root
    server = /usr/sbin/in.tftpd
    server_args = -s /var/lib/drlm/store
    disable = no
    per_source = 11
    cps = 100 2
    flags = IPv4
}
```

NFS

We don't have to configure the `/etc/exports` file, the file is automatically maintained by DRLM.

DHCP

Same as `/etc/exports` file, configuration of `/etc/dhcpd.conf` file is not required, the file is automatically maintained by DRLM.

but you have to change the location of `/etc/dhcpd.conf`

Edit `/etc/drlm/local.conf`

```
DHCP_DIR="/etc"
DHCP_FILE="$DHCP_DIR/dhcpd.conf"
```

`DHCPD_INTERFACE` by default is set as `DHCPD_INTERFACE=""` and `dhcpd` does not start, change it to "ANY"

Edit `/etc/sysconfig/dhcpd`

```
DHCPD_INTERFACE="ANY"
```

HTTP

```
$ a2enmod ssl
$ a2enmod rewrite
$ a2enmod cgi
$ a2enmod mod_access_compat
```

Edit /etc/apache2/httpd.conf file

```
# Include the DRLM Configuration:
Include /usr/share/drlm/conf/HTTP/https.conf
```

Add APACHE_LOG_DIR variable to /etc/sysconfig/apache2

Edit /usr/share/drlm/conf/HTTP/https.conf

```
echo "APACHE_LOG_DIR=/var/log/apache2" >> /etc/sysconfig/apache2
```

Edit /etc/apache2/listen.conf file

```
#Listen 80
Listen 443

#Listen 80

<IfDefine SSL>
  <IfDefine !NOSSL>
    <IfModule mod_ssl.c>

      Listen 443

    </IfModule>
  </IfDefine>
</IfDefine>
```

Restart & check services

```
$ systemctl restart xinetd.service
$ systemctl status xinetd.service

$ systemctl restart rpcbind.service
$ systemctl status rpcbind.service

$ systemctl restart apache2.service
$ systemctl status apache2.service

$ systemctl enable nfs-server
$ systemctl start nfs-server
$ systemctl status nfs-server
```

Note: DHCP and NFS servers are not running because there is no config yet! no worries they will be reloaded automatically after first DRLM client will be added.

3.3 DRLM Client Installation

3.3.1 Unattended Installation

Now ReaR can be installed and configured on a remote server from the DRLM server using **drlm instclient**

Let's explain a little bit the steps this feature does:

- Create the drlm user
- Install ReaR dependencies
- Install ReaR package
- Configure ReaR to be managed by DRLM
- Configure SUDO for drlm user.
- Start and configure required services

Supported OSs for instclient command

Unattended Client Installation has been tested on:

- SLES (11 & 12)
- OpenSUSE (13 & Leap 42)
- RHEL & CentOS (5, 6 & 7)
- Debian (6, 7 & 8)
- Ubuntu LTS (12.04, 14.04 & 16.04)

Note: It should work on other RedHat, Debian or SUSE variants.

Requirements

In order to install ReaR from DRLM server the client must have:

- Access to EPEL Repo to install rear from repo (CentOS,RHEL)
- instclient uses apt-get, yum and zypper, so repositories must be configured
- SSH enabled
- root user or user with administrator privileges to install, start services like rpcbind and configure ReaR, DHCP and sudo applications.

Run unattended install

To perform an unattended install of a DRLM client, just is needed to run **instclient** DRLM command like one of the following examples:

Warning: The client must be properly registered in DRLM with **addclient** command.

Examples:

```
$ drlm instclient -c ReaRcli1
$ drlm instclient -c ReaRcli1 -U http://download.opensuse.org/repositories/Archiving:/
↳Backup:/Rear/Debian_7.0/all/rear_1.17.2_all.deb
```

Note: See Client Operations for more information

3.3.2 Manual Installation

Debian 7 & Ubuntu 12.04 or 14.04

ReaR requirements for DRLM

As rear is written in bash you need bash as a bare minimum. Other requirements are:

- syslinux (for i386 based systems)
- ethtool
- genisoimage
- parted
- gawk
- attr
- sudo
- curl (rear need to get its configuration from DRLM server)
- mingetty (rear is depending on it in recovery mode)

```
$ apt-get install syslinux ethtool genisoimage parted gawk attr sudo curl mingetty
```

Download and install ReaR

Note: Minimum version required of ReaR: 1.17.0

Download ReaR

```
$ wget http://download.opensuse.org/repositories/Archiving:/Backup:/Rear/Debian_7.0/
↳all/rear_1.17.2_all.deb
```

You can download other ReaR versions from [ReaR Download Page](#)⁷ or from [OpenSuse Build Service](#)⁸ .

Install ReaR package

The DEB based package can be installed as follows

Execute the next command:

⁷ <http://relax-and-recover.org/download/>

⁸ <https://build.opensuse.org/project/show/Archiving:Backup:Rear>

```
$ dpkg -i rear_1.17.2_all.deb
```

Note: For more information about ReaR visit: <http://relax-and-recover.org/documentation>

Create DRLM User

```
$ useradd -d /home/drlm -c "DRLM User Agent" -m -s /bin/bash -p $(echo S3cret |  
↪openssl passwd -1 -stdin) drlm
```

Disable password aging for drlm user

```
$ chage -I -1 -m 0 -M 99999 -E -1 drlm
```

Copy rsa key from DRLM Server to the new client

Warning: You have to execute this code from DRLM Server. The password which you will be asked for is “S3cret” and “client_ipaddr” must be changed to the client ip address.

```
$ ssh-keygen -t rsa  
$ ssh-copy-id drlm@"client_ipaddr"
```

Disable password login

```
$ passwd -l drlm
```

Add Sudo roles for DRLM user

Edit `/etc/sudoers.d/drlm` and add the following lines

```
Cmd_Alias DRLM = /usr/sbin/rear, /bin/mount, /sbin/vgs  
drlm ALL=(root) NOPASSWD: DRLM
```

Change `/etc/sudoers.d/drlm` permissions

```
$ chmod 440 /etc/sudoers.d/drlm
```

Client configuration

We have to specify that this ReaR client is managed from a DRLM server. We have to edit the `/etc/rear/local.conf` and insert the next line.

```
DRLM_MANAGED=y
```

Debian 6

ReaR requirements for DRLM

As rear is written in bash you need bash as a bare minimum. Other requirements are:

- syslinux (for i386 based systems)
- ethtool
- lsb-release
- genisoimage
- iproute
- iputils-ping
- binutils
- parted
- openssl
- gawk
- attr
- sudo
- openssh-server (to enable communications between DRLM and ReaR client)
- curl (rear need to get its configuration from DRLM server)
- mingetty (rear is depending on it in recovery mode)

Note: Debian 6 is discontinued, make sure that you have the Squeeze archive repository in `/etc/apt/sources.list` (deb <http://archive.debian.org/debian/squeeze contrib main non-free>)

```
$ apt-get update
$ apt-get install syslinux ethtool lsb-release genisoimage iproute iputils-ping_
↪binutils parted openssl gawk attr sudo openssh-server curl mingetty nfs-common
```

Download and install ReaR

Note: Minimum version required of ReaR: 1.17.0 (Recommended ReaR 1.18)

Download ReaR

amd64 architecture:

```
$ wget http://download.opensuse.org/repositories/Archiving:/Backup:/Rear/Debian_7.0/
↪amd64/rear_1.18_amd64.deb
```

i386 architecture:

```
$ wget http://download.opensuse.org/repositories/Archiving:/Backup:/Rear/Debian_7.0/
↳i386/rear_1.18_i386.deb
```

You can download other ReaR versions from [ReaR Download Page](#)⁹ or from [OpenSuse Build Service](#)¹⁰.

Install ReaR package

The DEB based package can be installed as follows

Execute the next command:

```
$ dpkg -i rear_1.18_amd64.deb
```

Note: Use “dpkg -i rear_1.18_i386.deb” to install i386 architecture. For more information about ReaR visit: <http://relax-and-recover.org/documentation>

Create DRLM User

```
$ useradd -d /home/drlm -c "DRLM User Agent" -m -s /bin/bash -p $(echo S3cret |
↳openssl passwd -1 -stdin) drlm
```

Disable password aging for drlm user

```
$ chage -I -1 -m 0 -M 99999 -E -1 drlm
```

Copy rsa key from DRLM Server to the new client

Warning: You have to execute this code from DRLM Server. The password which you will be asked for is “S3cret” and “client_ipaddr” must be changed to the client ip address.

