Bare Metal Recovery for Linux

Bacula Systems Documentation

Contents

Contents

1 Executive Summary

This **Bacula Systems** User Guide explains how Bare Metal Recovery of Linux systems can be done with the LinuxBMR toolkit **Bacula Systems** provides.

The reader of this paper is expected to have a good understanding of Bacula in general, i. e. working with the configuration files and `bconsole` should be expected. Also, general system administration knowledge as required for the backed up environment is assumed. The terminology used with **Bacula** needs also to be known.

To make sure you understand the current state of this **Bacula Systems** LinuxBMR toolkit, please read the Release Notes.

As any component that is to become part of a Disaster Recovery solution, the LinuxBMR toolkit needs to be extensively tested on any hardware it will be used on. Test results, site- or system-specific procedures, and essential configuration should be documented in a Disaster Recovery manual.

2 Introduction to BMR

Bare Metal Recovery or BMR for short, is the term usually used to describe restoration of a complete operating system to new hardware without actually going through the operating system's installation procedure.

The main goal is to get from a new, empty machine (bare metal) to a fully functional operating system including all applications and data as quickly as possible. The real challenge is to set up the disk subsystem of the new machine in a way that closely resembles the original disk layout where data was backed up from, and to ensure the recovered operating system can be booted.

If the new hardware is of a different type than the machine the installed software was backed up from originally, BMR will very often not be possible. To be useful, BMR procedures need to take that into account. It is necessary to ensure that the BMR hardware being restored to is compatible with the source system.

In the example in table *BMR source and target examples*, some common situations for Linux users today are outlined. In particular it is important to understand that software written for 32-bit x86-CPUs can be run on 64-bit x86-CPUs, but not vice versa, and that software written to run on one CPU family, like Sparc or x86, will not run on any other CPU family.

Table 1: BMR source and target examples

Origin	New Hardware	Problems	BMR possible?
i386	i386	None	Yes
i386_64	i386_64	None	Yes
i386	i386_64	Existing software not optimized for hardware	Yes
i386_64	i386	64-bit software will not run on 32-bit hardware	No
i386	SPARC	Completely different hardware	No

3 Overview of Bacula Enterprise LinuxBMR

The **Bacula Enterprise** LinuxBMR toolkit consists of several parts, each designed to fulfill one of the tasks in a BMR-enabled backup and recovery environment. In addition, some configuration of **Bacula** is required to ensure the software is actually used. Accordingly, well-defined procedures of how to install, deploy and use the BMR backup tools as well as how to do a BMR exist.

We will describe the software, the required configuration, and how to use it together in more detail after giving an overview first.

The overall approach to allow BMR is to collect essential system information during the backup itself, and to use that information during a BMR session to set up the new hardware to allow immediate reuse of the restored system, applications and data after the BMR is finished.

For the actual backup, after configuring the backup jobs accordingly, no additional effort is required.

During recovery, the essential disk partitioning data is pulled from the existing backups, applied to the BMR machine so that the target system is set up as needed, and then the actual restore is initiated. These steps, plus selection of what client to restore to, and what point in time to recover from is all handled by an easy to use GUI interface which is part of **Bacula Systems**' LinuxBMR recovery image.

The LinuxBMR recovery image may be downloaded from your download area. This ISO image may be written to a CD, a DVD, or a USB key. It is also possible to customize the image with some information related to your infrastructure. The image size varies with different releases of the product, so it may exceed the usable capacity of CDs.

The important fact is that, unlike other BMR approaches, no source system specific information is stored on the recovery media itself, which means that only one site-specific disaster recovery media is required.

4 Setting up LinuxBMR

To set up Linux BMR for **Bacula Enterprise**, several steps are required.

This chapter guides through all the steps necessary to set up and configure the needed LinuxBMR infrastructure.

4.1 Configuration Overview

These can be grouped into the following categories:

- Installing Bacula Enterprise Client
- Enabling BMR in your backup jobs
- Preparing BMR restore
- Creating BMR rescue media.

4.2 Installing Bacula Enterprise Client

In order to protect a server with the LinuxBMR toolkit, the Bacula Enterprise Client needs to be installed on this server. The client package is available in your download area. You will need at least the bacula-enterprise-client and the bacula-enterprise-libs (or bacula-enterprise-common) packages.

The `Bacula-rescue.sh` program is executed on the Client to collect critical system information. This program is included in the client package beginning with the 8.8.0 release of Bacula Enterprise. If an older version is in use, the bacula-enterprise-client-bmr package needs to be installed on the client(s). This package is available in your download area.

4.3 Enabling BMR in Jobs

For the jobs that should be BMR-enabled, two requirements have to be met. One is that critical system information has to be generated and backed up, and the other is that all file systems that are required must be included in the job.

The first requirement – ensuring that the system information required for BMR is available – is actually quite simple: The script that generates this information needs to be executed prior to the actual backup. This is done with a **Run Script** for the BMR-enabled jobs. To make things simpler, we recommend to use a **JobDefs** resource for those jobs. An example JobDefs resource and the configuration for an actual LinuxBMR Job are shown below:

```
JobDefs {
  Name = "LinuxBMR-JD"
  Client = lsb-245-fd

  # This FileSet contains the entire Linux System
  File Set = "AllUnix-U1004like"

  Type = Backup
  Level = Incremental
  Storage = File
  Messages = Standard
  Pool = Tier1
  Priority = 10
  Write Bootstrap = "/var/lib/bacula/%c-%n.bsr"

  # Enable LinuxBMR
  ClientRunBeforeJob = "/opt/bacula/bin/Bacula-rescue.sh"
}
```

Note: Note the **ClientRunBeforeJob** directive which ensures that the `Bacula-rescue.sh` script is executed to collect the critical system information. Alternatively, a **RunScript** block can be used.

```
Job {
  Name = lsb-246-bmr
  JobDefs = LinuxBMR-JD
  Client = lsb-246-fd
}
```

The second requirement is to ensure that all data is actually backed up. Typically, this means **everything** except your specific application data such as SQL database files. This requires a **File Set** which includes all local file systems.

