TECHNICAL WHITE PAPER

How It Works:
Bare Metal Restore (BMR) for Linux

Marcus Faust, Damani Norman, and Eric Chang
May 2020
RWP-0508

TABLE OF CONTENTS

3 CHALLENGES

3 RUBRIK'S BMR DESIGN AND PROCESS FLOW

6 LINUX BMR: A STEP-BY-STEP INSTRUCTION GUIDE

- 6 Installation
- 9 Configuration
- 10 Backup
- 11 Recovery
  - 11 Prerequisites
  - 12 Booting the recovery system
  - 17 Recover data

21 KNOWN ISSUES

22 TROUBLESHOOTING

23 VERSION HISTORY

CHALLENGES

Rubrik was founded in 2014 to reinvent the data management software space which had not seen transformative innovation in 20 plus years. At that time the company was focused on all things modern - virtualization, cloud, APIs, DevOps, containers, etc... As the company grew, customers also asked Rubrik to take it's simple and elegant design to other areas of the data center including databases (SQL Server, Oracle), NAS shares, and physical servers (Windows, Linux, AIX, Solaris).

VMware ESX 1.0 was released nearly 20 years ago but believe it or not there are still many physical servers still standing (mainly due to performance requirements) that need to be managed.[1] Part of managing these physical assets is, of course, standard backup and recovery of the data residing on those servers. Before the era of virtualization, backup and recovery software vendors provided the ability to perform restores of the operating systems to similar or dissimilar hardware platforms using a feature called "bare metal restore" or BMR.

Supporting BMR is challenging for software companies. Some of the issues include:

- Finding the right version of the OS
- Re-applying patches to the correct level
- Finding and reinstalling drivers for specific hardware
- Reinstalling the backup agent
- Remembering the disk partitioning configurations and recreating them

Instead of writing their own software to support BMR, many backup software vendors partnered or acquired technologies - (VERITAS Software made the acquisition of a company called The Kernel Group (TKG)[2] earlier in 2002 for its bare metal capabilities while EMC acquired a company called Indigo Stone in 2007 for its HomeBase BMR software.[3]

Even today many modern companies today are partnering with 3rd-party companies to address this gap in their software portfolios. Partnering with others is one way to solve the BMR problems but these technologies generally require separate management consoles, doubling the number of agents on the hosts, bloating the size of a single agent, and most importantly consuming more disk storage as the BMR images are usually separate from the "normal" backup images.

RUBRIK'S BMR DESIGN AND PROCESS FLOW

Rubrik's maniacal desire to simplify what was once extremely complex in legacy backup and recovery software is everywhere within the product line:

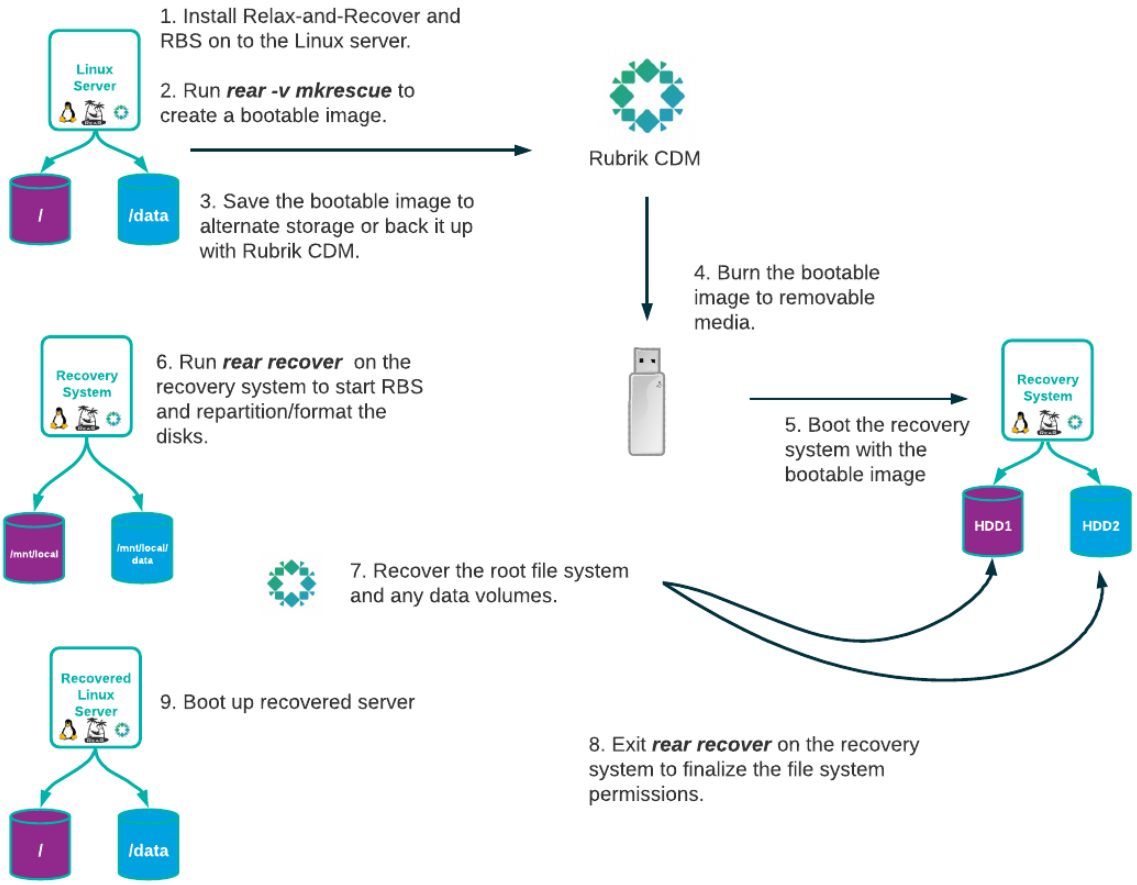
- Eliminating storage configuration complexity
- Increase operational efficiency by using software to automate the scheduling and retention of backups via Rubrik's SLA Domain Policies[4]
- Managing data replication and archival to cloud
- Google-like search functionality
- Simplifying restores using Instant Recovery

1 https://www.virten.net/vmware/esxi-release-build-number-history/
2 https://www.bizjournals.com/austin/stories/2002/01/07/daily14.html
3 https://www.crn.com/news/storage/199400218/emc-adds-bare-metal-recovery-by-acquiring-indigo-stone.htm
4 https://www.rubrik.com/blog/rubrik-sla-domain-settings-ops/

To address these challenges of BMR for Linux, Rubrik looked towards the open source project Relax-and-Recover[5] (ReaR[6]) . The approach the Relax-and-Recover project took fits well with Rubrik's design philosophy. It allows for:

- Restore to dissimilar hardware: The product should support restores from one hardware platform to another.
- Remove the requirement and reliance of costly 3rd-party software (3rd party meaning another software company, non-open source).

The Rubrik CDM integration with Relax and Recover allows Rubrik Cloud Data Management (CDM) to perform bare metal recovery of Linux systems that are supported by Relax-and-Recover. This is done by including the installed Rubrik CDM Rubrik Backup Service (RBS) connector files in the bootable image that is created by Relax-and-Recover. Relax-and-Recover itself works by producing a bootable image of a Linux system's operating system. When the recovery system is booted from this image, it can repartition the target disks. Once that is done it initiates a restore from backup. Restores to different hardware are also possible, which can enable migrations.



At a high level Linux servers (physical and virtual) are protected at a file level by installing RBS on them. In this example the `/` and `/data` file systems are being protected. These file system level backups share the same characteristics as other Rubrik backup object types, incremental-forever backup approach, search/indexing, Instant Recovery/Live Mount, etc...

5 https://github.com/rear/rear
6 http://relax-and-recover.org/

If this server is unable to boot and needs to be recovered at the bare metal level, a bootable image needs to be created. This is done by installing Relax-and-Recover along with the Rubrik RBS connector (step 1). After installing Relax-and-Recover its configuration file is updated to specify that Rubrik CDM is the backup software. The Rubrik integration with Relax-and-Recover causes it to include RBS in the bootable image. This allows the recovery system to access Rubrik CDM during restore. After the configuration file is updated, the command `rear -v mkrescue` is run to create a bootable image (step 2). The -v option is used to see the verbose output and troubleshoot any errors.