```
$ ssh-keygen -t rsa
$ ssh-copy-id drlm@"client_ipaddr"
```

Disable password login

```
$ passwd -l drlm
```

Add Sudo roles for DRLM user

Edit `/etc/sudoers.d/drlm` and add the following lines

⁹ <http://relax-and-recover.org/download/>

¹⁰ <https://build.opensuse.org/project/show/Archiving:Backup:Rear>

```
Cmd_Alias DRLM = /usr/sbin/rear, /bin/mount, /sbin/vgs
drlm ALL=(root) NOPASSWD: DRLM
```

Change `/etc/sudoers.d/drlm` permissions

```
$ chmod 440 /etc/sudoers.d/drlm
```

Client configuration

We have to specify that this ReaR client is managed from a DRLM server. We have to edit the `/etc/rear/local.conf` and insert the next line.

```
DRLM_MANAGED=y
```

CentOS & RHEL 6

ReaR requirements for DRLM

As rear is written in bash you need bash as a bare minimum. Other requirements are:

- mkisofs
- mingetty (rear depends on it in recovery mode)
- syslinux (for i386 based systems)
- nfs-utils
- cifs-utils
- rpcbind
- wget
- sudo
- curl (rear needs it to get the configuration from DRLM server)

```
$ yum -y install mkisofs mingetty syslinux nfs-utils cifs-utils rpcbind wget curl sudo
```

Download and install ReaR

Note: Minimum version required of ReaR: 1.17.0

Download ReaR

```
$ DISTRO="CentOS_CentOS-6" or DISTRO="RedHat_RHEL-6"
$ wget http://download.opensuse.org/repositories/Archiving:/Backup:/Rear/$DISTRO/
↪ $(uname -m)/rear-1.17.2-1.el6.$(uname -m).rpm
```

You can download other ReaR versions from [ReaR Download Page](http://relax-and-recover.org/download/)¹¹ or from [OpenSuse Build Service](https://build.opensuse.org/project/show/Archiving:Backup:Rear)¹².

¹¹ <http://relax-and-recover.org/download/>

¹² <https://build.opensuse.org/project/show/Archiving:Backup:Rear>

Install ReaR package

The RPM based package can be installed as follows

Execute the next command:

```
$ yum install rear-1.17.2-1.el6.x86_64.rpm
```

Note: For more information about ReaR visit: <http://relax-and-recover.org/documentation>

Create DRLM User

```
$ useradd -d /home/drlm -c "DRLM User Agent" -m -s /bin/bash -p $(echo S3cret |  
→openssl passwd -1 -stdin) drlm
```

Disable password aging for drlm user

```
$ chage -I -1 -m 0 -M 99999 -E -1 drlm
```

Copy rsa key from DRLM Server to the new client

Warning: You have to execute this code from DRLM Server. The password which you will be asked for is “S3cret” and “client_ipaddr” must be changed to the client ip address.

```
$ ssh-keygen -t rsa  
$ ssh-copy-id drlm@"client_ipaddr"
```

Disable password login

```
$ passwd -l drlm
```

Add Sudo roles to DRLM user

Edit `/etc/sudoers.d/drlm` and add the following lines

```
Cmd_Alias DRLM = /usr/sbin/rear, /bin/mount, /sbin/vgs  
drlm    ALL=(root)    NOPASSWD: DRLM
```

Change `/etc/sudoers.d/drlm` permissions

```
$ chmod 440 /etc/sudoers.d/drlm
```

Client configuration

We have to specify that this ReaR client is managed from a DRLM server. We have to edit the `/etc/rear/local.conf` and insert the next line.

```
DRLM_MANAGED=y
```

Services

rpcbind

```
$ service rpcbind start
$ chkconfig rpcbind on
```

nfs

```
$ service nfs start
$ chkconfig nfs on
```

3.4 DRLM Client Recover

In this section we show how to recover a system which has been backed up.

In this example your client and server has the following configuration. You have to adapt it to your case.

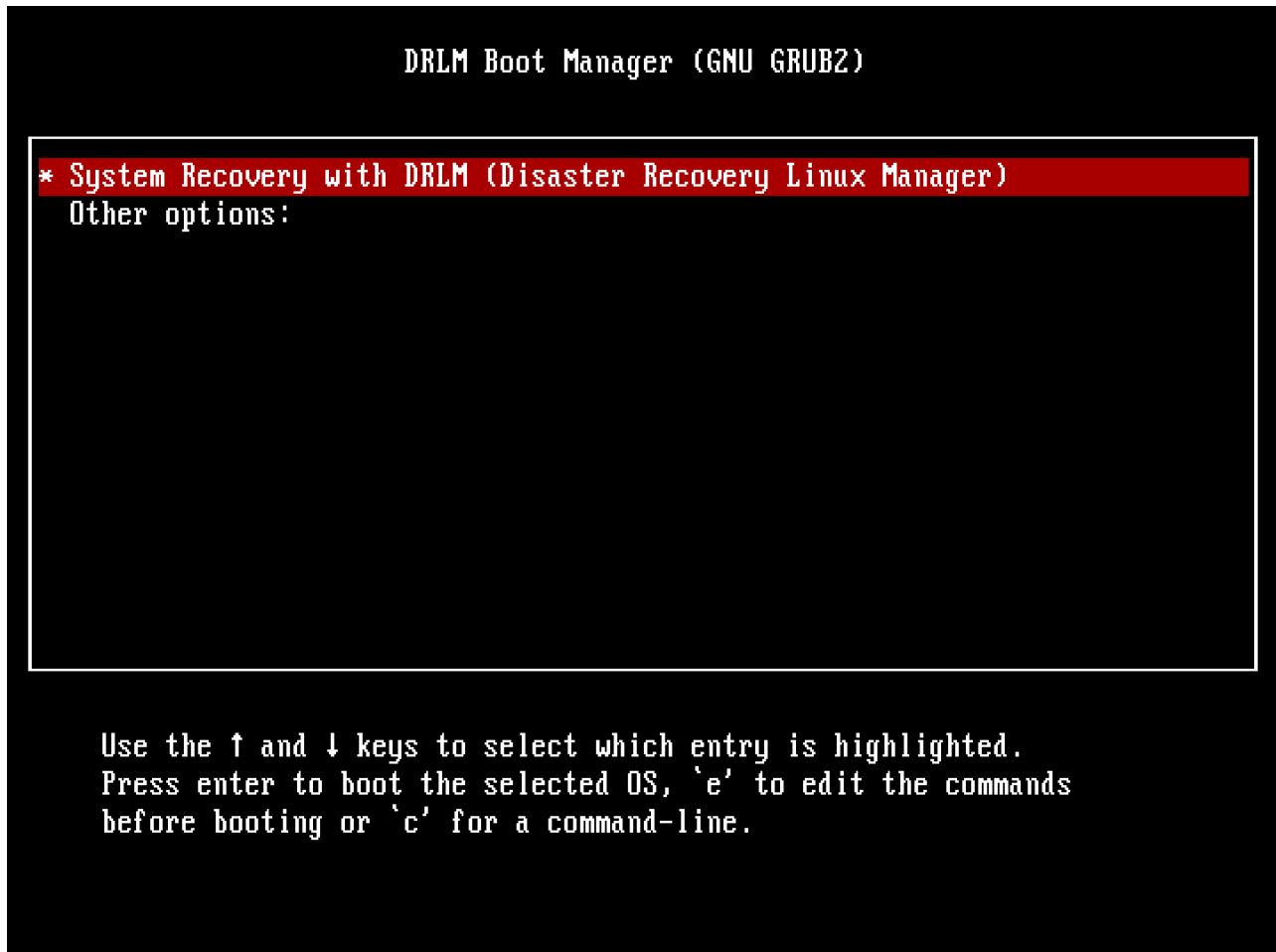
```
DRLM Server Host Name: DRLMsrv
DRLM Server IP: 192.168.2.120

ReaR Client Host Name: fosdemcli4
ReaR Client IP: 192.168.2.102
```

3.4.1 Step by Step Client Recover

Reboot the Client and select boot from network. Automatically will boot from PXE.

1. The DRLM server gives us through PXE/TFTP the client boot system. We just have to select first menu option to enter in the recovery system.



2. Once we have the system ready Login as “root”. No password required.