In general, this is not difficult to achieve. However, depending on your site's requirements, it may be helpful to create the **File Set** includes automatically, on the client side, when the job is run. An example of how to automatically include all local file systems with selected filesystem types is shown below (if you have other Linux filesystems in use, you will have to extend the list below):

```
FileSet {
 Name = AllUnix-U1004like
 Include {
   Options {
      signature=MD5
      compression = LZO
      xattrsupport = yes
```

(continues on next page)

(continued from previous page)

```
      aclsupport = yes
      onefs = no
      fstype = ext2, ext3, ext4, xfs, msdos
   }
   File = /
  }
  Exclude {
    File = /tmp                   # exclude temp files
    File = /var/tmp
    File = /opt/bacula/working/*   # exclude Bacula working files

    File = /var/lib/postgres/data  # ensure that your databases
                                   # are saved by an other way.
  }
}
```

4.4 Preparing BMR Restore

To make the BMR configuration complete, a Rescue Client resource that will be used during restore needs to be defined:

```
Client {
  Name = rescue-fd                 # Use your rescue client name
  Address = 0.0.0.0                # Will be set automatically by LinuxBMR
  Password = bacularescue          # USE YOUR OWN PASSWORD
  Catalog = MyCatalog
}
```

During a LinuxBMR restore session, the LinuxBMR system needs to contact the **Director**. To do so a Console is required in the Director's configuration. For security reasons a named Console is used, which may be configured with restricted permissions. A Console resource for use with the LinuxBMR toolkit could be set up as follows:

```
Console {
  Name = rescue-fd                # MUST HAVE same name as Client resource
  Password = bacularescue         # USE YOUR OWN!
  CommandACL = *all*
  ClientACL = *all*
  CatalogACL = *all*
  JobACL = *all*
  StorageACL = *all*
  ScheduleACL = *all*
  PoolACL = *all*
  FileSetACL = *all*
  WhereACL = *all*
  # The next two ACLs are required when using
  # Bacula Enterprise 8.8.0 and above
  UserIdACL = *all*
  DirectoryACL = *all*
  # This last ACL is available when using
  # Bacula Enterprise 8.8.0 and above but is not required
  RestoreClientACL = *all*
}
```

Finally a restore job resource which does not have any run scripts is needed by the LinuxBMR restore process.

4.5 Creating LinuxBMR Rescue Media

The LinuxBMR recovery image may be downloaded from your download area. This ISO image may be written to a CD, a DVD, or a USB key. It is also possible to customize the image with some information related to your infrastructure.

- LinuxBMR-rescue-amd64-2.0.0.iso

The size of the image can vary from version to version. The latest version is currently too big to fit on a CD media, but that may change for future releases, as we do our best to keep the size as small as possible. The ISO image file may be used directly on VMware or VirtualBox to boot a Virtual Machine.

On , tools like `Win32DiskImager` are available to write the image to a USB stick. This tool may be downloaded from https://sourceforge.net/projects/win32diskimager/

Warning: Attention!

On Linux, the `cp` or `dd` commands can be used to do the job. This needs to be done carefully, as the procedures described in this section will destroy anything already on the target! Make very sure that you use the correct device name for your USB stick. If you use the wrong device the result could be that all information on your hard disk is lost.

With current systems, a USB stick should be automatically recognized when inserted. If it is not, you should check that the usb-storage kernel module is loaded. When the USB stick is inserted, it will be mapped to a block storage device named `/dev/sdX`, where the "X" usually is a letter in the range a-z. You should be able to see to which device the USB stick was mapped by running the command `dmesg` after inserting it.

To write to your stick, you may have to turn off its write protection switch.

```
root# cp LinuxBMR-rescue-amd64-2.0.0.iso /dev/sdX
root# sync
```

Make sure to use `/dev/sdX` and not `/dev/sdX1`. This is a very common error.

5 Doing BMR

Bare-Metal Recovery itself is done by booting the target system from the BMR image which should have already been created as described in section *Creating LinuxBMR Rescue Media*. Depending on the target computer's setup, booting from the BMR image may require some BIOS settings changes or interaction during the boot process – check with the system's manual if unsure!

5.1 Starting the Recovery System

Once the BMR system boots, you are first presented with the `Isolinux` language selection screen which will look similar to the one in the screenshot below. Use the up or down arrow keys on the keyboard to get to your language and then press enter.

Note: You are selecting the language of the Linux Desktop environment, but the BMR tool itself is available only in English.