One the `rear -v mkrescue` command is working properly it can be scheduled to run regularly in cron or via the Rubrik software as a pre-command to the backup (step 3). Having the bootable image file created by the Rubrik CDM fileset pre-process step guarantees that any changes to the operating system are stored in the backups. Scheduling the boot image creation outside of Rubrik CDM, especially on a non-daily basis may result in the boot image not being current.

By default `rear -v mkrescue` saves the ISO file to `/var/lib/rear/output/rear-<hostname>.iso`. Rubrik CDM will backup the ISO file from this location as part of a regular fileset backup. Alternatively the ISO file can be stored in another location that is easy to access.

When it is time to recover the Linux server, either back to the same hardware or to new hardware, verify that the recovery system has a compatible disk layout with the Linux system that is being restored to it. See the Relax-and-Recover Layout configuration[7] page for more details. Next burn the bootable image to boot media that is supported by the recovery system (step 4). The boot image can be recovered from Rubrik CDM by searching the Linux system's backups if it was previously included in its fileset. Otherwise a copy of the bootable image will need to be obtained from whatever storage location it was saved to.

The recovery system is booted using the newly created boot media (step 5). Once it is running the command `rear recover` is run on the recovery system (step 6). This command allows the parameters necessary to run the Rubrik RBS connector to be entered. It then starts the Rubrik RBS connector. After starting RBS, the `rear recover` command repartitions the recovery system's disk to match what was on the original Linux server. Once the recovery system is repartitioned the `rear recover` command requests that the operator recover the file system data from Rubrik CDM. The operator then returns to the Rubrik console and performs an export of any data to restore, including the `/` file system. The export is redirected to the `/mnt/local` directory on the recovery system[8]. This directory points to the repartitioned file system(s) on the recovery system . If the original hardware is being restored to the export is performed directly. If the recovery system is not replacing the original Linux system the export is redirected to the new recovery system.

Once the export process finishes on Rubrik CDM, return to the recovery system and exit the `rear recover` command prompt (step 8). At this point Relax-and-Recover will fix the operating system file permissions and set up the bootloader. When the process finishes the recovery system is rebooted (step 9). Upon reboot the Linux system will be recovered and ready for use.

Note: Care should be taken in this setup with the recovered Linux system's networking. If static IP addresses were used the original IP address will be configured. This will cause a conflict if the original Linux system is still running. Booting the recovered Linux system in isolation and changing its IP address is advisable. Another issue may occur if DHCP was being used on the original Linux system and it was restored to new hardware. The MAC address of the recovered Linux system will have changed from the original causing it to get a new IP address. Any systems needing to access the recovered Linux system will need to use this new IP address.

7 https://github.com/rear/rear/blob/master/doc/user-guide/06-layout-configuration.adoc

8 While booted from the bootable media the `/` file system on the recovery system points to the bootable media.

LINUX BMR: A STEP-BY-STEP INSTRUCTION GUIDE

INSTALLATION

At this time the only version of Relax-and-Recover that supports Rubrik CDM is in the master branch of the Relax-and-Recover project. That can be found here: https://github.com/rear/rear. Once the next release of Relax-and-Recover is produced the regular OS package installers can be used to install Relax-and-Recover with support for Rubrik CDM. This process is described in the Relax-and-Recover project website[9]. In the meantime the Relax-and-Recover is installed by running the ***make install*** command from within the cloned project directory.

1. Install the Rubrik RBS Agent as directed by the Rubrik Users Guide.

   CentOS Example:

   Run `curl -kLOJ https://<rubirk_node_ip>/connector/rubrik-agent.x86_64.rpm`

```
[root@rear-cdm ~]# mkdir rubrik
[root@rear-cdm ~]# cd rubrik
[root@rear-cdm rubrik]# curl -kLOJ https://[redacted]/connector/rubrik-agent.x86_64.rpm
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 69.2M    0 69.2M    0     0  13.7M      0 --:--:--  0:00:05 --:--:-- 14.7M
```

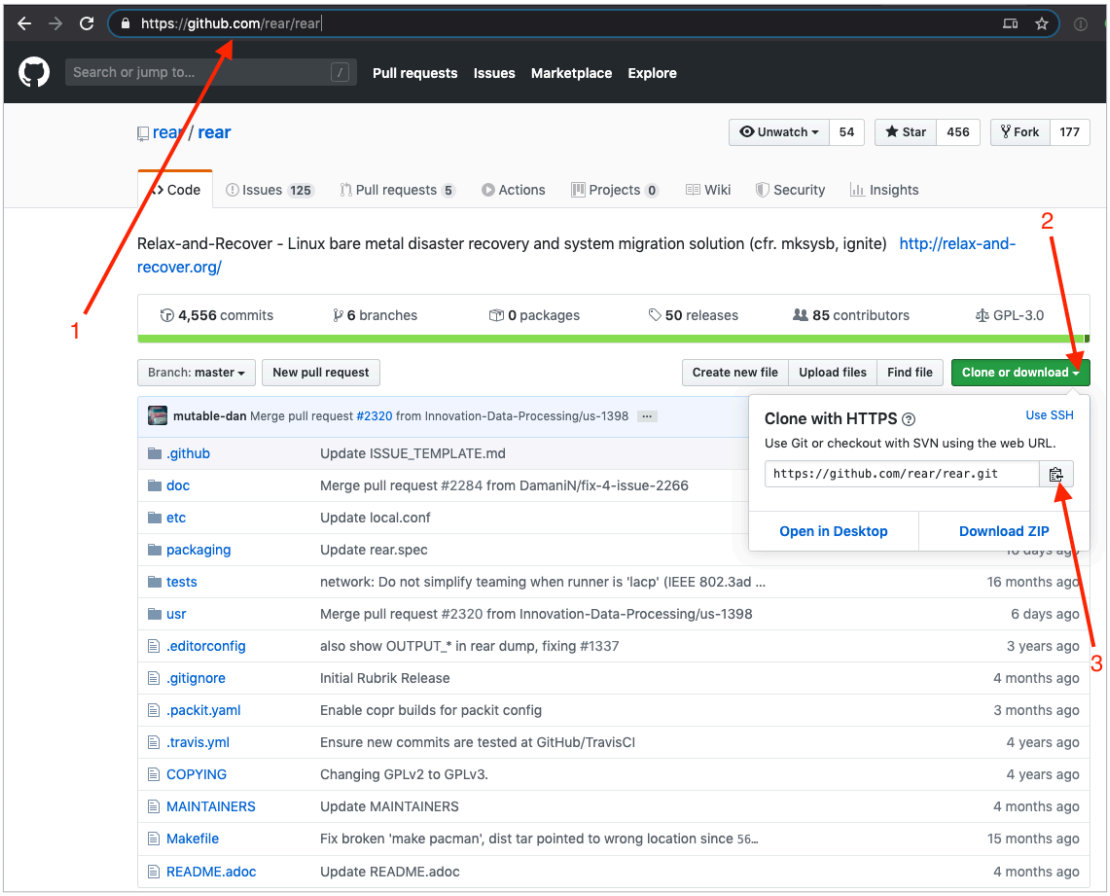
   Run `rpm -ihv rubrik-agent.x86_64.rpm`

```
[root@rear-cdm rubrik]# rpm -ihv rubrik-agent.x86_64.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:rubrik-agent-5.1.2.8188-1.0      ################################# [100%]
Generate uuid
Generate credentials
/etc/rubrik
Generating RSA private key, 2048 bit long modulus
.............+++
......+++
e is 65537 (0x10001)
Using configuration from /tmp/tmp.VW9l73jpt6/conf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'[redacted] agent.rubrik.local'
countryName           :PRINTABLE:'US'
stateOrProvinceName   :ASN.1 12:'California'
localityName          :ASN.1 12:'Palo Alto'
organizationName      :ASN.1 12:'Rubrik, Inc.'
Certificate is to be certified until Sep 11 05:49:12 2047 GMT (10000 days)

Write out database with 1 new entries
Data Base Updated
Status daemon: bootstrap_agent_main [DOWN]
Starting daemon: bootstrap_agent_main
I0424 22:49:13.049520  1836 common.cpp:23] Binary built on Mar 10 2020 at 12:17:16
                                                           [  OK  ]
Status daemon: bootstrap_agent_main [UP]
Status daemon: backup_agent_main [DOWN]
Starting daemon: backup_agent_main
I0424 22:49:13.128242  1870 common.cpp:23] Binary built on Mar 10 2020 at 12:17:16
I0424 22:49:13.128319  1870 thrift_util.cpp:129] Skipping FIPS initialization
I0424 22:49:13.129815  1870 thrift_util.cpp:147] Initialized Thrift SSL
                                                           [  OK  ]
Status daemon: backup_agent_main [UP]
```