```
Running 42-engage-scsi.sh...
Running 45-serial-console.sh...
Running 55-migrate-network-devices.sh...
Running 58-start-dhclient.sh...
Attempting to start the DHCP client daemon
Running 60-network-devices.sh...
Running 62-routing.sh...
Running 99-makedev.sh...
* * * Rescue System is ready * * *
INIT: Entering runlevel: 3

Relax-and-Recover 1.17.2 / Git

Relax-and-Recover comes with ABSOLUTELY NO WARRANTY; for details see
the GNU General Public License at: http://www.gnu.org/licenses/gpl.html

Host fosdemcli4 using Backup NETFS and Output PXE
Build date: Sun, 31 Jan 2016 15:24:28 +0100

Debian GNU/Linux 7 fosdemcli4 tty1

fosdemcli4 login: root_
```

3. We indicate that we want to recover the system with the command “rear recover” and the following variables SERVER=”DRLM Server Ip” REST_OPTS=-k ID=”Rear Client Host Name”, in our case “rear recover SERVER=192.168.2.120 REST_OPTS=-k ID=fosdemcli4”

```
Running 60-network-devices.sh...
Running 62-routing.sh...
Running 99-makedev.sh...
* * * Rescue System is ready * * *
INIT: Entering runlevel: 3

Relax-and-Recover 1.17.2 / Git

Relax-and-Recover comes with ABSOLUTELY NO WARRANTY; for details see
the GNU General Public License at: http://www.gnu.org/licenses/gpl.html

Host fosdemcli4 using Backup METFS and Output PXE
Build date: Sun, 31 Jan 2016 15:24:28 +0100

Debian GNU/Linux 7 fosdemcli4 tty1

fosdemcli4 login: root

Welcome to Relax and Recover. Run "rear recover" to restore your system !

RESCUE fosdemcli4:~ # rear recover SERVER=192.168.2.120 REST_OPTS=-k ID=fosdemc
li4_
```

4. The system is recovering.

```
Debian GNU/Linux 7 fosdemcli4 tty1
fosdemcli4 login: root

Welcome to Relax and Recover. Run "rear recover" to restore your system !

RESCUE fosdemcli4:~ # rear recover SERVER=192.168.2.120 REST_OPTS=-k ID=fosdemcli4
Relax-and-Recover 1.17.2 / Git
Using log file: /var/log/rear/rear-fosdemcli4.log
Calculating backup archive size
Backup archive size is 332M      /tmp/rear.c7HvG81lh5Xu4EO/outputfs/BKP/backup.tar.gz (compressed)
Comparing disks.
Disk configuration is identical, proceeding with restore.
Start system layout restoration.
Creating partitions for disk /dev/sda (msdos)
Creating ext4-filesystem / on /dev/sda1
Mounting filesystem /
Creating swap on /dev/sda5
Disk layout created.
Decrypting disabled
Restoring from '/tmp/rear.c7HvG81lh5Xu4EO/outputfs/BKP/backup.tar.gz'
Restored 305 MiB [avg 20861 KiB/sec]_
```

5. System recovered! So we only have to restart the client.

```

RESCUE fosdemcli4:~ # rear recover SERVER=192.168.2.120 REST_OPTS=-k ID=fosdemcli4
Relax-and-Recover 1.17.2 / Git
Using log file: /var/log/rear/rear-fosdemcli4.log
Calculating backup archive size
Backup archive size is 332M      /tmp/rear.c7HvG81lh5Xu4EO/outputfs/BKP/backup.tar.gz (compressed)
Comparing disks.
Disk configuration is identical, proceeding with restore.
Start system layout restoration.
Creating partitions for disk /dev/sda (msdos)
Creating ext4-filesystem / on /dev/sda1
Mounting filesystem /
Creating swap on /dev/sda5
Disk layout created.
Decrypting disabled
Restoring from '/tmp/rear.c7HvG81lh5Xu4EO/outputfs/BKP/backup.tar.gz'
Restored 873 MiB [avg 21305 KiB/sec]OK
Restored 873 MiB in 43 seconds [avg 20810 KiB/sec]
Installing GRUB2 boot loader
Installation finished. No error reported.

Finished recovering your system. You can explore it under '/mnt/local'.

RESCUE fosdemcli4:~ # _

```

3.5 Error Reporting Configuration

DRLM can be configured to report errors on scheduled backups if required. It is possible to report by mail or integrating with your monitoring service. At this time (DRLM 2.0) we support error reporting by mail and integration with Nagios, Zabbix and HPOM(OVO) monitoring services.

Note: All reporting configuration samples are located in: /usr/share/drlm/conf/samples

3.5.1 Enable DRLM reporting

```

$ vi /usr/share/drlm/conf/default.conf

#####
#
# Defines HowTo report Errors using some known and wide used methods
#
#   ERR_REPORT=[yes|no]
#   default: no

```

(continues on next page)

(continued from previous page)

```
# REPORT_TYPE=[ovo|nsca-ng|nsca|zabbix|mail|...]
# default: empty
#
#####

ERR_REPORT=yes
REPORT_TYPE=<type>
```

3.5.2 Configure nsca-ng (Nagios based) reporting

In order to configure Nagios Error reporting on DRLM, the Nagios NSCA Client must be installed.

Note: We're using nsca-ng because nsca is deprecated, but if you have nsca DRLM supports it

Debian 7/8

```
$ apt-get install nsca-ng-client
```

RHEL/Centos 6/7

if nsca-ng-client is not in the repositories, it can be downloaded from:

- <https://www.nasca-ng.org/>

The following options are DRLM defaults, change any of them to your installation requirements in `/etc/drlm/local.conf`.

```
$ vi /etc/drlm/local.conf

#
# REPORT_TYPE=nagios
# NAGIOS VARIABLES
#
# These are default values and can be overwritten in local.conf according to your_
↳NAGIOS installation and configuration.
#

NAGCMD="/usr/sbin/send_nasca"
NAGSVC="DRLM"
NAGHOST="$HOSTNAME"
NAGCONF"/etc/drlm/alerts/nagios.cfg"
```

nagios_sample.cfg

Copy the sample DRLM configuration for Nagios to previously defined \$NAGCONF and adjust it to your environment needs.