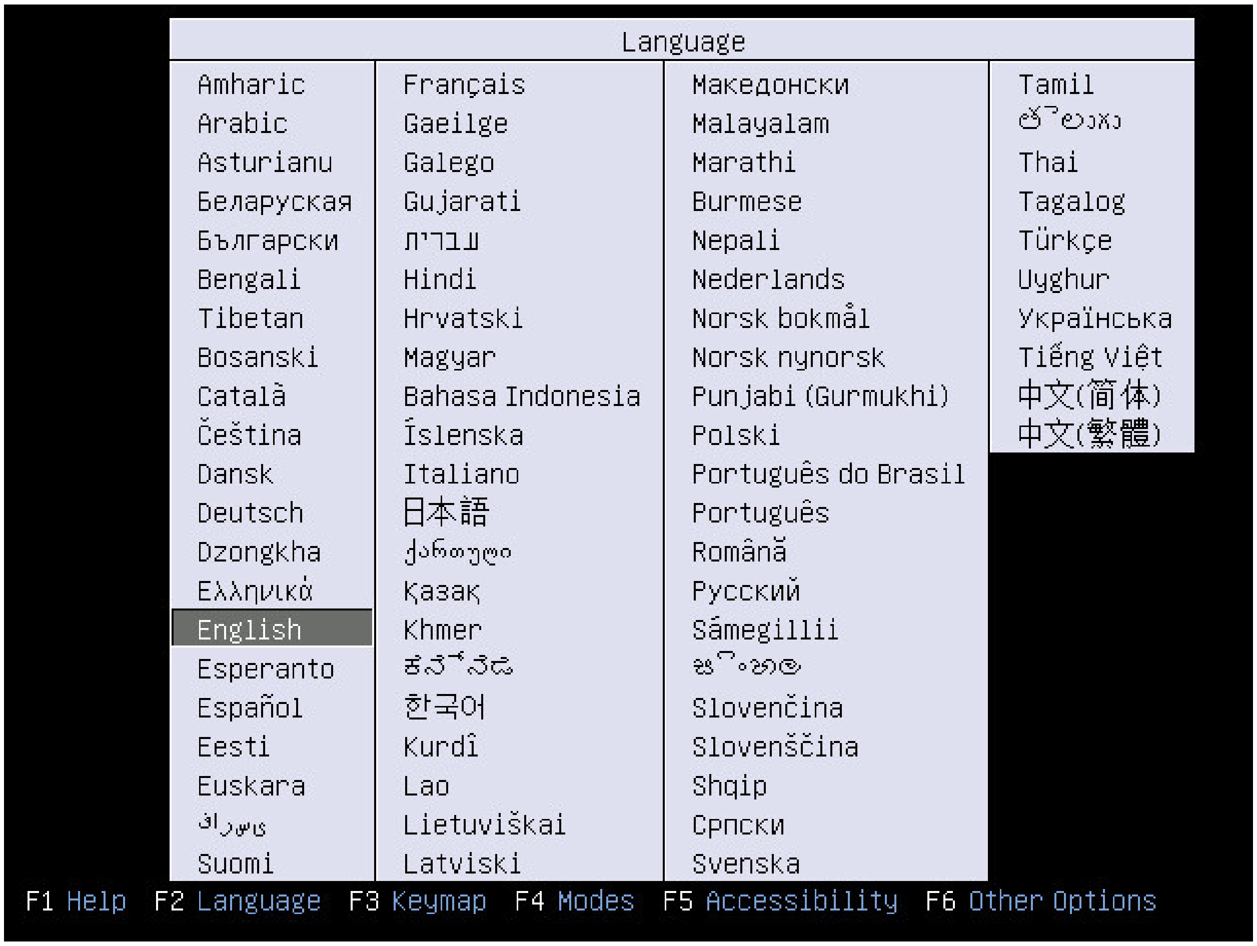


Fig. 1: The BMR Boot Screen – Language Selection

Then press `F3` to select your keyboard mapping:

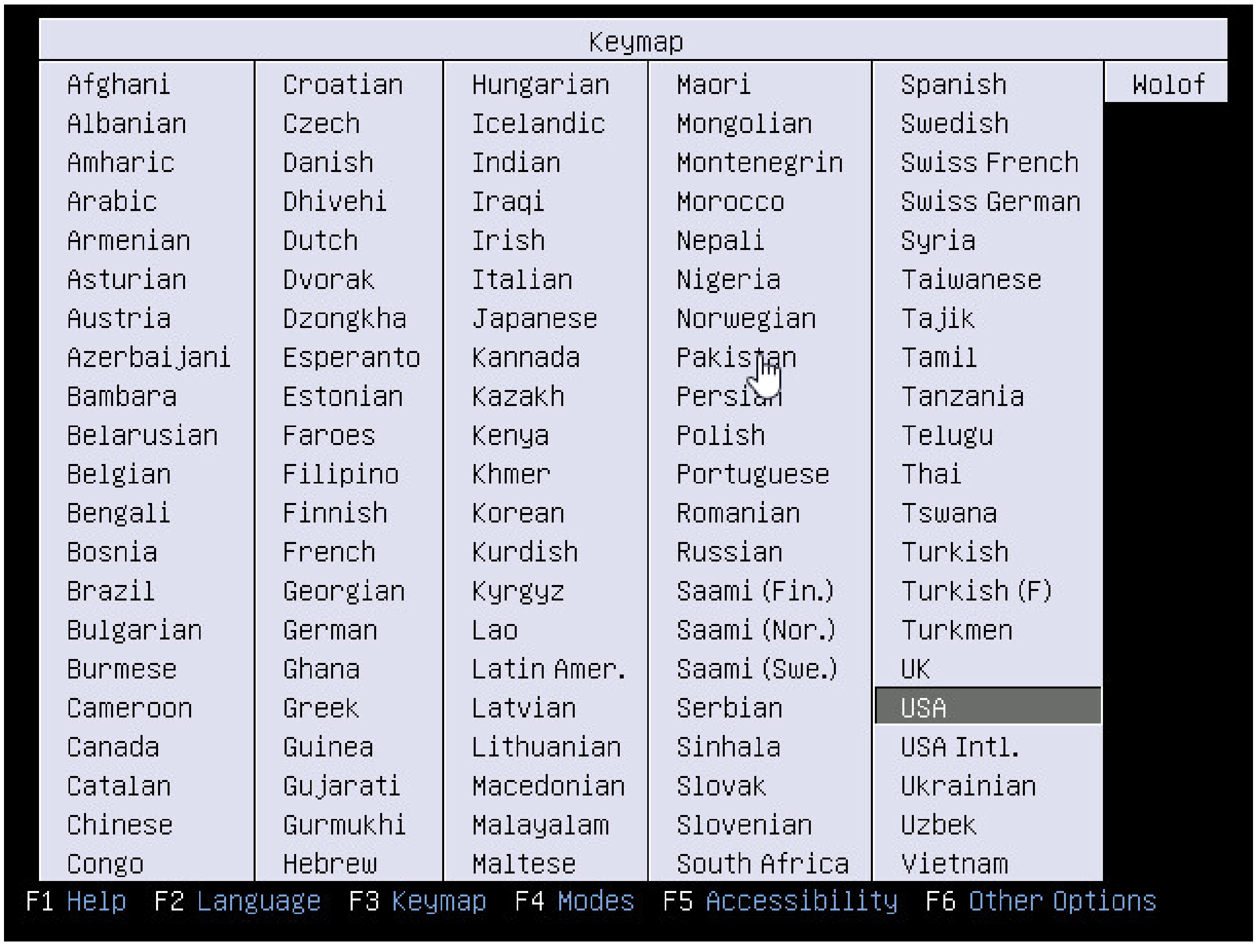


Fig. 2: The BMR Boot Screen – Keymap Selection

Finally select “Start Bacula LinuxBMR” and press `Enter`.

After a short time, the Recovery system will be booted into a graphical desktop as shown below:

Before starting the BMR process itself, it may be useful to verify that the automatically determined network settings fit your environment. This is especially important if there is more than one network card (i. e., a production network and a dedicated backup network), or VLANs are used and no automatic address assignment is available. In these cases, manual configuration of the network settings may be required.

To do that, click the `Network Manager` tray icon in the bottom right of the screen (left of the clock) as shown above.