9 https://relax-and-recover.org/

2. Clone the Rear Project

   Get the URL for the project:



   Run `git clone` https://github.com/rear/rear.git

```
[root@rear-cdm ~]# git clone https://github.com/rear/rear.git
Cloning into 'rear'...
remote: Enumerating objects: 191, done.
remote: Counting objects: 100% (191/191), done.
remote: Compressing objects: 100% (136/136), done.
remote: Total 46134 (delta 80), reused 47 (delta 17), pack-reused 45943
Receiving objects: 100% (46134/46134), 8.57 MiB | 8.74 MiB/s, done.
Resolving deltas: 100% (22336/22336), done.
```

3. Install Relax-and-Recover

   Run `cd rear`

   Run `make install`

```
[root@rear-cdm rear]# make install
== Validating scripts and configuration ==
find etc/ usr/share/rear/conf/ -name '*.conf' | xargs -n 1 bash -n
bash -n usr/sbin/rear
find . -name '*.sh' | xargs -n 1 bash -O extglob -O nullglob -n
find usr/share/rear -name '*.sh' | grep -v -E '(lib|skel|conf)' | while read FILE ; do \
        num=$(echo ${FILE##*/} | cut -c1-3); \
        if [[ "$num" = "000" || "$num" = "999" ]] ; then \
                echo "ERROR: script $FILE may not start with $num"; \
                exit 1; \
        else \
                if $( grep '[_[:alpha:]]' <<< $num >/dev/null 2>&1 ) ; then \
                        echo "ERROR: script $FILE must start with 3 digits"; \
                        exit 1; \
                fi; \
        fi; \
done
== Prepare manual ==
make -C doc man
make[1]: Entering directory `/root/rear/doc'
asciidoc -b docbook -d manpage rear.8.adoc
xmlto man rear.8.xml
Note: Writing rear.8
rm rear.8.xml
make[1]: Leaving directory `/root/rear/doc'
== Installing configuration ==
install -d -m0700 /etc/rear/
install -d -m0700 /etc/rear/cert/
[[ ! -e /etc/rear/local.conf ]] && \
        install -Dp -m0600 etc/rear/local.conf /etc/rear/local.conf
make: [install-config] Error 1 (ignored)
[[ ! -e /etc/rear/os.conf && -e etc/rear/os.conf ]] && \
        install -Dp -m0600 etc/rear/os.conf /etc/rear/os.conf
make: [install-config] Error 1 (ignored)
find /etc/rear/ -name '.gitignore' -exec rm -rf {} \; &>/dev/null
== Installing binary ==
install -Dp -m0755 usr/sbin/rear /usr/sbin/rear
sed -i -e 's,^CONFIG_DIR=.*,CONFIG_DIR="/etc/rear",' \
        -e 's,^SHARE_DIR=.*,SHARE_DIR="/usr/share/rear",' \
        -e 's,^VAR_DIR=.*,VAR_DIR="/var/lib/rear",' \
        /usr/sbin/rear
== Installing scripts ==
install -d -m0755 /usr/share/rear/
cp -a usr/share/rear/. /usr/share/rear/
find /usr/share/rear/ -name '.gitignore' -exec rm -rf {} \; &>/dev/null
== Installing working directory ==
install -d -m0755 /var/lib/rear/
install -d -m0755 /var/log/rear/
== Installing documentation ==
make -C doc install
make[1]: Entering directory `/root/rear/doc'
install -Dp -m0644 rear.8 /usr/share/man/man8/rear.8
make[1]: Leaving directory `/root/rear/doc'
sed -i -e 's,/etc,/etc,' \
        -e 's,/usr/sbin,/usr/sbin,' \
        -e 's,/usr/share,/usr/share,' \
        -e 's,/usr/share/doc/packages,/usr/share/doc,' \
        /usr/share/man/man8/rear.8
```

CONFIGURATION

1. Edit /etc/rear/local.conf and enter:

```
# Sets output to an be an ISO file
OUTPUT=ISO

# Specifies CDM as the backup and recovery application
BACKUP=CDM
```

```
[root@rear-cdm ~]# cat /etc/rear/local.conf
# Default is to create Relax-and-Recover rescue media as ISO image
# set OUTPUT to change that
# set BACKUP to activate an automated (backup and) restore of your data
# Possible configuration values can be found in /usr/share/rear/conf/default.conf
#
# This file (local.conf) is intended for manual configuration. For configuration
# through packages and other automated means we recommend creating a new
# file named site.conf next to this file and to leave the local.conf as it is.
# Our packages will never ship with a site.conf.
OUTPUT=ISO
BACKUP=CDM
```

2. Optionally redirect the ISO file to a directory other than /var/lib/rear/output.

```
# Default "local" ISO directory (usually /var/lib/rear/output). However, to avoid
# duplicateISO images when also using the OUTPUT_URL variable with a file syntax, it is
# then better only to use ISO_DIR. Keep in mind that ISO_DIR works only with an absolute
# directory path and does not replace OUTPUT_URL which supports the NETFS syntax
# (to copy the ISO image across the network).
ISO_DIR=$VAR_DIR/output
```

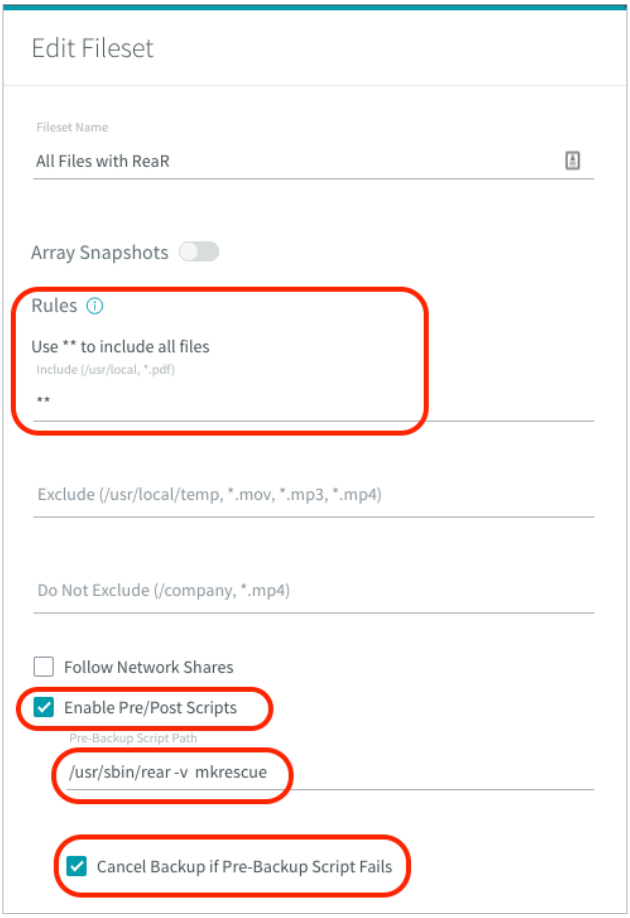
3. To have Rubrik CDM create a create ISO during each backup, create or configure a fileset backup with the following properties:

    a. Include at least the root (/) filesystem

    b. Enable Pre/Post scripts.

    c. Add `/usr/sbin/rear -v mkrescue` as the ***Pre-Backup script path***.

    d. It is highly recommended to select ***Cancel Backup if Pre-Backup Script Fails***. This will ensure that notifications are sent if the `rear -v mkrescue` command fails, instead of the backup failing silently.



BACKUP

1. Before running scheduled backups using Relax-and-Recover, first make sure that an ISO can be made using the `rear -v mkrescue` command. By default this command will create an ISO file called `/var/lib/rear/output/rear-<hostname>.iso`. If the `rear -v mkrescue` command fails, errors can be found in `/var/log/rear/rear-<hostname>.log`.