```
#### DRLM (Disaster Recovery Linux Manager) Nagios error reporting sample_
↳configuration file.
#### Default: /etc/drlm/alerts/nagios.cfg

### identity = <string>
# Send the specified client identity to the server.
# By default, localhost will be used.
```

(continues on next page)

(continued from previous page)

```
identity = "< client identity >"

### server = <string>
# Connect and talk to the specified server address or hostname.
# The default server is "localhost".

server = "< nagios based server >"

### port = <string>
# Connect to the specified service name or port number on the
# server instead of using the default port (5668).

port = < nagios based listening port >
password = "change-me"
```

Note: The configuration on the server side is not in the scope of this documentation. Please check your Nagios service documentation to configure properly the NSCA service and how to report DRLM alerts.

For reference you can check:

- <https://www.nasca-ng.org/documentation/nsca-ng.pdf>
- <https://www.nasca-ng.org/documentation/nsca-ng.cfg.pdf>
- https://www.nasca-ng.org/documentation/send_nasca.pdf
- https://www.nasca-ng.org/documentation/send_nasca.cfg.pdf

3.5.3 Configure Zabbix reporting

In order to configure Zabbix Error reporting on DRLM, the Zabbix Agent must be installed.

Debian 7/8

```
$ apt-get install zabbix-agent
```

Warning: On debian 7 (wheezy) the backports repository must be configured in order to install zabbix-agent.

RHEL/Centos 6/7

```
$ yum install zabbix-agent
```

Warning: May be needed to add EPEL repositories if not present, because those packages are not included in distribution repositories.

The following options are DRLM defaults, change any of them to your installation requirements in `/etc/drlm/local.conf`.

```
$ vi /etc/drlm/local.conf
```

(continues on next page)

(continued from previous page)

```
#
# REPORT_TYPE=zabbix
# ZABBIX VARIABLES
#
# These are default values and can be overwritten in local.conf according to your
# ↪ZABBIX installation and configuration.
#
ZABBCMD="/usr/bin/zabbix_sender"
ZABBKEY="DRLM"
ZABBCONF="/etc/drlm/alerts/zabbix.cfg"
```

zabbix_sample.cfg

Copy the sample DRLM configuration for Zabbix to previously defined \$ZABBCONF and adjust it to your environment needs.

```
#### DRLM (Disaster Recovery Linux Manager) Zabbix error reporting sample
# ↪configuration file.
#### Default: /etc/drlm/alerts/zabbix.cfg

### Option: ServerActive
# List of comma delimited IP:port (or hostname:port) pairs of Zabbix servers for
# ↪active checks.
# If port is not specified, default port is used.

#ServerActive=monitoring_server:port,monitoring_proxy:port

### Option: Hostname
# Unique, case sensitive hostname.
# Required for active checks and must match hostname as configured on the server.

#Hostname=drlm_server_hostname
```

Note: The configuration on the server side is not in the scope of this documentation. Please check your Zabbix service documentation to configure properly the Trapper item and how to report DRLM alerts.

For reference you can check:

- <https://www.zabbix.com/documentation/3.2/manual/config/items/itemtypes/trapper>
 - https://www.zabbix.com/documentation/3.2/manpages/zabbix_sender
-

3.5.4 Configure Mail reporting

In order to configure Zabbix Error reporting on DRLM, the Heirloom Mailx must be installed.

Debian 7/8

```
$ apt-get install heirloom-mailx
```

RHEL/Centos 6/7

```
$ yum install mailx
```

The following options are DRLM defaults, change any of them to your installation requirements in `/etc/drlm/local.conf`.

```
$ vi /etc/drlm/local.conf

#
# REPORT_TYPE=mail
# MAIL VARIABLES
#
# These are default values and can be overwritten in local.conf according to your
↪MAIL installation and configuration.
#

MAILCMD="/bin/mailx"
MAILSUBJECT="DRLM ERROR ALERT ($HOSTNAME)"
MAILCONF="/etc/drlm/alerts/mail.cfg"
MAIL_TO="root@localhost"
MAIL_CC=""
MAIL_BCC=""
```

mail_sample.cfg

Copy the sample DRLM configuration for Mailx to previously defined \$MAILCONF and adjust it to your environment needs.

```
##### DRLM (Disaster Recovery Linux Manager) Mail error reporting sample configuration
↪file.
##### Default: /etc/drlm/alerts/mail.cfg

### Configure MAIL_FROM [ address(friendly_name) ].

#set from="john@doe.org(John Doe) "

### Set SMTP server configuration [ ipaddr_or_dnsname:port ].

#set smtp=smtp_server:25

### Set SMTP server Auth Options [ Username (mail address) and Password ] if required.

#set smtp-auth=login
#set smtp-auth-user=john@doe.org
#set smtp-auth-password=SoMePaSsWoRd

#####
##### Example using external Gmail smtp servers:

#set from="john@doe.org(John Doe) "
#set smtp-use-starttls
#set ssl-verify=ignore
#set smtp-auth=login
#set smtps=smtp://smtp.gmail.com:587
#set smtp-auth-user=some_user@gmail.com
#set smtp-auth-password=pAsSwOrD
#set nss-config-dir=/etc/ssl/certs
```

Note: The configuration on the Mail server is not in the scope of this documentation. Please check your Mail service configuration to configure properly mailx to report DRLM alerts.

3.5.5 Configure HPOM (former OVO) reporting

In order to configure HPOM(OVO) Error reporting on DRLM, the HPOM(OVO) agent must be installed. This may vary depending on your version, please check your product documentation in order to install it properly. DRLM uses **opcmsg** binary to report errors to HPOM server.

The following options are DRLM defaults, change any of them according to your installation requirements in `/etc/drlm/local.conf`.

```
$ vi /etc/drlm/local.conf:

#
# REPORT_TYPE=ovo
# HP OVO VARIABLES
#
# These are default values and can be overwritten in local.conf according to your HP
↳OVO installation and configuration.
#

OVOCMD="/opt/OV/bin/OpC/opcmsg"
OVOAPP="DRLM"
OVOSEV="Major"
OVOOBJ="OS"
OVOMSGGRP="LINUX"
```

Note: The configuration on the server side is not in the scope of this documentation. Please check HPOM (OVO) documentation to configure properly the server side and define how to report DRLM alerts.

3.6 Network Operations

DRLM can make backups of clients in different networks. So the first step we have to do for the proper functioning of DRLM is register the networks in which later we will register the clients.

DRLM network operations allow us to add, remove, modify and list network of database.

3.6.1 Add Network

This command is used to add networks to DRLM database. It is called like this:

```
$ drlm addnetwork [options]
```

The **drlm addnetwork** has some required options:

- n network_name, --netname network_name**
Select Network name to add.
- i ip, --ipaddr ip**
Network IP address.
- g gateway_ip, --gateway gateway_ip**
Network gateway address.
- m network_mask, --mask network_mask**
Network mask

-s server_ip, **--server** server_ip
Network server address.

Examples:

```
$ drlm addnetwork -i 13.74.90.0 -g 13.74.90.1 -m 255.255.255.0 -s 13.74.90.222 -
↪n vlan12
$ drlm addnetwork -i 13.74.90.0 --gateway 13.74.90.1 --mask 255.255.255.0 --
↪server 13.74.90.222 -n vlan12
$ drlm addnetwork --ipaddr 13.74.90.0 -g 13.74.90.1 -m 255.255.255.0 --server 13.
↪74.90.222 -n vlan12
```

Help options:

-h, **--help**
Show drlm addnetwork help.

Examples:

```
$ drlm addnetwork -h
$ drlm addnetwork --help
```

3.6.2 Delete Network

This command is used to delete networks from DRLM database. It is called like this:

```
$ drlm delnetwork [options]
```

The **drml delnetwork** has some options:

-n network_name, **--netname** network_name
Select Network to delete by NAME.

Examples:

```
$ drlm delnetwork -n vlan12
$ drlm delnetwork -name vlan12
```

-I network_id, **--id** network_id
Select Network to delete by ID.

Examples:

```
$ drlm delnetwork -I 12
$ drlm delnetwork --id 12
```

Help options:

-h, **--help**
Show drlm delnetwork help.

Examples:

```
$ drlm delnetwork -h
$ drlm delnetwork --help
```

3.6.3 Modify Network

This command is used to modify networks from DRLM database. It is called like this:

```
$ drlm modnetwork [options]
```

The **drml modnetwork** has some required options:

-n network_name, --netname network_name
Select Network to change by NAME.

-I network_id, --id network_id
Select Network to change by ID.

Additional options:

-g gateway_ip, --gateway gateway_ip
Set new GATEWAY address to network.

Examples:

```
$ drlm modnetwork -I 12 -g 13.74.91.1 -m 255.255.255.0
$ drlm modnetwork --id 12 --gateway 13.74.91.1 -m 255.255.255.0
$ drlm modnetwork -n vlan12 -g 13.74.91.1 -m 255.255.255.0
$ drlm modnetwork --netname vlan12 --gateway 13.74.91.1 -m 255.255.255.0
```

-m network_mask, --mask network_mask
Assign new MASK to network.

Examples:

```
$ drlm modnetwork -I 12 -m 255.255.0.0
$ drlm modnetwork --id 12 -m 255.255.0.0
$ drlm modnetwork -n vlan12 -m 255.255.0.0
$ drlm modnetwork --netname vlan12 --mask 255.255.0.0
```

-s server_ip, --server server_ip
Assign new SERVER to network.

Examples:

```
$ drlm modnetwork -I 12 -s 13.74.91.221 -m 255.255.255.0
$ drlm modnetwork --id 12 --server 13.74.91.221 -m 255.255.255.0
$ drlm modnetwork -n vlan12 -s 13.74.91.221 -m 255.255.255.0
$ drlm modnetwork --netname vlan12 --server 13.74.91.221 -m 255.255.255.0
```

Help option:

-h, --help
Show drlm modnetwork help.

Examples:

```
$ drlm modnetwork -h
$ drlm modnetwork --help
```

3.6.4 List Networks

This command is used to list the networks from DRLM database. It is called like this:

```
$ drlm listnetwork [options]
```

The **drmlm listnetwork** has some options:

-n network_name, **--netname** network_name
Select Network to list.

Exmples:

```
$ drlm listnetwork -n vlan12
$ drlm listnetwork --netname vlan12
```

-A, --all
List all networks. This option is set by default if any option is specified.

Examples:

```
$ drlm listnetwork
$ drlm listnetwork -A
$ drlm listnetwork -all
```

Help options:

-h, --help
Show drlm listnetwork help.

Examples:

```
$ drlm listnetwork -h
$ drlm listnetwork --help
```

3.7 Client Operations

DRLM client operations allow us to add, remove, modify and list clients of database.

3.7.1 Add Client

This command is used to add clients to DRLM database. It is called like this:

```
$ drlm addclient [options]
```

If the client you wish to add is online (network reachable), you will only need to set its IP in CIDR notation in order to add it to the database. It will then automatically fetch and prompt all the required client parameters (hostname, network and MAC address), leaving to you the option to keep and save those parameters or to enter them manually in case you refuse.

In this case, the **drmlm addclient** has the following required options:

-i ip_in_CIDR_format, **--ipaddr** ip_in_CIDR_format
Examples:

```
$ drlm addclient -i 192.168.0.15/24
```

If the **drmlm addclient** does not correctly fetch the client's hostname, you can set it manually in the same command.

Examples:

```
$ drlm addclient -i 192.168.0.15/24 -c rear-debian
```

-I, --installclient

If the client is network reachable you can also automatically install the client when is added to DRLM. So in only one command the client is added and installed. Installclient have additional options than you can add behind the -I. For more information about Installclient read the “Install Client” section.

Examples:

```
$ drlm addclient -i 192.168.0.15/24 -I
$ drlm addclient -i 192.168.0.15/24 -c rear-debian -I
$ drlm addclient -i 192.168.0.15/24 -c rear-debian -I -u root -U http://url.to.
↪rear/download
```

If the client is not network reachable when you want to register it in the database or you wish to manually enter all the required parameters, you can do it with the required options available:

-c client_name, --client client_name
Set the client’s name.

Note: It is not mandatory, but recommended that the client_name is the same as the client hostname.

-i ip, --ipaddr ip
Client IP address (not in CIDR notation if you are manually adding all the required parameters).

-M mac_address, --macaddr mac_address
Client MAC address.

-n network_name, --netname network_name
Client NETWORK.

Examples:

```
$ drlm addclient -c clientHost1 -M 00-40-77-DB-33-38 -i 13.74.90.10 -n vlan12
$ drlm addclient --client clientHost1 --macaddr 00-40-77-DB-33-38 -i 13.74.90.10 -
↪n vlan12
```

Warning: If the network_name doesn’t exist in DRLM database you will get an error. First of all register the network where the client will be registered.

Help option:

-h, --help
Show drlm addclient help.

Examples:

```
$ drlm addclient -h
$ drlm addclient --help
```

3.7.2 Install Client

This command is used to install and configure DRLM and ReaR on a remote Server. It is called like this:


```
$ drlm instclient [options]
```

The **drlm instclient** has some required options:

- c** client_name, **--client** client_name
Select Client name to add.
- I** client_id, **--id** client_id
Client Id.

Additional options:

- u** user, **--user** user
User with admin privileges to install and configure software

Note: if not user is specified root will be used.

- U** url_rear, **--url_rear** url_rear
rpm or deb package for specific distro. For example http://download.opensuse.org/repositories/Archiving:/Backup:/Rear/Debian_7.0/all/rear_1.17.2_all.deb

Note: If not url is specified will be used the package defined in “REAR DEB PACKAGE URL” section of /usr/share/drlm/conf/default.conf

Examples:

```
$ drlm instclient -c ReaRCli1 -u admin -U http://download.opensuse.org/
↪repositories/Archiving:/Backup:/Rear/Debian_7.0/all/rear_1.17.2_all.deb
$ drlm instclient -c ReaRCli2
```

Help option:

- h, --help**
Show drlm instclient help.

Examples:

```
$ drlm instclient -h
```

3.7.3 Delete Client

This command is used to delete clients from DRLM database. It is called like this:

```
$ drlm delclient [options]
```

The **drlm delclient** has some required options:

- c** client_name, **--client** client_name
Select Client to delete by NAME.
- I** client_id, **--id** client_id
Select Client to delete by ID.

Examples:

```
$ drlm delclient -c clientHost1
$ drlm delclient --client clientHost1
$ drlm delclient -I 12
$ drlm delclient --id 12
```

Help option:

-h, --help
Show drlm delclient help.

Examples:

```
$ drlm delclient -h
$ drlm delclient --help
```

3.7.4 Modify Client

This command is used to modify clients from DRLM database. It is called like this:

```
$ drlm modclient [options]
```

The **drlm modclient** has some required options:

-c client_name, **--client** client_name
Select Client to change by NAME

-I client_id, **--id** client_id
Select Client to change by ID

Additional options:

-i ip, **--ipaddr** ip
Set new IP address to client.

Examples:

```
$ drlm modclient -c clientHost1 -i 13.74.90.10
```

-M mac_address, **--macaddr** mac_address
Set new MAC address to client.

Examples:

```
$ drlm modclient -c clientHost1 -M 00-40-77-DB-33-38
$ drlm modclient --client clientHost1 --macaddr 00-40-77-DB-33-38
$ drlm modclient -I 12 --macaddr 00-40-77-DB-33-38
$ drlm modclient --id 12 -M 00-40-77-DB-33-38
```

-n network_name, **--netname** network_name
Assign new NETWORK to client.

Examples:

```
$ drlm modclient -c clientHost1 -n vlan12
$ drlm modclient --client clientHost1 --netname vlan12
$ drlm modclient -I 12 --netname vlan12
$ drlm modclient --id 12 -n vlan12
```

Help option:

-h, --help
Show drlm modclient help.

Examples:

```
$ drlm modclient -h
$ drlm modclient --help
```

3.7.5 List Clients

This command is used to list the clients stored at the database. It is called like this:

```
$ drlm listclient [options]
```

The **drlm listclient** has some options:

-c client_name, --client client_name
Select Client to list.

Examples:

```
$ drlm listclient -c clientHost1
$ drlm listclient --client clientHost1
```

-A, --all
List all clients. This option is set by default if any option is specified.

Examples:

```
$ drlm listclient
$ drlm listclient -A
$ drlm listclient --all
```

Help option:

-h, --help
Show drlm listclient help.

Examples:

```
$ drlm listclient -h
$ drlm listclient --help
```

3.8 Backup Operations

DRLM backup operations allow us to remotely create new backups of clients, enable and disable restore points and make listings of backups created among other things.

3.8.1 Run Backup

This command is used to Run remote client backup from DRLM. It is called like this:

```
$ drlm runbackup [options]
```

The **drlm runbackup** has several options:

-c client_name, **--client** client_name
Select Client to remotely run backup by name.

Examples:

```
$ drlm runbackup -c clientHost1
$ drlm runbackup --client clientHost1
```

-I client_id, **--id** client_id
Select Client to remotely run backup by ID.

Examples:

```
$ drlm runbackup -I 12
$ drlm runbackup --id 12
```

Help option:

-h, --help
Show drlm runbackup help.

Examples:

```
$drlm runbackup -h
$drlm runbackup --help
```

3.8.2 Delete Backup

This command is used to delete backups from DRLM database. It is called like this:

```
$ drlm delbackup [options]
```

Warning: To remove a backup, it must be disabled.

The **drlm delbackup** has some required options:

-c client_name, **--client** client_name
Select Client to delete the backups.

-I backup_id, **--id** backup_id
Select Backup to delete by ID.

-A, --all
Delete All backup.

Examples:

```
$ drlm delbackup -I 1.2015030121245
$ drlm delbackup --id 1.2015030121245
$ drlm delbackup -c clientHost1 -A
$ drlm delbackup --client clientHost1 --all
```

Help option:

-h, --help
Show drlm delbackup help.

Examples:

```
$ drlm delbackup -h
$ drlm delbackup --help
```

3.8.3 List Backups

This command is used to list the backups that we have stored on the server. It is called like this:

```
$ drlm listbackup [options]
```

The **drlm listbackup** has some options:

-c *client_name*, **--client** *client_name*
Select Client to list its backups.

Examples:

```
$ drlm listbackup -c clientHost1
$ drlm listbackup --client clientHost1
```

-A, --all
List all backups. This option is set by default if any option is specified.

Examples:

```
$ drlm listbackup
$ drlm listbackup -A
$ drlm listbackup --all
```

Help option:

-h, --help
Show this help

Examples:

```
$ drlm listbackup -h
$ drlm listbackup --help
```

3.8.4 Backup Manager

This command is used to enable or disable client restore points. Is also used to set a restore point by default. It is called like this:

```
$ drlm bkpmgr [options]
```

The **drlm bkpmgr** has some required options:

-I *backup_id*, **--id** *backup_id*
Select Backup ID to modify

-e, --enable
Enable Backup

-d, --disable
Disable Backup

Examples:

```
$drlm bkpmgr -I 1.20140519065512 -e
$drlm bkpmgr -I 1.20140519065512 -d
$drlm bkpmgr --id 1.20140519065512 -e
```

Help option:

-h, --help
Show drlm bkmgr help.

Examples:

```
$ drlm bkmgr -h
$ drlm bkmgr --help
```

3.8.5 Export/Import Backups

Since version 2.1.0 the possibility to import or export backups from other DRLM servers has been added. To export a backup:

Export Backups

This command is used to export a backup that we have stored on the server. It is called like this:

```
$ drlm expbackup [options]
```

The **drlm expbackup** has the following required options:

-I backup_id, --id backup_id
Enter the backup ID you would like to export.

-f destination_file, --file destination_file
Enter the output path in which you would like to export the backup,

Examples:

```
$ drlm expbackup -I 2.20170125103105 -f /tmp/export.dr
```

You could now save or copy the exported backup to another DRLM server.

Help option:

-h, --help
Shows help menu.

Examples:

```
$ drlm expbackup -h
$ drlm expbackup --help
```

Import Backups

This command is used to import a backup that we have received from other DRLM server. It is called like this:

```
$ drlm impbackup [options]
```

The **drlm impbackup** has the following required options:

-c client_name, **--client** client_name

You need to first register the client in the database before importing an exported DRLM backup.

-f file, **--file** file

Set the destination path of the backup to import.

Examples:

```
$ drlm impbackup --client rear-debian -f /tmp/export.dr
```

Help option:

-h, **--help**

Shows help menu.

Examples:

```
$ drlm expbackup -h
$ drlm expbackup --help
```

3.8.6 Backup Job Scheduler

Since version 2.1.0 backup tasks can be scheduled. The **drlm backup scheduler** allows you to **add**, **list** and **delete** scheduled jobs. You can also enable or disable the schedule function (by default it is enabled). You can set backup operations to run on a specified date and time by running:

Add Jobs

This command is used to plan backup jobs in DRLM. It is called like this:

```
$ drlm addjob [options]
```

Required options:

-c client_name, **--client** client_name

Client for which you want to run a scheduled backup.

-s start_date, **--start_date** start_date

Start date and time for the scheduled backup. Format: YYYY-MM-DDTHH:MM

Optional arguments:

-e end_date, **--end_date** end_date

End date and time for the scheduled backup. Format: YYYY-MM-DDTHH:MM

-r repeat_time, **--repeat** repeat_time

This argument specifies the time a backup will be performed between the start and the end date of a scheduled backup (if any end_date is set). You can specify the repeating pattern in min(s) or minute(s), hour(s), day(s), week(s), month(s) and year(s).

Examples:

```
$ drlm addjob -c rear-debian -s 2017-01-30T21:00
$ drlm addjob --client rear-centos -s 2017-02-03T08:00 -e 2017-02-05T23:00 -r ↵
↵1hour
```

Help option:

-h, --help
Shows help menu.

Examples:

```
$ drlm addjob -h
$ drlm addjob --help
```

List Jobs

This command is used to list backup jobs planned in DRLM. It is called like this:

```
$ drlm listjob [options]
```

-J job_id, --job_id job_id
To list a job by its ID.

-c client_name, --client client-name
To list all the jobs scheduled for a specific client.

-A, --all
To list all the active scheduled jobs.

Examples:

```
$ drlm listjob -A
$ drlm listjob -c rear-suse
$ drlm listjob --job_id 3
```

Help option:

-h, --help
Shows help menu.

Examples:

```
$ drlm listjob -h
$ drlm listjob --help
```

Delete Jobs

This command is used to delete planned backup jobs in DRLM. It is called like this:

```
$ drlm deljob [options]
```

-c client_name, --client client_name
To delete all scheduled jobs for a specific client.

-J job_id, --job_id job_id
To delete a specific scheduled backup job.

Examples:

```
$ drlm deljob -J 5
$ drlm deljob -c rear-centos
```

Help option:

-h, --help
Shows help menu.

Examples:

```
$ drlm deljob -h
$ drlm deljob --help
```

Scheduler Management

With this command you can **enable or disable** the job scheduler facility or force to **run** jobs planned at “now” by running:

```
drmlm sched [options]
```

-e, --enable
Enables job scheduler utility.

-d, --disable
Disables job scheduler utility.

-r, --run
Runs all planned jobs (starting from the nearest date).

Examples:

```
$ drlm sched -e
$ drlm sched -r
```

Help option:

-h, --help
Shows help menu.

Examples:

```
$ drlm sched -h
$ drlm sched --help
```

3.9 Building GRUB2 for diferent platfoms

Since DRLM version 2, we moved to GRUB2 to provide the netboot images to start ReaR recovery images from network. This movement was the first step to provide support for mulitple platforms for GNU/Linux because GRUB2 supports multiple architerctures.

At this time DRLM built packages include all documented platforms in this guide.

3.9.1 Prepare your build host

Note: This document describes the process of building DRLM GRUB2 netboot images for diferent platforms with a debian machine. The process should be the same on other distros, just adjusting package dependencies for target distro and install them with the package management tools provided by each distro should work without problems.

Install required packages