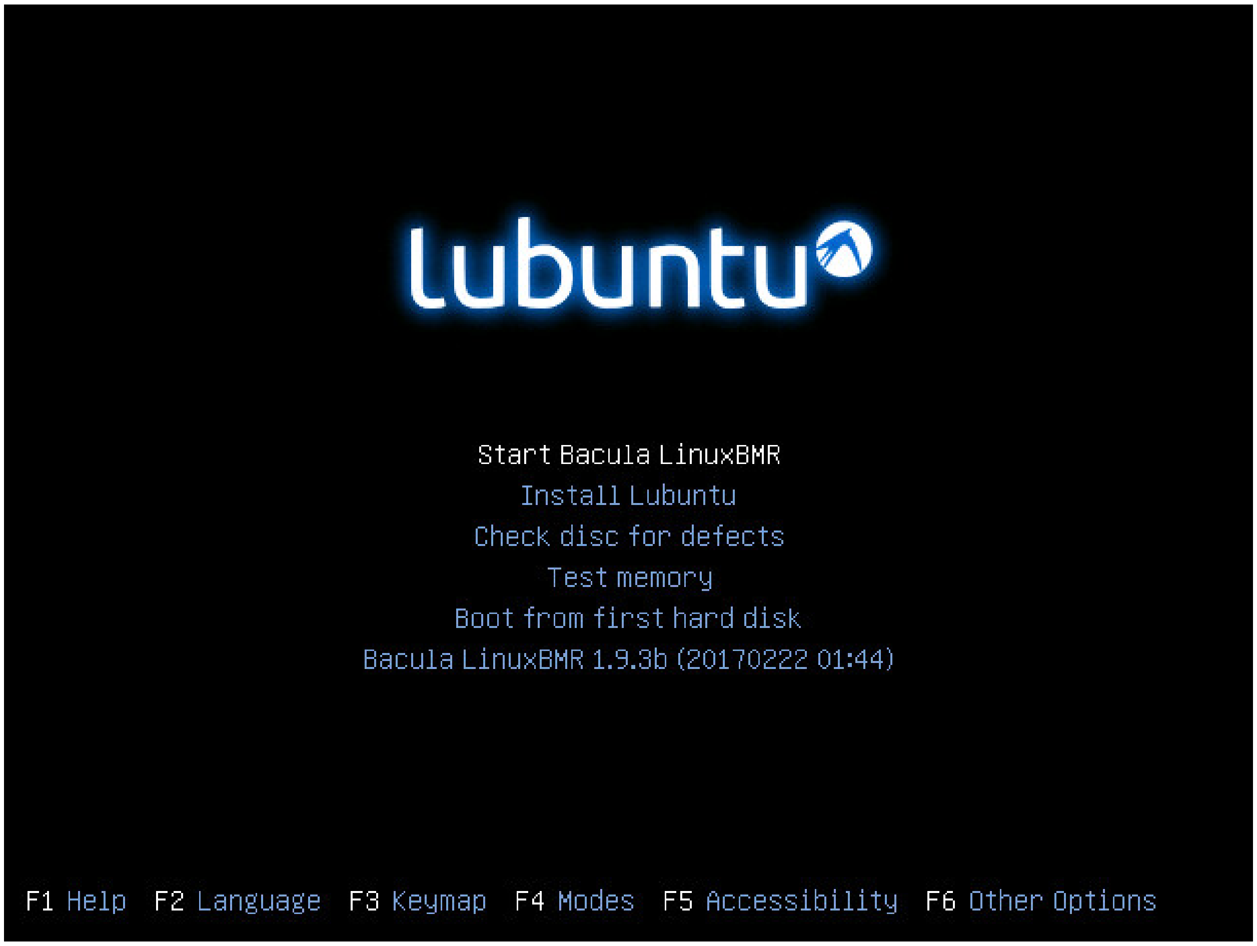

lubuntu
Start Bacula LinuxBMR
Install Lubuntu
Check disc for defects
Test memory
Boot from first hard disk
Bacula LinuxBMR 1.9.3b (20170222 01:44)
F1 Help F2 Language F3 Keymap F4 Modes F5 Accessibility F6 Other Options