NOTE: See the Troubleshooting section if problems occur. Also refer to the Relax-and-Recover Troubleshooting[10] page for other troubleshooting tips.

10 https://github.com/rear/rear/blob/master/doc/user-guide/08-troubleshooting.adoc

```
[root@rear-cdm ~]# rear -v mkrescue
Relax-and-Recover 2.5 / Git
Running rear mkrescue (PID 13975)
Using log file: /var/log/rear/rear-rear-cdm.log
Using autodetected kernel '/boot/vmlinuz-3.10.0-1062.4.1.el7.x86_64' as kernel in
 the recovery system
Creating disk layout
Overwriting existing disk layout file /var/lib/rear/layout/disklayout.conf
Using guessed bootloader 'GRUB' (found in first bytes on /dev/sda)
Verifying that the entries in /var/lib/rear/layout/disklayout.conf are correct ..
.
Creating recovery system root filesystem skeleton layout
To log into the recovery system via ssh set up /root/.ssh/authorized_keys or spec
ify SSH_ROOT_PASSWORD
Copying logfile /var/log/rear/rear-rear-cdm.log into initramfs as '/tmp/rear-rear
-cdm-partial-2020-04-24T23:06:57-0700.log'
Copying files and directories
Copying binaries and libraries
Copying all kernel modules in /lib/modules/3.10.0-1062.4.1.el7.x86_64 (MODULES co
ntains 'all_modules')
Copying all files in /lib*/firmware/
Broken symlink '/usr/lib/modules/3.10.0-1062.4.1.el7.x86_64/build' in recovery sy
stem because 'readlink' cannot determine its link target
Broken symlink '/usr/lib/modules/3.10.0-1062.4.1.el7.x86_64/source' in recovery s
ystem because 'readlink' cannot determine its link target
Testing that the recovery system in /tmp/rear.oilxU7tJZrkVOod/rootfs contains a u
sable system
Creating recovery/rescue system initramfs/initrd initrd.cgz with gzip default com
pression
Created initrd.cgz with gzip default compression (399522682 bytes) in 51 seconds
Making ISO image
Wrote ISO image: /var/lib/rear/output/rear-rear-cdm.iso (391M)
Exiting rear mkrescue (PID 13975) and its descendant processes ...
Running exit tasks
```

2. Once the `rear -v mkrescue` command runs successfully scheduled backups of the system can be run.

RECOVERY

Currently the Rubrik CDM integration with Relax-and-Recover supports recovering to the original server and recovering to a new server. It also supports Linux systems with static IP addresses or DHCP IP addresses. Only interactive recovery is supported at this time.

PREREQUISITES

1. **IP Address assignment**

   Before starting the recovery process verify how the IP addresses will be handled on the recovery system. If the original Linux system used static IP addresses, the recovery system will boot with this same IP address. If the original Linux system is being replaced and is down this may be fine. However, if the original Linux system is still running with the same static IP address the recovery system will need to be booted in isolation at first. While in isolation there will be an opportunity to change the static IP address to something new.

   If DHCP addresses were used on the original Linux system a new IP address will be assigned to the recovery system. No IP address conflict should occur.

2. **Boot image**

   A copy of the boot image that was created by Relax-and-Recover will be needed to execute the steps below.

3. **Hardware**

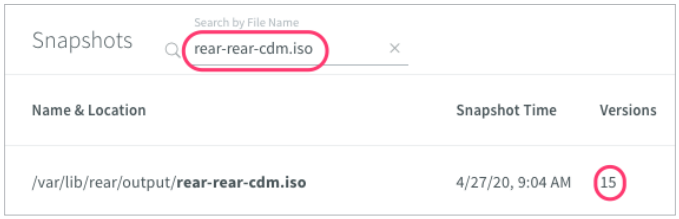
Recovery to dissimilar hardware is supported. The disk layout and capacities must match or exceed the original Linux system though. See the Relax-and-Recover Layout configuration[11] page for more details.

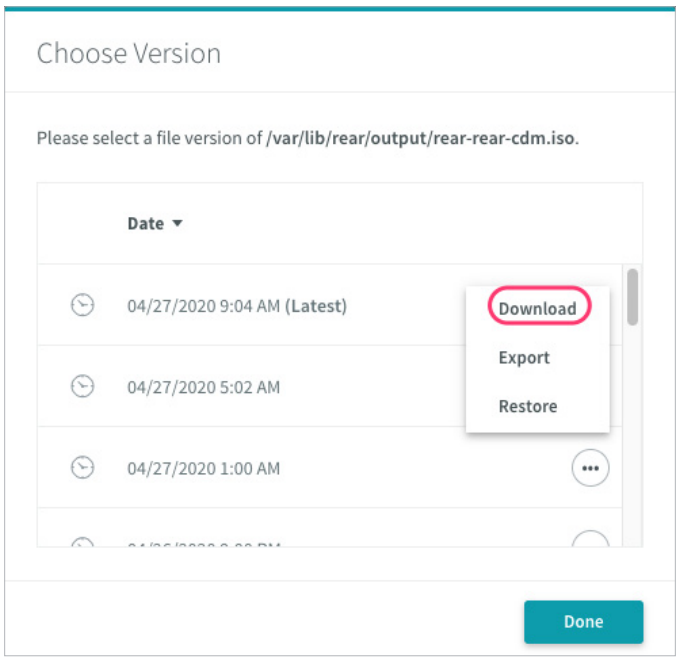
BOOTING THE RECOVERY SYSTEM

1. To begin the recovery process first obtain a copy of the recovery image.

   a. Typically this will be `rear-<hostname>.iso` which was saved in `/var/lib/rear/output/` on the protected system unless the default options have been changed.

   b. This file can be downloaded from a Rubrik fileset backup if it was protected as part of the filesystem data.



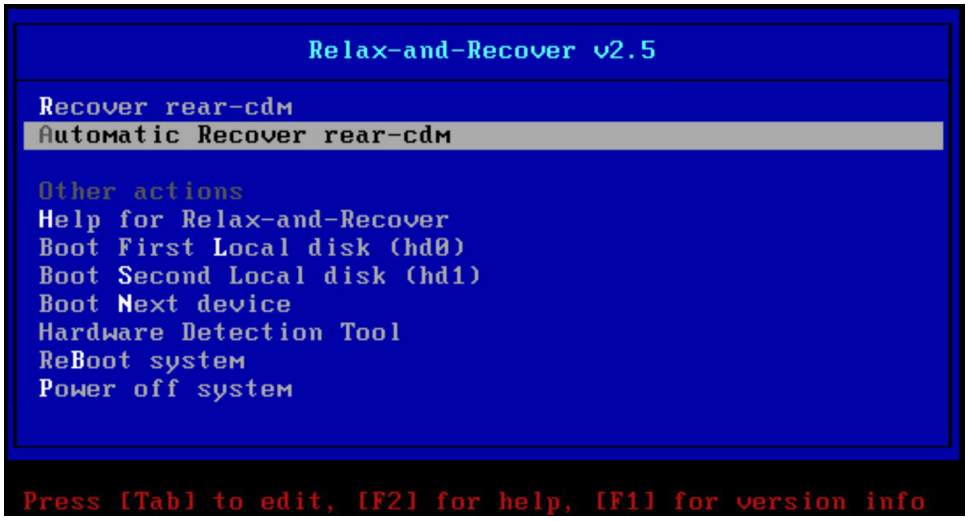


   c. This file may have been stored externally as well.