```
$ apt-get install bison libopts25 libseline1-dev autogen \  
m4 autoconf help2man libopts25-dev flex libfont-freetype-perl \  
automake autotools-dev libfreetype6-dev texinfo
```

Download GRUB2 sources

```
$ cd /usr/src  
  
$ wget http://alpha.gnu.org/gnu/grub/grub-2.02~beta3.tar.gz  
  
$ tar -xzf grub-2.02~beta3.tar.gz  
  
$ cd grub-2.02~beta3
```

3.9.2 Start build process

Warning: All documented grub2 image builds are included in drlm packages, this document will be a kind of guide for troubleshooting and testing on new GRUB2 versions and also a guide to, contributors of future drlm grub2 images, on new supported platforms to the project.

Provide DRLM branded GRUB2 build

```
$ vi grub-core/normal/main.c  
  
.. replace:  
msg_formatted = grub_xasprintf (_("GNU GRUB version %s"), PACKAGE_VERSION);  
  
.. with:  
msg_formatted = grub_xasprintf (_("DRLM Boot Manager (GNU GRUB2)"), PACKAGE_VERSION);
```

Prepare your build environment:

```
$ ./autogen.sh
```

On next steps we will proceed with configuration and build for each platform needed.

For i386-pc:

```
$ ./configure --disable-werror  
$ make && make install  
  
$ /usr/local/bin/grub-mknetdir -d /usr/local/lib/grub/i386-pc --net-directory=/tmp  
Netboot directory for i386-pc created. Configure your DHCP server to point to /tmp/  
↪boot/grub/i386-pc/core.0
```

For 32-bit EFI:

```
$ ./configure --with-platform=efi --target=i386 --disable-werror
$ make && make install

$ /usr/local/bin/grub-mknetdir -d /usr/local/lib/grub/i386-efi --net-directory=/tmp
Netboot directory for i386-efi created. Configure your DHCP server to point to /tmp/
↪boot/grub/i386-efi/core.efi
```

For 64-bit (U)EFI:

```
$ ./configure --with-platform=efi --target=x86_64 --disable-werror
$ make && make install

$ /usr/local/bin/grub-mknetdir -d /usr/local/lib/grub/x86_64-efi --net-directory=/tmp
Netboot directory for x86_64-efi created. Configure your DHCP server to point to /tmp/
↪boot/grub/x86_64-efi/core.efi
```

Create a tarball with targeted platform netboot image