Fig. 3: The BMR Boot Screen – Boot

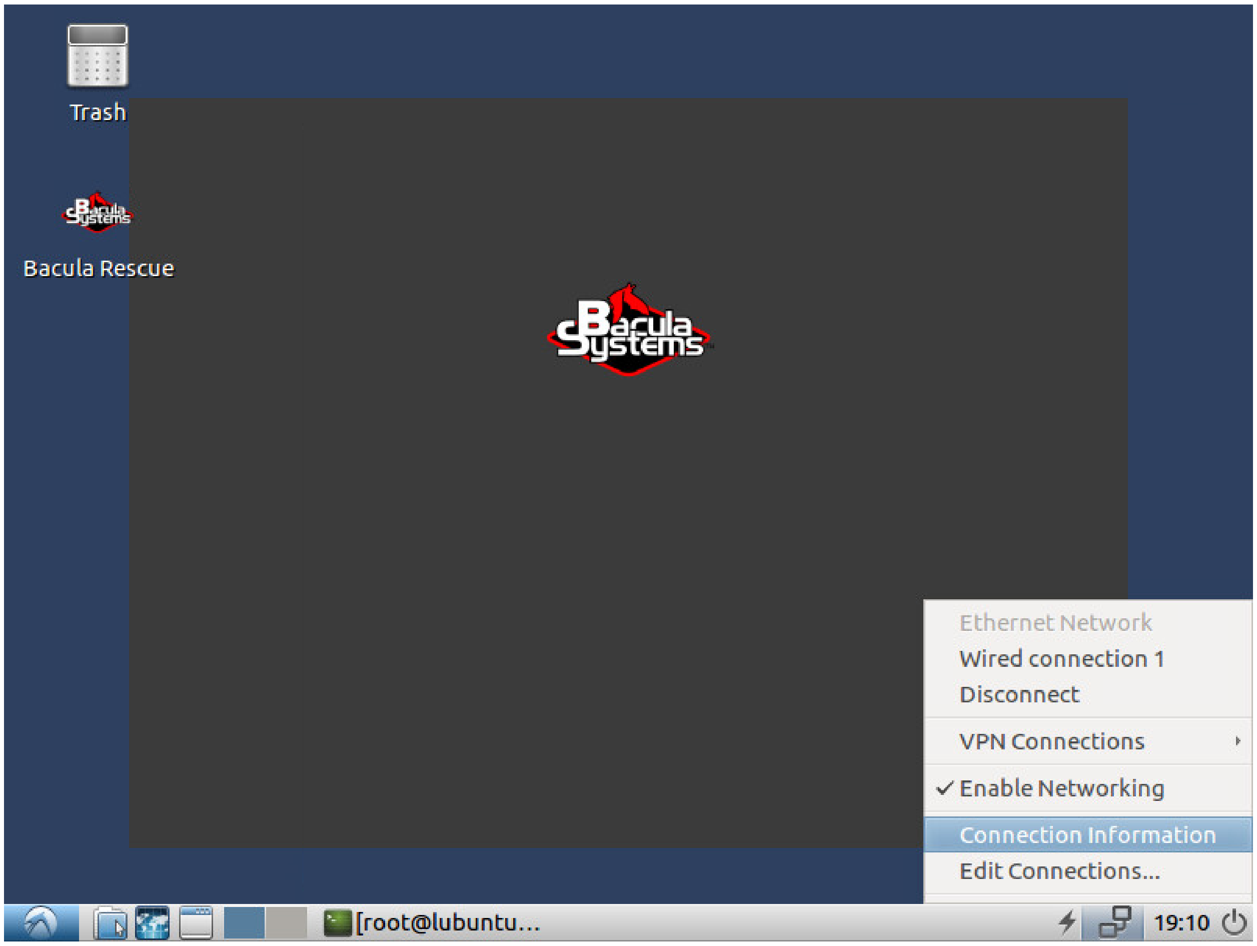
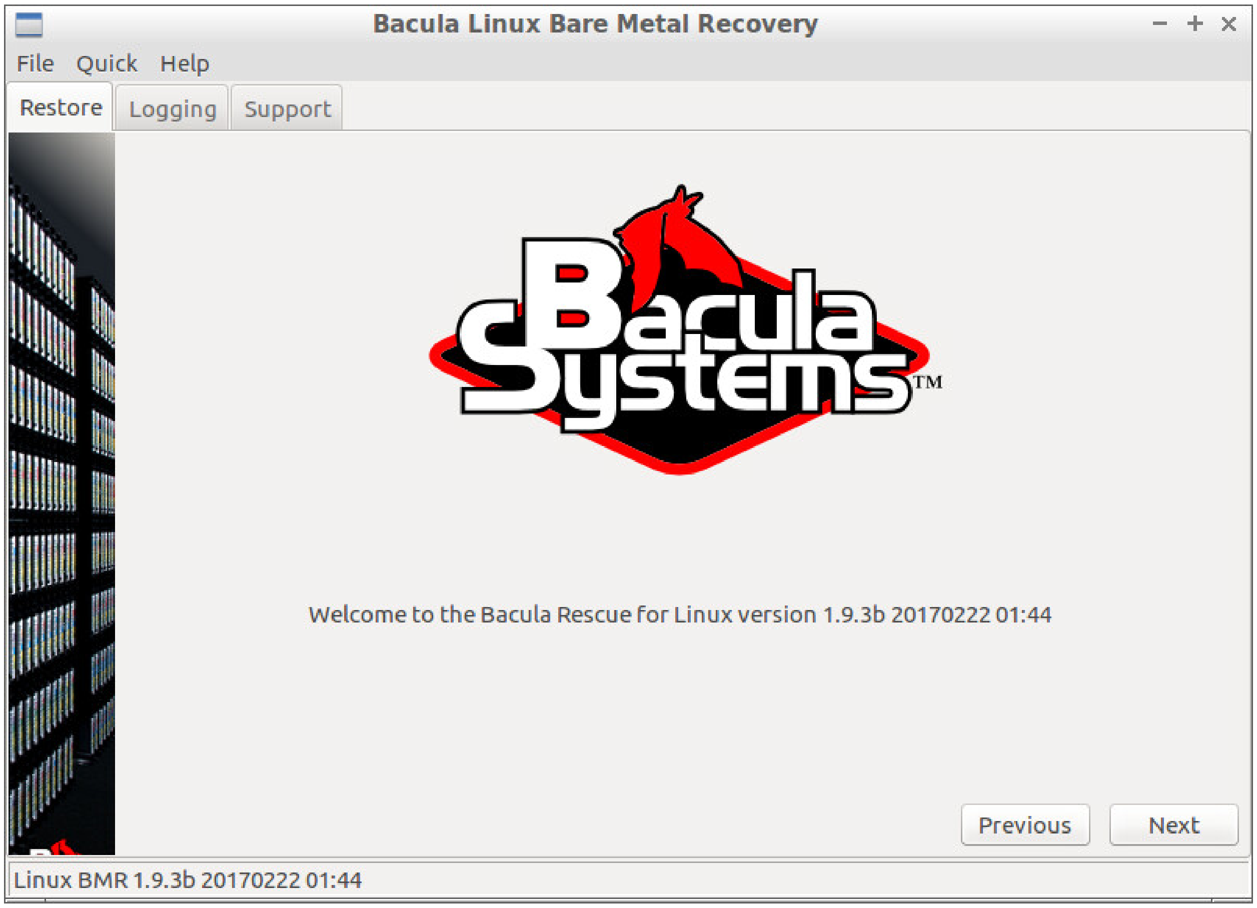


Fig. 4: The BMR System's Desktop

5.2 Starting the BMR Process

Once the network is configured correctly, you can start the Bacula Rescue session by double-clicking its icon on the desktop.

The first screen presented is a welcome message:



Welcome to the Bacula Rescue for Linux version 1.9.3b 20170222 01:44

Linux BMR 1.9.3b 20170222 01:44

Fig. 5: Welcome Page

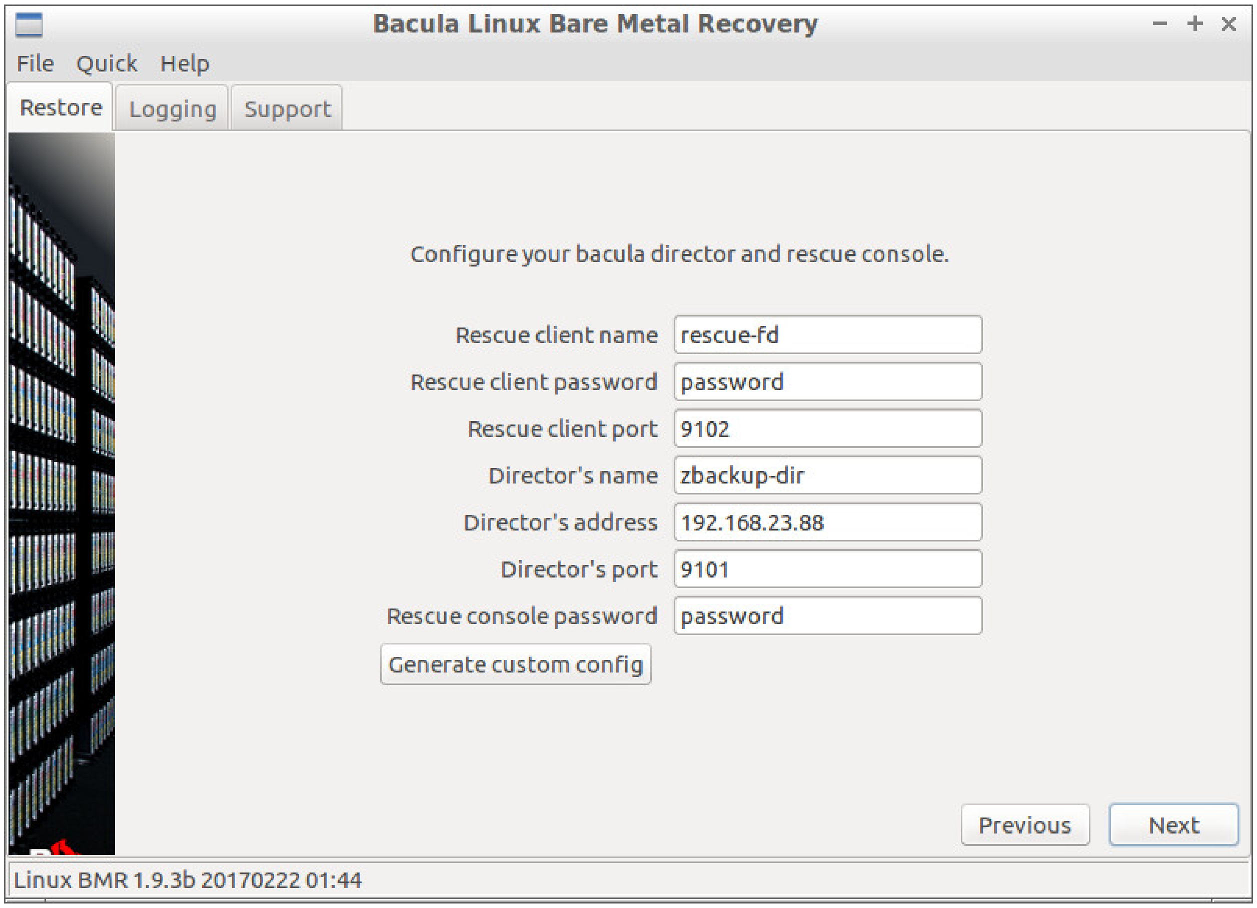
The menu and the 3 tabs in the windows top left corner are explained in section *More about the GUI*. Click `Next` to continue.

5.3 The Configuration Screen

The next screen is the Configuration screen:

The Rescue client name, password and port have to match the ones defined in the the **Client** resource in *Preparing BMR Restore*.

The Director name, address and port are required to connect to your **Director**. Make sure you can connect to your Director from your machine: that the network is well configured, that the Name resolution (DNS) will resolve the name of your **Director** if you are not using an IP address, and that on the other side, your firewall will not block the incoming packages. The Rescue console password is the password set in *Preparing BMR Restore*. These values are used to create the client and console configuration files `bacula-fd.conf` and `bconsole.conf` in the directory `/bs-rescue`.



Configure your bacula director and rescue console.

Rescue client name	rescue-fd
Rescue client password	password
Rescue client port	9102
Director's name	zbackup-dir
Director's address	192.168.23.88
Director's port	9101
Rescue console password	password

Linux BMR 1.9.3b 20170222 01:44

Fig. 6: Bacula Configuration

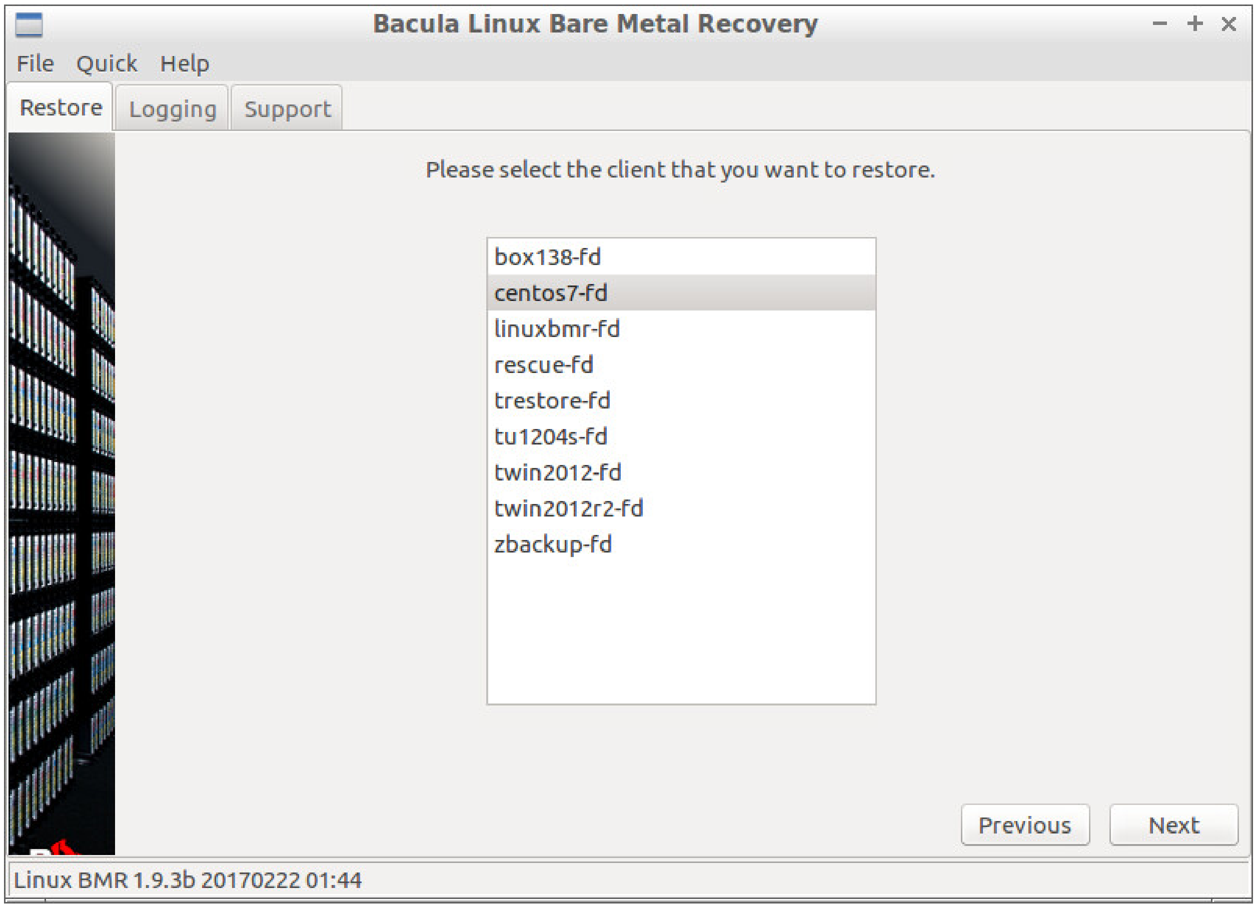
The “Generate custom config” button creates a copy of the client configuration file into `bacula-fd.custom` and opens it in a text editor. You may modify it and even add directives as needed. For example you can add TLS or data encryption to your configuration. Once saved, this file will be used by the BMR Client.