2. Burn the `rear-<hostname>.iso` file to a bootable media that is compatible with the recovery system.

3. Boot the recovery system using the bootable media that was created from the `rear-<hostname>.iso` file.

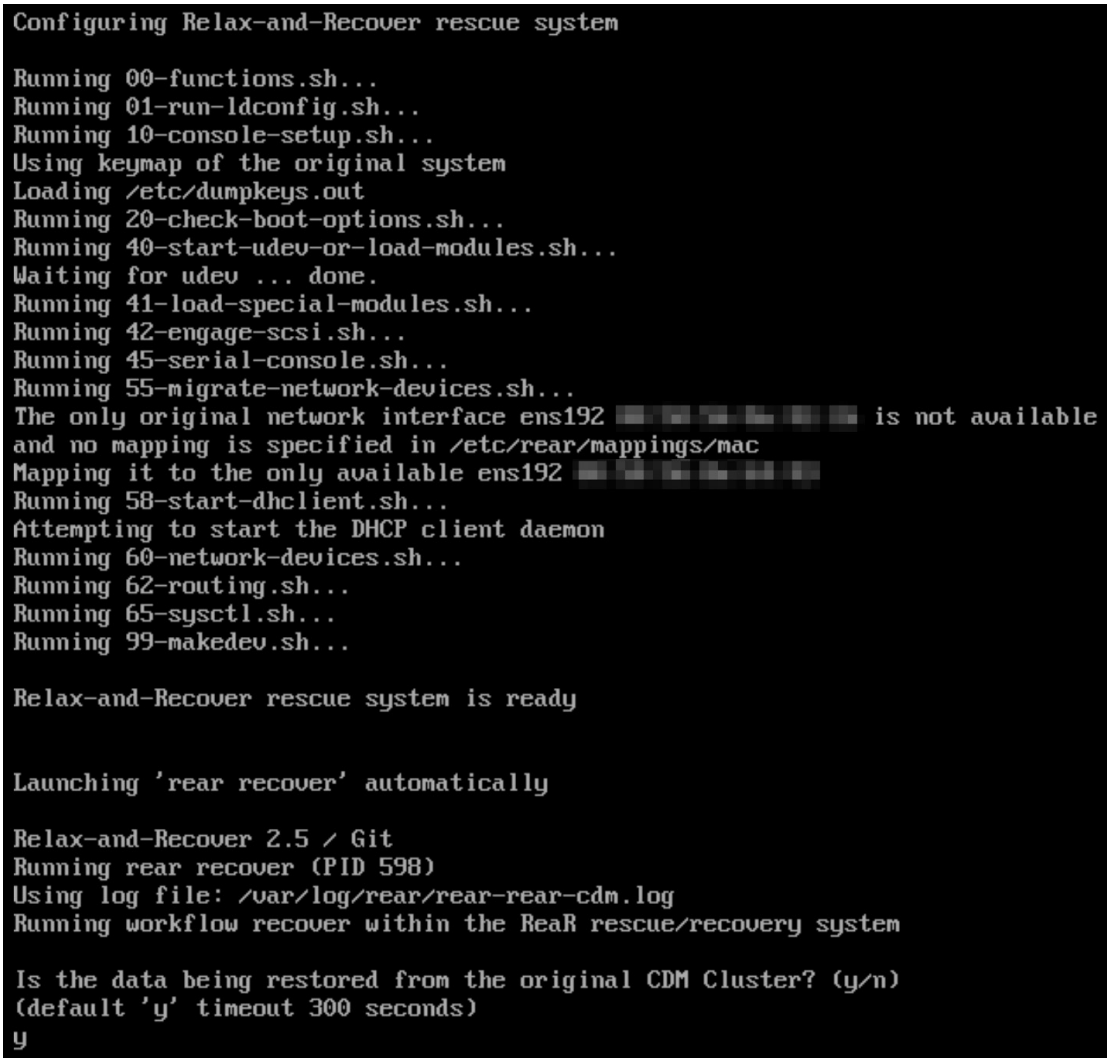
11 https://github.com/rear/rear/blob/master/doc/user-guide/06-layout-configuration.adoc



4. Select ***Automatic Recover <hostname>*** from the Relax-and-Recover boot menu.
    a. This option automatically logs into the recovery system and runs rear recover.
    b. Selecting Recover <hostname> will present a login prompt.
        i. Enter any username (usually `root`).
        ii. This will present a command prompt. Run any commands needed before starting recovery.
            1. In some cases stopping the Linux firewall is needed in this step.
        iii. Run `rear recover`.
5. Recovering from the same Rubrik CDM cluster as the backup was performed is supported. Recovering from a Rubrik CDM cluster where the backup was replicated too is also supported. Recovering from the replica is useful for disaster recovery scenarios or migration where recovery to another datacenter is required.

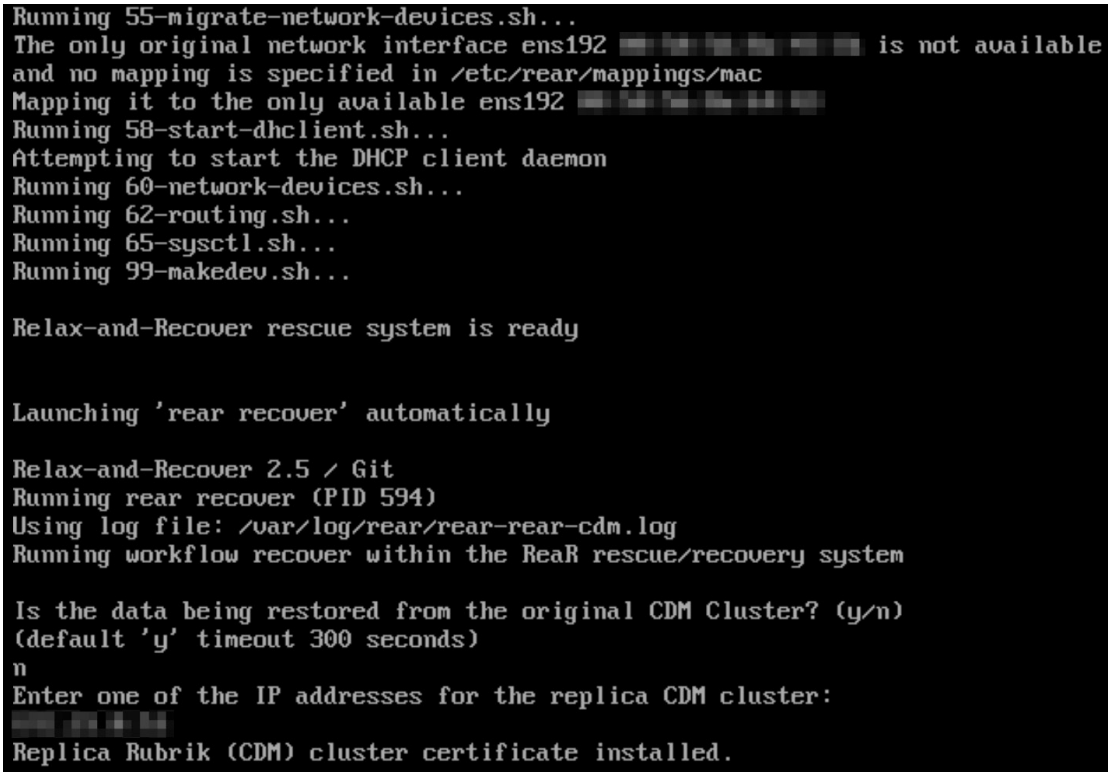
    Indicate if you are recovering from the same Rubrik CDM cluster or a different one.

    a. If recovering from the same Rubrik cluster enter '`y`'.



b. If recovering from a different Rubrik CDM cluster enter 'n'.

   i. Enter the IP address for one of the Rubrik CDM nodes on the new cluster. This will cause Relax-and-Recover to download the RBS client from the cluster and authorize the recovery system to restore from the new cluster.



6. Indicate if the same IP address is being used on the recovery server as on the original Linux server.

    a. Enter '**y**' if the IP address of the recovery system is the same as the original Linux system.

    b. Enter '**n**' if the IP address of the recovery system is different from the original Linux system. The recovery system's unique Rubrik ID will be regenerated so that it does not conflict with the original Linux host.

```
Found the following IP addresses on this system:
    inet [redacted]
    inet6 [redacted]
    inet6 [redacted]
    inet6 [redacted]

Does this client have the same IP address as the original? (y/n)
(default 'y' timeout 300 seconds)
n
Rubrik (CDM) RBS agent now has new UUID.
Rubrik (CDM) RBS agent started.
```

7. Follow the prompts to properly repartition the recovery system's disks. If failures occur on this step see the Relax-and-Recover Layout configuration[12] and the Relax-and-Recover Troubleshooting[13] pages for troubleshooting tips.