```
$ cd /tmp
$ tar -cvzf drlm_grub2_<target>-<platform>.tar.gz boot/
```

Note: This gzipped tarball can be extracted to DRLM \$STORDIR on your DRLM server, for testing purposes or to provide support to new platforms not yet provided by DRLM package builds.

Note: This section should change continuously due to changes in DRLM development, please be patient. Any question regarding DRLM development, please use [DRLM Dev Forum](#)¹³. Thanks!

3.10 About DRLM Docs

DRLM Docs contains comprehensive documentation on the DRLM (Disaster Recovery Linux Manager). This page describes documentation's licensing, editions, and versions, and describes how to contribute to the DRLM Docs.

For more information on DRLM, see [About DRLM Project](#)¹⁴. To download DRLM, see the downloads page.

3.10.1 License

This documentation is licensed under a Creative Commons [Attribution-NonCommercial-ShareAlike 4.0 International](#)¹⁵ (i.e. "CC-BY-NC-SA") license.

The DRLM Manual is copyright © 2017 Brain Updaters, S.L.L.

¹³ <https://groups.google.com/forum/#!forum/drlm-dev>

¹⁴ <http://drlm.org/about/>

¹⁵ <http://creativecommons.org/licenses/by-nc-sa/4.0/>

3.10.2 Contributing

Please, we encourage you to help us to improve this documentation.

To contribute to documentation the Github interface enables users to report errata or missing sections, discuss improvements and new sections through the issue-tracker at: [DRLM Docs GitHub Issue Tracker](#)¹⁶.

3.10.3 Product Features

The following features are supported on the most recent releases of DRLM. Anything labeled as (NEW!) was added as the most recent release. New functionality for previous releases can be seen in the next chapter that details each release.

- Hot maintenance capability. A client backup can be made online while the system is running.
- Command line interface. DRLM doesnot require a graphical interface to run. (console is enough).
- Multiarch netboot client support (x86_64-efi, i386-efi, i386-pc)
- Automatic client intallation from DRLM server
- Parallel backups
- Error reporting support to:
 - HP OpenView
 - Nagios (NSCA & NSCA-ng) (NEW!)
 - Zabbix
 - Mail
- Centralized backup scheduling with a job scheduler (NEW!)
- Export and Import backup between DRLM servers or DRLM clients (NEW!)

3.10.4 DRLM Version 2.1.2 (March 2017) - Release Notes

- SUDO_CMDS_DRLM added in default.conf allowing to easy add new sudo commands.
- Automatic creation of /etc/sudoers.d if not exists on systems RedHat/CenOS 5.
- Fixed some errors for dependencies on default.conf.
- DRLM_USER variable deleted on addclient and help.
- Added sudo for command stat to allow check size on File Systems without perms.
- Sudo configuration files are dynamically created according to the OS type.
- Solved problem for start services with non root user.

3.10.5 DRLM Version 2.1.1 (February 2017) - Release Notes

- Solved some of bugs. (issue #49, #50)
- No Client ID required for delete backups. (issue #40)
- No Client ID required for manage backups. (issue #46)

¹⁶ <https://github.com/brainupdaters/drlm-docs/issues>

- bkpmgr: Persistent mode deleted.
- Solved PXE files: forced console=ttyS0 in kernel options. (issue #52)
- Solved hardcoded PXE filenames (initrd.xz (lzma) now supported). (issue #52)
- While recommended, It ain't mandatory to use hostname as client_name. (issue #52)
- Solved drlm user hardcoded in installclient. (issue #51)
- NAGSRV and NAGPORT added in default.conf.

3.10.6 DRLM Version 2.1.0 (February 2017) - Release Notes

- DRLM reporting with nsca-ng, nsca. (issue #47)
- DRLM Server for SLES. (issue #45)
- Support for drlm unattended installation (instclient) on Ubuntu (issue #43)
- NEW Import & Export DR images between DRLM servers. (issue #39)
- Pass DRLM global options to ReaR. (issue #37)
- New DRLM backup job scheduler (issue #35)
- Addclient install mode (automatize install client after the client creation) (issue #32)
- Solved lots of bugs

3.10.7 DRLM Version 2.0.0 (July 2016) - Release Notes

- Multiarch netboot with GRUB2 - x86_64-efi i386-efi i386-pc - (issue #2)
- New installclient workflow (issue #5)
- Added support for systemd distros - RHEL7 CentOS7 Debian8 - (issue #14)
- Use bash socket implementation instead of netcat (issue #15)
- runbackup workflow enhancement with sparse raw images with qemu-img reducing backup time and improving management (issue #16)
- Added support for parallel backups on DRLM (issue #22)
- Added support for new DB backend sqlite3 (issue #23)
- Added support for Nagios error reporting (issue #28)
- Added support for Zabbix error reporting (issue #29)
- Added support for Mail error reporting (issue #30)
- Added timeout var for Sqlite in sqlite3-driver.sh for avoiding database locks.
- Added source of local.conf and site.conf files in drlm-stord
- Solved lots of bugs
- DRLM documentation updated to reflect version 2.0 changes

Note: This documentation is under constant development. Please be patient. . .

CHAPTER 4

Indexes and tables

- genindex

Symbols

- A, -all
 - 'drlm-delbackup' command line option, 48
 - 'drlm-listbackup' command line option, 49
 - 'drlm-listclient' command line option, 47
 - 'drlm-listjob'-arguments: command line option, 52
 - 'drlm-listnetwork' command line option, 43
- I backup_id, -id backup_id
 - 'drlm-bkpmgr' command line option, 49
 - 'drlm-delbackup' command line option, 48
 - 'drlm-expbackup' command line option, 50
- I client_id, -id client_id
 - 'drlm-delclient' command line option, 45
 - 'drlm-instclient' command line option, 45
 - 'drlm-modclient' command line option, 46
 - 'drlm-runbackup' command line option, 48
- I network_id, -id network_id
 - 'drlm-delnetwork' command line option, 41
 - 'drlm-modnetwork' command line option, 42
- I, -installclient
 - 'drlm-addclient' command line option, 44
- J job_id, -job_id job_id
 - 'drlm-deljob'-required-options: command line option, 52
 - 'drlm-listjob'-arguments: command line option, 52
- M mac_address, -macaddr mac_address
 - 'drlm-addclient' command line option, 44
 - 'drlm-modclient' command line option, 46
- U url_rear, -url_rear url_rear
 - 'drlm-instclient' command line option, 45
- c client_name, -client client_name
 - 'drlm-listjob'-arguments: command line option, 52
- c client_name, -client client_name
 - 'drlm-addclient' command line option, 44
 - 'drlm-addjob' command line option, 51
 - 'drlm-delbackup' command line option, 48
 - 'drlm-delclient' command line option, 45
 - 'drlm-deljob'-required-options: command line option, 52
- 'drlm-expbackup' command line option, 50
- 'drlm-instclient' command line option, 45
- 'drlm-listbackup' command line option, 49
- 'drlm-listclient' command line option, 47
- 'drlm-modclient' command line option, 46
- 'drlm-runbackup' command line option, 47
- d, -disable
 - 'drlm-bkpmgr' command line option, 49
 - 'drlm-sched'-available-options: command line option, 53
- e end_date, -end_date end_date
 - 'drlm-addjob' command line option, 51
- e, -enable
 - 'drlm-bkpmgr' command line option, 49
 - 'drlm-sched'-available-options: command line option, 53
- f destination_file, -file destination_file
 - 'drlm-expbackup' command line option, 50
- f file, -file file
 - 'drlm-expbackup' command line option, 51
- g gateway_ip, -gateway gateway_ip
 - 'drlm-addnetwork' command line option, 40
 - 'drlm-modnetwork' command line option, 42
- h, -help
 - 'drlm-addclient' command line option, 44
 - 'drlm-addjob' command line option, 51
 - 'drlm-addnetwork' command line option, 41
 - 'drlm-bkpmgr' command line option, 50
 - 'drlm-delbackup' command line option, 48
 - 'drlm-delclient' command line option, 46
 - 'drlm-deljob'-required-options: command line option, 52
 - 'drlm-delnetwork' command line option, 41
 - 'drlm-expbackup' command line option, 50, 51
 - 'drlm-instclient' command line option, 45
 - 'drlm-listclient' command line option, 47
 - 'drlm-listjob'-arguments: command line option, 52
 - 'drlm-listnetwork' command line option, 43
 - 'drlm-modclient' command line option, 46
 - 'drlm-modnetwork' command line option, 42

- 'drlm-runbackup' command line option, 48
- 'drlm-sched'-available-options: command line option, 53
- h, --help
 - 'drlm-listbackup' command line option, 49
- i ip, --ipaddr ip
 - 'drlm-addclient' command line option, 44
 - 'drlm-addnetwork' command line option, 40
 - 'drlm-modclient' command line option, 46
- i ip_in_CIDR_format, --ipaddr ip_in_CIDR_format
 - 'drlm-addclient' command line option, 43
- m network_mask, --mask network_mask
 - 'drlm-addnetwork' command line option, 40
 - 'drlm-modnetwork' command line option, 42
- n network_name, --netname network_name
 - 'drlm-addclient' command line option, 44
 - 'drlm-addnetwork' command line option, 40
 - 'drlm-delnetwork' command line option, 41
 - 'drlm-listnetwork' command line option, 43
 - 'drlm-modclient' command line option, 46
 - 'drlm-modnetwork' command line option, 42
- r repeat_time, --repeat repeat_time
 - 'drlm-addjob' command line option, 51
- r, --run
 - 'drlm-sched'-available-options: command line option, 53
- s server_ip, --server server_ip
 - 'drlm-addnetwork' command line option, 40
 - 'drlm-modnetwork' command line option, 42
- s start_date, --start_date start_date
 - 'drlm-addjob' command line option, 51
- u user, --user user
 - 'drlm-instclient' command line option, 45
- 'drlm-addclient' command line option
 - I, --installclient, 44
 - M mac_address, --macaddr mac_address, 44
 - c client_name, --client client_name, 44
 - h, --help, 44
 - i ip, --ipaddr ip, 44
 - i ip_in_CIDR_format, --ipaddr ip_in_CIDR_format, 43
 - n network_name, --netname network_name, 44
- 'drlm-addjob' command line option
 - c client_name, --client client_name, 51
 - e end_date, --end_date end_date, 51
 - h, --help, 51
 - r repeat_time, --repeat repeat_time, 51
 - s start_date, --start_date start_date, 51
- 'drlm-addnetwork' command line option
 - g gateway_ip, --gateway gateway_ip, 40
 - h, --help, 41
 - i ip, --ipaddr ip, 40
 - m network_mask, --mask network_mask, 40
 - n network_name, --netname network_name, 40
 - s server_ip, --server server_ip, 40
- 'drlm-bkpmgr' command line option
 - I backup_id, --id backup_id, 49
 - d, --disable, 49
 - e, --enable, 49
 - h, --help, 50
- 'drlm-delbackup' command line option
 - A, --all, 48
 - I backup_id, --id backup_id, 48
 - c client_name, --client client_name, 48
 - h, --help, 48
- 'drlm-delclient' command line option
 - I client_id, --id client_id, 45
 - c client_name, --client client_name, 45
 - h, --help, 46
- 'drlm-deljob'-required-options: command line option
 - J job_id, --job_id job_id, 52
 - c client_name, --client client_name, 52
 - h, --help, 52
- 'drlm-delnetwork' command line option
 - I network_id, --id network_id, 41
 - h, --help, 41
 - n network_name, --netname network_name, 41
- 'drlm-expbackup' command line option
 - I backup_id, --id backup_id, 50
 - c client_name, --client client_name, 50
 - f destination_file, --file destination_file, 50
 - f file, --file file, 51
 - h, --help, 50, 51
- 'drlm-instclient' command line option
 - I client_id, --id client_id, 45
 - U url_rear, --url_rear url_rear, 45
 - c client_name, --client client_name, 45
 - h, --help, 45
 - u user, --user user, 45
- 'drlm-listbackup' command line option
 - A, --all, 49
 - c client_name, --client client_name, 49
 - h, --help, 49
- 'drlm-listclient' command line option
 - A, --all, 47
 - c client_name, --client client_name, 47
 - h, --help, 47
- 'drlm-listjob'-arguments: command line option
 - A, --all, 52
 - J job_id, --job_id job_id, 52
 - c client_name, --client client-name, 52
 - h, --help, 52
- 'drlm-listnetwork' command line option
 - A, --all, 43
 - h, --help, 43
 - n network_name, --netname network_name, 43
- 'drlm-modclient' command line option
 - I client_id, --id client_id, 46

- M mac_address, --macaddr mac_address, 46
- c client_name, --client client_name, 46
- h, --help, 46
- i ip, --ipaddr ip, 46
- n network_name, --netname network_name, 46

‘drlm-modnetwork‘ command line option

- I network_id, --id network_id, 42
- g gateway_ip, --gateway gateway_ip, 42
- h, --help, 42
- m network_mask, --mask network_mask, 42
- n network_name, --netname network_name, 42
- s server_ip, --server server_ip, 42

‘drlm-runbackup‘ command line option

- I client_id, --id client_id, 48
- c client_name, --client client_name, 47
- h, --help, 48

‘drlm-sched‘-available-options: command line option

- d, --disable, 53
- e, --enable, 53
- h, --help, 53
- r, --run, 53