Click “Next” to check if the configuration is correct and then go to the next step (for connection troubleshooting see section *Connection Troubleshooting*).

Now the client daemon is running. It is accessible from the **Director**, and the client and server-side programs will communicate to restore data.

5.4 Selecting What to Restore

Now is the time to choose the system to be restored to the current host. For this purpose, the BMR tool presents you with a list of all clients known to the **Director** in alphabetical order, which looks like the one shown in the screenshot below. Select the correct client and continue by clicking the “Next” button. If you selected a client which does not have any backups available, a message stating that no backups are available is displayed and you can go back to the client selection step.



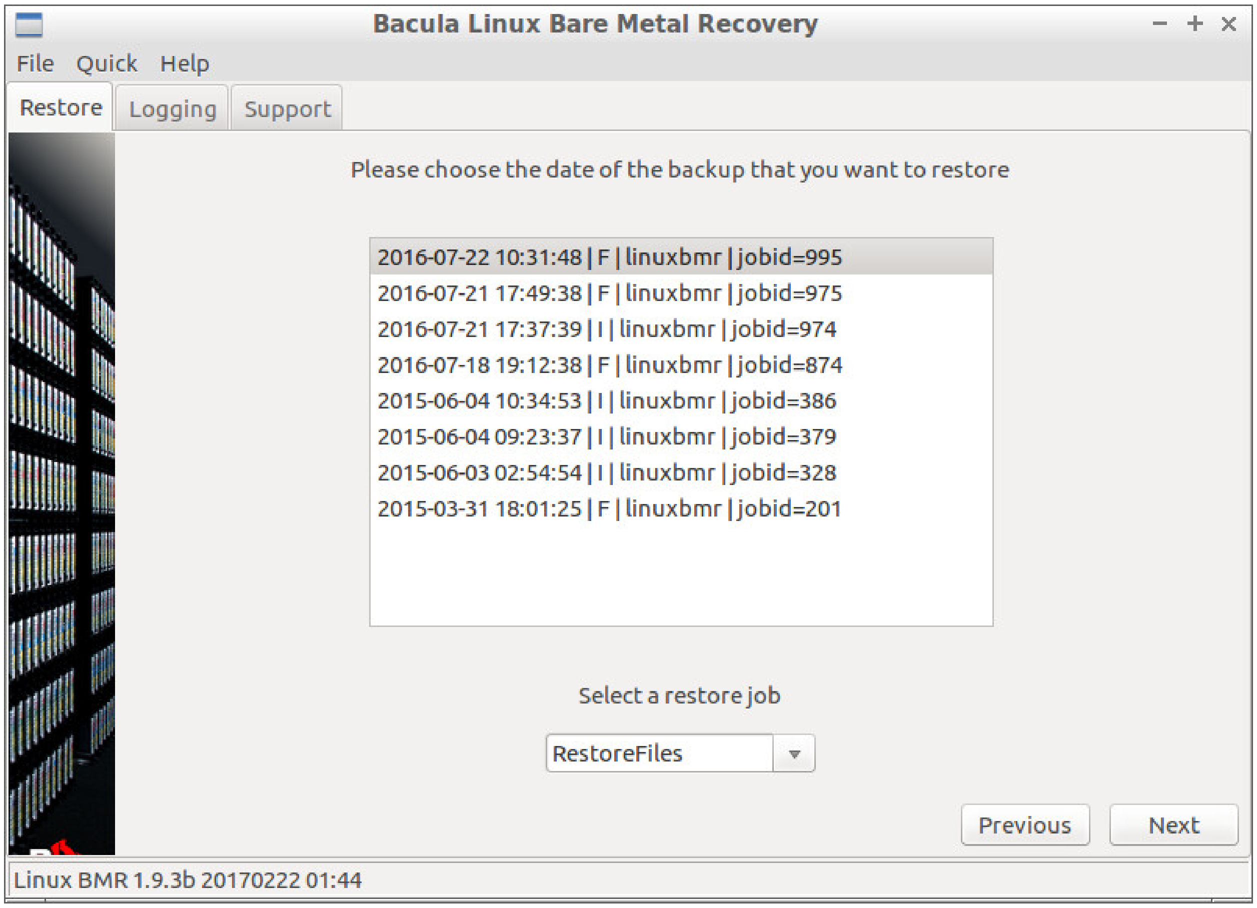
Please select the client that you want to restore.

box138-fd
centos7-fd
linuxbmr-fd
rescue-fd
trestore-fd
tu1204s-fd
twin2012-fd
twin2012r2-fd
zbackup-fd

Linux BMR 1.9.3b 20170222 01:44

Fig. 7: Source Client Selection

The next screen allows you to pick the date you want to restore from. All existing backups for the selected client are shown. You must select a BMR enabled backup. If the chosen job is not a good one, a dialog box will warn you. Previous versions of the LinuxBMR product were filtering the list of backups, to display only the BMR enabled backups, but this was sometimes too slow, so the filter has been removed (a simple and reliable solution is to clearly name all jobs that are BMR-enabled, for example with a job name prefix or suffix of “LinBMR”).



Please choose the date of the backup that you want to restore

```
2016-07-22 10:31:48 | F | linuxbmr | jobid=995
2016-07-21 17:49:38 | F | linuxbmr | jobid=975
2016-07-21 17:37:39 | I | linuxbmr | jobid=974
2016-07-18 19:12:38 | F | linuxbmr | jobid=874
2015-06-04 10:34:53 | I | linuxbmr | jobid=386
2015-06-04 09:23:37 | I | linuxbmr | jobid=379
2015-06-03 02:54:54 | I | linuxbmr | jobid=328
2015-03-31 18:01:25 | F | linuxbmr | jobid=201
```

Select a restore job

RestoreFiles

Linux BMR 1.9.3b 20170222 01:44

Fig. 8: Source Date Selection

A restore job that does not have any **RunScript** directives is automatically selected. You can choose a different one in the combobox at the bottom of the screen.

5.5 Selecting Where to Restore to

If a suitable backup was chosen, the BMR tool loads information about the disk layout of the source host. The same layout, or part of it will be reproduced on the target host, then the data will be restored to the freshly formated disk(s) and finally the tool will configure the boot loader.