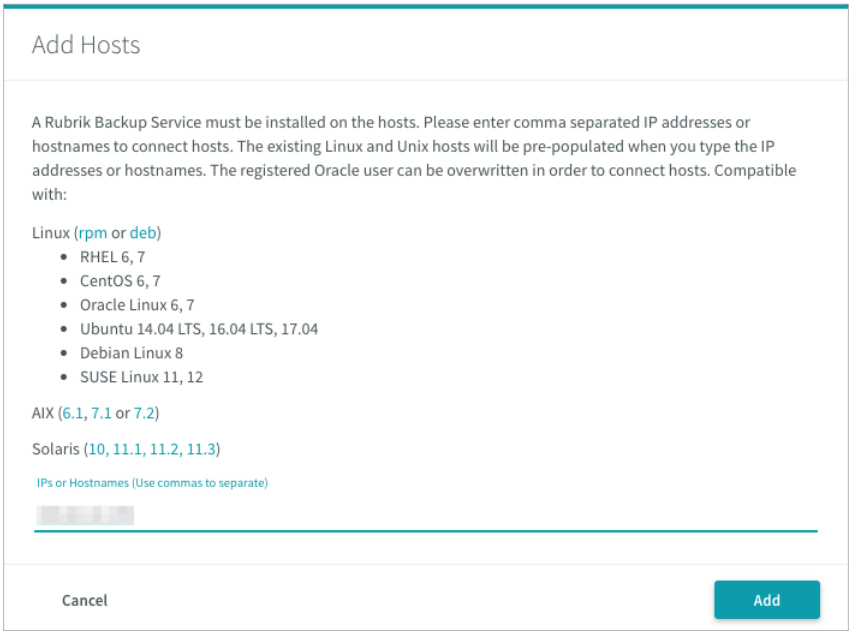
```
Comparing disks
Device sda has expected (same) size 17179869184 bytes (will be used for 'recover')
Disk configuration looks identical
Proceed with 'recover' (yes) otherwise manual disk layout configuration is enforced
(default 'yes' timeout 30 seconds)
y
User confirmed to proceed with 'recover'
Start system layout restoration.
Disk '/dev/sda': creating 'msdos' partition table
Disk '/dev/sda': creating partition number 1 with name 'primary'
Disk '/dev/sda': creating partition number 2 with name 'primary'
Creating LVM PV /dev/sda2
Restoring LVM VG 'centos'
Sleeping 3 seconds to let udev or systemd-udevd create their devices...
Creating filesystem of type xfs with mount point / on /dev/mapper/centos-root.
Mounting filesystem /
Creating filesystem of type xfs with mount point /boot on /dev/sda1.
Mounting filesystem /boot
Creating swap on /dev/mapper/centos-swap
Disk layout created.
Please start the restore process on the Rubrik (CDM) cluster.

Register the appropriate IP address from this list with Rubrik (CDM):
    inet [redacted]
    inet6 [redacted]
    inet6 [redacted]
    inet6 [redacted]

Make sure all required data is restored to /mnt/local .

Next type 'exit' to continue the recovery.
Info: You can check the recovery process i.e. with the command 'df'.

rear>
```

8. When the ***rear>*** prompt appears, go to the Rubrik UI.

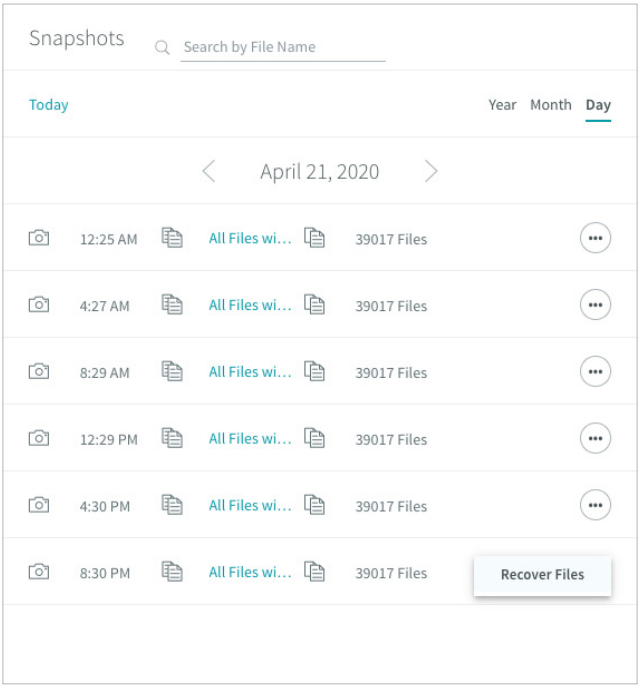
12 https://github.com/rear/rear/blob/master/doc/user-guide/06-layout-configuration.adoc
13 https://github.com/rear/rear/blob/master/doc/user-guide/08-troubleshooting.adoc

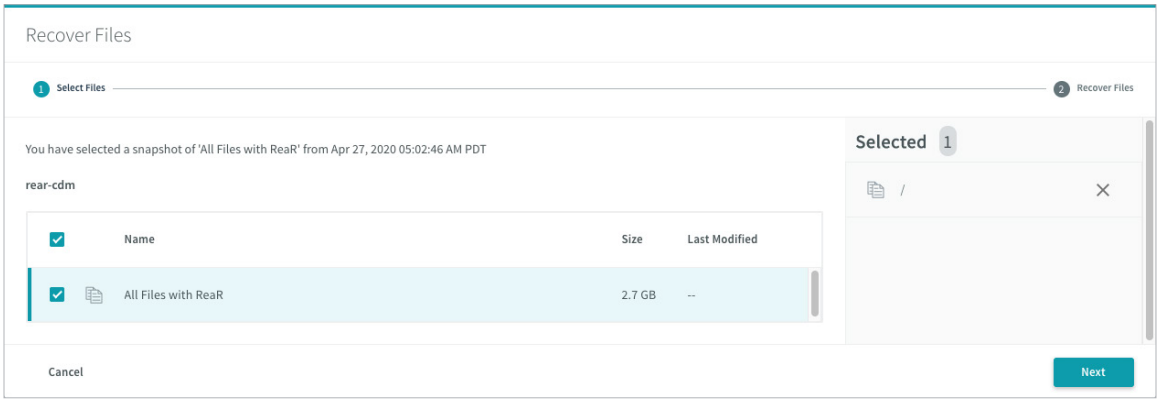
RECOVER DATA

1. If the recovery system is using a different IP address than the original Linux system it must be registered in Rubrik CDM. Add a new Linux host using the Rubrik CDM GUI. Use the IP address of the recovery system if it is not in DNS or it's hostname if it is in DNS. There is no need to download and install the RBS client. It was already included in the Relax-and-Recover boot image.



2. Perform a Recover Files of at least the root file system for the original Linux system. All of the data for the Linux system can also be recovered in this step. The recovery needs to be redirected to `/mnt/local` as this is where the disks were mounted on the recovery system. The `/` (root) file system on the recovery system is from the boot media.



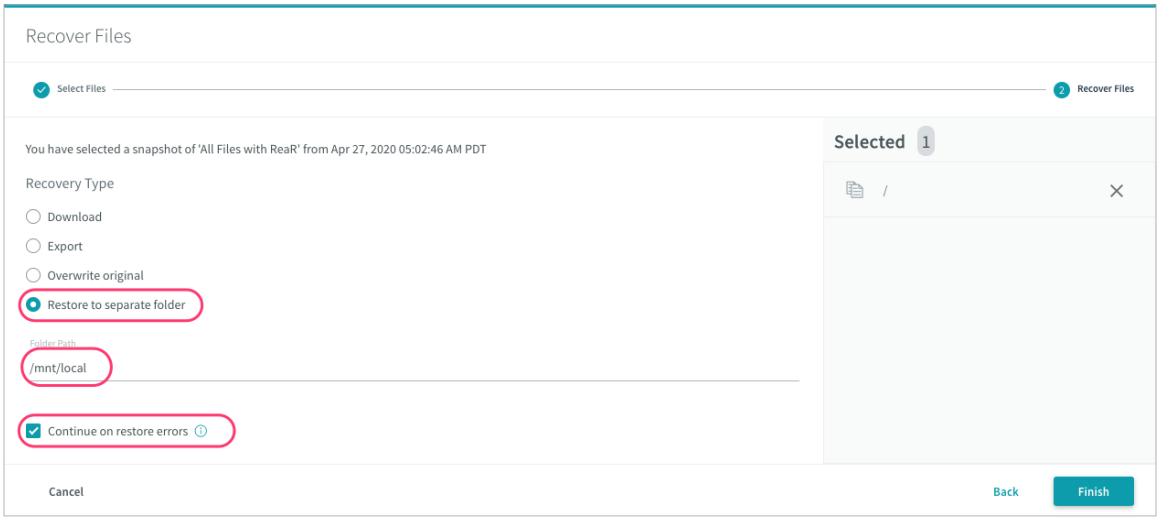


a. If the recovery system is using the IP address of the original Linux system do the following:

   i. Select ***Restore to separate folder***.

   ii. Enter `/mnt/local` for ***Export Path***.

   iii. Select ***Continue on restore errors***.



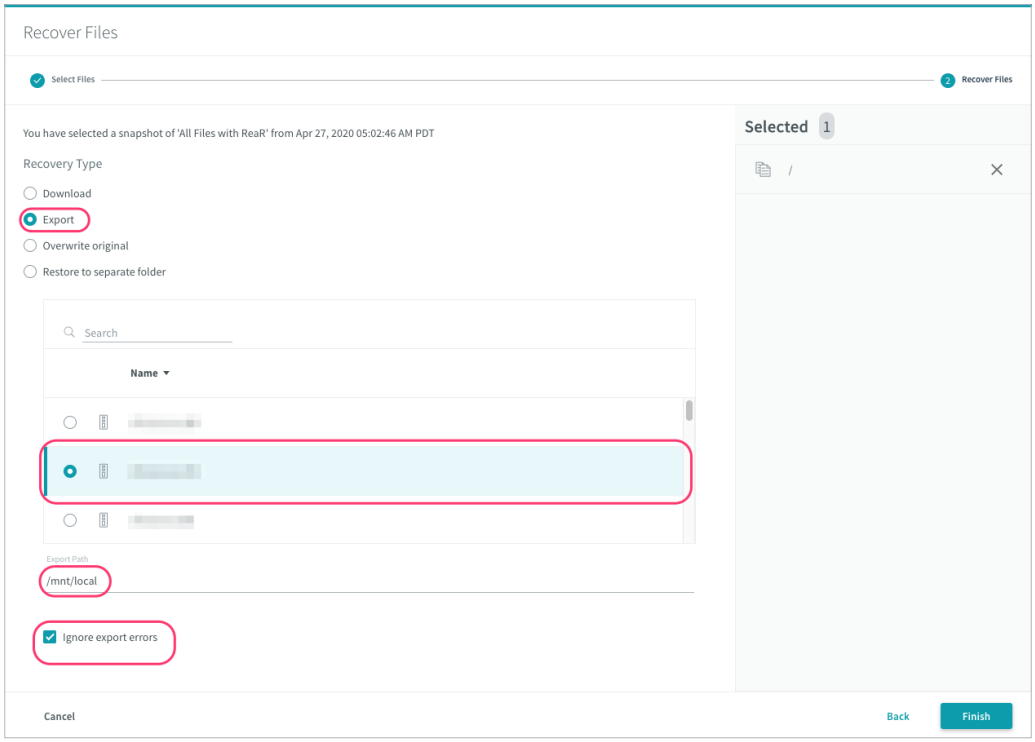
b. If the recovery recovery system is using a different IP address than the original Linux system do the following:

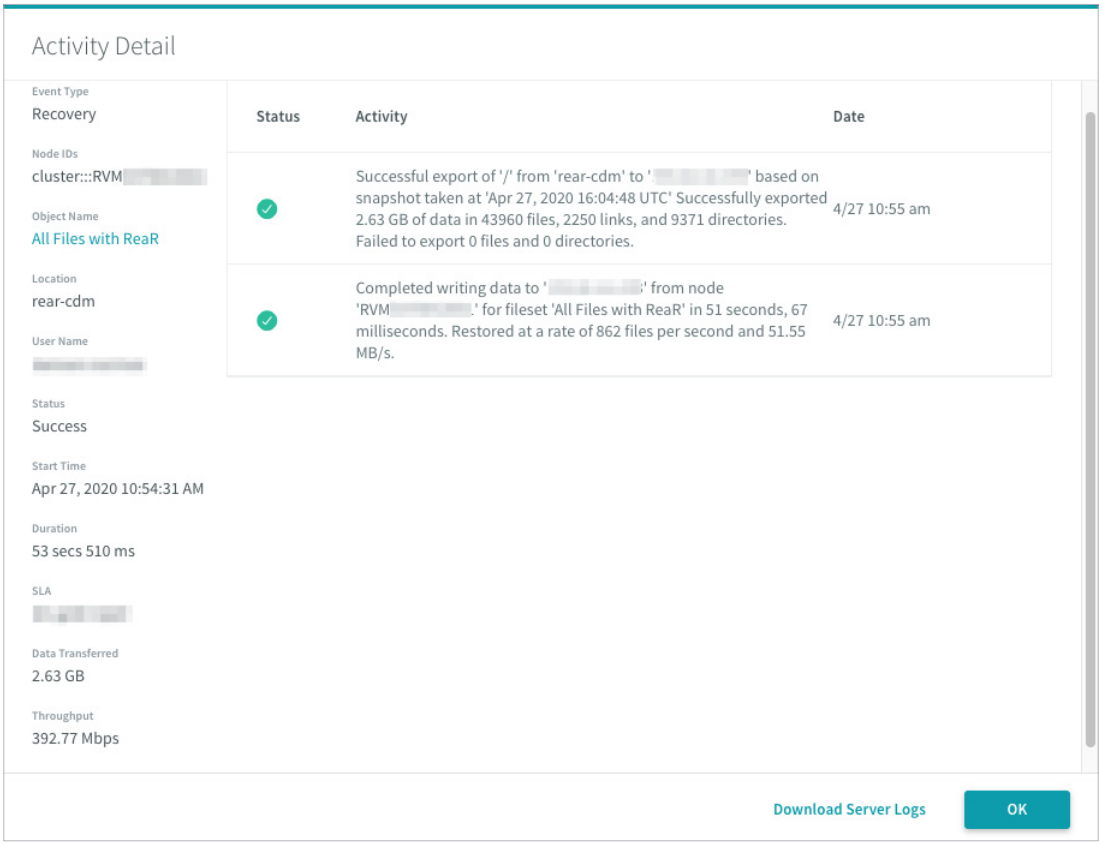
   i. Select ***Export***.

   ii. Select the hostname or IP address of the recovery system.

   iii. Enter `/mnt/local` for ***Export Path***

   iv. Select ***Ignore export errors***

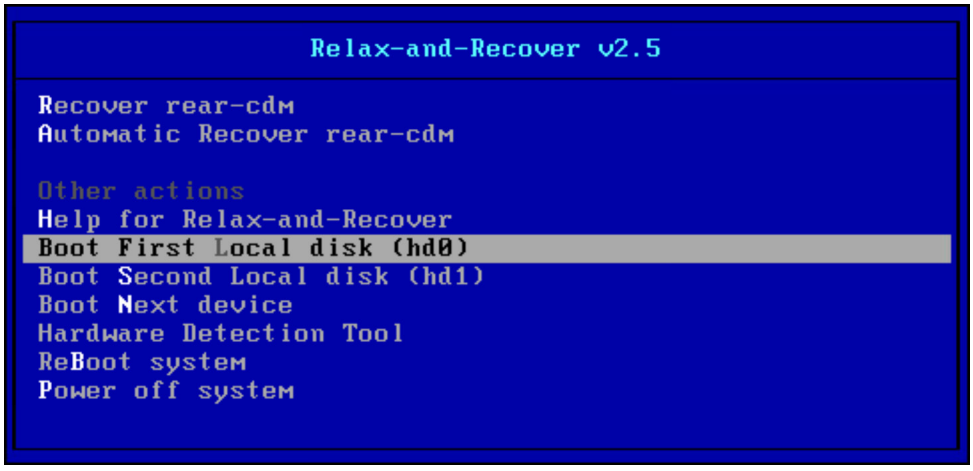

Recover Files
Select Files
2 Recover Files
You have selected a snapshot of 'All Files with ReaR' from Apr 27, 2020 05:02:46 AM PDT
Recovery Type
Download
Export
Overwrite original
Restore to separate folder
Search
Name
Export Path
/mnt/local
Ignore export errors
Selected 1
/
Cancel
Back
Finish


Activity Detail
Event Type
Recovery
Node IDs
cluster:::RVM
Object Name
All Files with ReaR
Location
rear-cdm
User Name
Status
Success
Start Time
Apr 27, 2020 10:54:31 AM
Duration
53 secs 510 ms
SLA
Data Transferred
2.63 GB
Throughput
392.77 Mbps
Status
Activity
Date
Successful export of '/' from 'rear-cdm' to ' ' based on snapshot taken at 'Apr 27, 2020 16:04:48 UTC' Successfully exported 2.63 GB of data in 43960 files, 2250 links, and 9371 directories. Failed to export 0 files and 0 directories.
4/27 10:55 am
Completed writing data to ' ' from node 'RVM ' for fileset 'All Files with ReaR' in 51 seconds, 67 milliseconds. Restored at a rate of 862 files per second and 51.55 MB/s.
4/27 10:55 am
Download Server Logs
OK

3. Once Rubrik CDM finishes recovering the data return to the recovery system and type `exit` at the ***rear>*** prompt.
4. Enter '`y`' at the restore completion prompt question
5. Relax-and-Recover will do some housekeeping like fixing the root file system permissions and setting up the bootloader.
6. Once the prompt returns, gracefully reboot the system by selecting '`3`'.

```
rear> exit
Has the restore been completed and are you ready to continue the recovery? y
exit
Recreating directories (with permissions) from /var/lib/rear/recovery/directories_permissions_owner_group
Migrating disk-by-id mappings in certain restored files in /mnt/local to current disk-by-id mappings ...
Migrating network configuration files according to the mapping files ...
Running mkinitrd...
Updated initrd with new drivers for kernel 3.10.0-1062.4.1.el7.x86_64.
Running mkinitrd...
Updated initrd with new drivers for kernel 3.10.0-1062.el7.x86_64.
Skip installing GRUB Legacy boot loader because GRUB 2 is installed (grub-probe or grub2-probe exist).
Installing GRUB2 boot loader...
Determining where to install GRUB2 (no GRUB2_INSTALL_DEVICES specified)
Found possible boot disk /dev/sda - installing GRUB2 there
Finished 'recover'. The target system is mounted at '/mnt/local'.
Exiting rear recover (PID 595) and its descendant processes ...
Running exit tasks

'rear recover' finished successfully

1) View Relax-and-Recover log file(s)  3) Reboot
2) Go to Relax-and-Recover shell
Select what to do 3_
```

7. If the Relax-and-Recover boot loader starts, select the correct hard drive to boot from.



8. Allow the system to boot normally and it will be restored.
9. Eject the boot media from the restored Linux system.

KNOWN ISSUES

The following are known to be issues at the time of this writing:

- Until Relax-and-Recover v2.6 has been released and downstream package installers created follow the instructions in note 3 below to install rear from the https://github.com/rear/rear project page[14] using `make install`.
    - Package installers can be made from the master branch by following these instructions[15].
    - The `make install` process may fail with missing packages on a given system. Install the missing packages and try again. For example a basic Ubuntu installation needs to also have the `isolinux`, `binutils`, `genisoimage` and `syslinux` packages installed.
- Recovery via IPv6 is not yet supported.
- Automatic recovery from replica CDM cluster is not supported
- Rubrik CDM may take some time to recognize that the IP address has moved from one system to another. When restoring using the same IP, give Rubrik CDM up to 10 minutes to recognize that the agent is running on another machine. This usually comes up during testing when the original machine is shutdown but not being restored to.
- Recovery from a Rubirk CDM replication target cluster is only supported with CDM v4.2.1 and higher.
- Care must be taken with SUSE systems on DHCP. They tend to request the same IP as the original host. If this is not the desired behavior the recovery system should be booted in isolation and reconfigured after logging in with the ***Recover <hostname>*** boot option.
- If multiple restores are performed using the same temporary IP, the temporary IP must first be deleted from Rubirk CDM under ***Servers & Apps -> Linux and Unix Servers*** and re-added upon each reuse.
- Relax-and-Recover's ***ldd*** check of other binaries or libraries may result in libraries not being found. This can generally be worked around by adding the path to those libraries to the `LD_LIBRARY_PATH` variable in `/etc/rear/local.conf`. Do this by adding the following line in `/etc/rear/local.conf`:

  ```
  export LD_LIBRARY_PATH-"$LD_LIBRARY_PATH:<path>"
  ```

    - To make CentoOS v7.7 work the following line was needed:

      ```
      export LD_LIBRARY_PATH="$LD_LIBRARY_PATH:/usr/lib64/bind9-export"
      ```

    - To make CentOS v8.0 work the following line was needed:

      ```
      export LD_LIBRARY_PATH="$LD_LIBRARY_PATH:/usr/lib64/bind9-export: \
      /usr/lib64/eog:/usr/lib64/python3.6/site-packages:/usr/lib64/samba: \
      /usr/lib64/firefox"
      ```

- Rear may not set the static IP on a system when the ISO boots. To workaround this set the following in `/etc/rear/local.conf`:

14 https://github.com/rear/rear
15 http://relax-and-recover.org/documentation/installation

```
# Specify networking commands to reset static IP if the ReaR ISO doesn't boot with a
# static IP address
NETWORKING_PREPARATION_COMMANDS=( 'ip addr add <STATIC_IP_ADDRESS> dev eth0' \
'ip link set dev eth0 up' \
'route add -net <LOCAL_SUBNET>>/<LOCAL_SUBNET_PREFIX/MASK> eth0' \
'route add default gw <DEFAULT_GATEWAY>' 'return' )
```

See https://github.com/rear/rear/blob/master/usr/share/rear/conf/default.conf[16] for more details on these options.

- When using the Rubrik CDM integration on virtual systems with 1GB of RAM, the recovery system may experience a kernel panic during boot. This can be worked around by increasing the RAM to 2GB.

TROUBLESHOOTING

If Relax-and-Recover is failing use the following troubleshooting tips to isolate the problem:

- Verify that Relax-and-Recover will recover the Linux system without using the CDM backup and restore method. Most errors are due to configuration with Relax-and-Recover itself and not Rubrik CDM. Use the default Relax-and-Recover backup and restore method to test this.
- Follow the OS specific configuration guides as mentioned at the beginning of this document.
- Example configurations for specific operating systems can be found in these links:
  - Red Hat

    https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/6/html/deployment_guide/ch-relax-and-recover_rear
  - Ubuntu

    http://manpages.ubuntu.com/manpages/disco/en/man8/rear.8.html
  - SUSE

    https://en.opensuse.org/SDB:Disaster_Recovery
    https://documentation.suse.com/sle-ha/15-SP1/html/SLE-HA-all/cha-ha-rear.htm
  - Generic

    https://github.com/rear/rear

NOTE: Ignore any instructions to configure external storage like NFS, CIFS/SMB or ftp. Also ignore any instructions to configure a specific backup method. This will be taken care of in the next steps.

NOTE: Ignore any instructions to schedule ReaR to run via the host based scheduler (cron). Rubrik CDM will run ReaR via a pre-script in the fileset. If this is not preferred ReaR can be scheduled on the host, however, the ISOs created may not be in sync with the backups.

16 https://github.com/rear/rear/blob/master/usr/share/rear/conf/default.conf

VERSION HISTORY

Version	Date	Summary of Changes
1.0	May 2020	Initial Release

rubrik

Global HQ
1001 Page Mill Rd., Building 2
Palo Alto, CA 94304
United States

1-844-4RUBRIK
inquiries@rubrik.com
www.rubrik.com

Rubrik, the Multi-Cloud Data Control™ Company, enables enterprises to maximize value from data that is increasingly fragmented across data centers and clouds. Rubrik delivers a single, policy-driven platform for data recovery, governance, compliance, and cloud mobility. For more information, visit www.rubrik.com and follow @rubrikInc on Twitter. © 2020 Rubrik. Rubrik is a registered trademark of Rubrik, Inc. Other marks may be trademarks of their respective owners.

20200513_v1