The process is split into the following operations:

- Disk mapping and exclusion
- Partition and volume resizing
- Partitioning
- Volume mapping and exclusion
- Restoring the data
- Configuring the boot loader

When the source and your target systems are identical, clicking “Next” will do the job most of the time. If the systems are different, the proposed setup will often give good results, but you should double check and then validate with “Next”, or adjust the inputs to meet your needs.

A small glossary for the following sections:

- A **disk** is a physical or virtual hard-disk.
- A **partition** is a part of a disk.
- A **volume** is a partition, a logical volume built on top of LVM, or a device created by the Linux RAID Software, that holds a filesystem and can be mounted to a mount point.
- A **mount point** is a path inside the filesystem where a volume is mounted. We will often use the path as an alias to refer to the underlying volume.

Disk Mapping

The next screenshot shows what the Disk mapping screen looks like. The two lists on the left show the disks of the source host (from backup) and the ones available on the target host. The right panel shows how volumes and their related mount points depend on the underlying disks of the source host.

The task here is to map and align every disk of the source host to one disk of the target. This is useful if you have disks that are different in size or speed.

Use the up and down arrows on the left to move a selected disk up and down in the left list. Use the right arrows to move the selected disk on the right list.

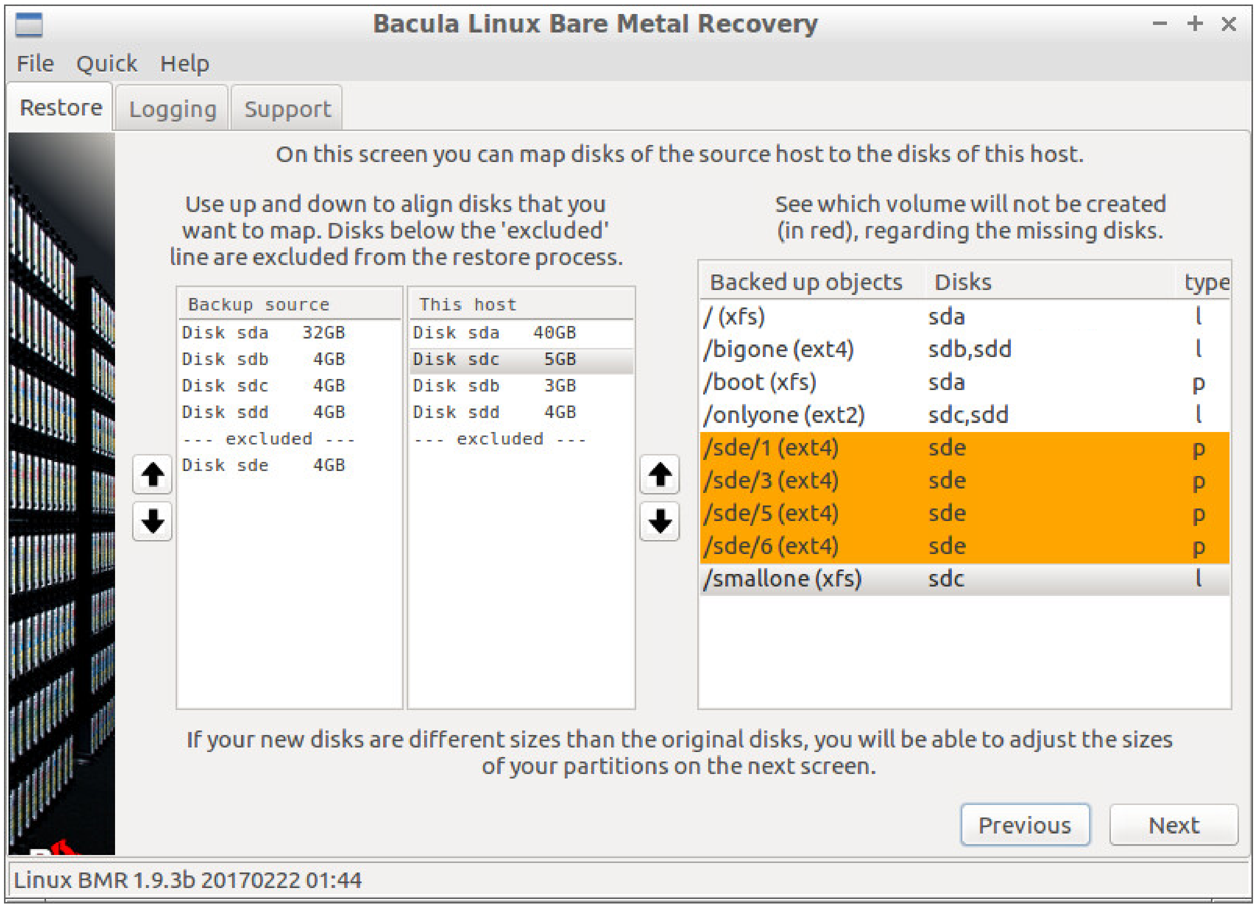
If the number of disks on both systems does not match, then some disks get excluded and are moved below the `- excluded -` separation line in order to have the same number of disks above the line for a one to one mapping.

You can move some disks below this separation line yourself to exclude them during the restore process.

Note: Existing `LVM` and `RAID` software information on these excluded disks **will be deleted** to avoid name collisions, but the partitions will remain untouched.

All volumes and filesystems that are related to these excluded disks will not be created and are shown in orange on the right panel.

The first disk of the target host will be the one the boot loader is installed to. It must also be chosen as the bootable disk in your BIOS. This is often the first disk in the list.



On this screen you can map disks of the source host to the disks of this host.

Use up and down to align disks that you want to map. Disks below the 'excluded' line are excluded from the restore process.

See which volume will not be created (in red), regarding the missing disks.

```
Backup source          This host
Disk sda    32GB       Disk sda    40GB
Disk sdb     4GB       Disk sdc     5GB
Disk sdc     4GB       Disk sdb     3GB
Disk sdd     4GB       Disk sdd     4GB
--- excluded ---       --- excluded ---
Disk sde     4GB
```

Backed up objects	Disks	type
/ (xfs)	sda	l
/bigone (ext4)	sdb,sdd	l
/boot (xfs)	sda	p
/onlyone (ext2)	sdc,sdd	l
/sde/1 (ext4)	sde	p
/sde/3 (ext4)	sde	p
/sde/5 (ext4)	sde	p
/sde/6 (ext4)	sde	p
/smallone (xfs)	sdc	l

If your new disks are different sizes than the original disks, you will be able to adjust the sizes of your partitions on the next screen.

Linux BMR 1.9.3b 20170222 01:44

Fig. 9: Disk Mapping between Source and Target hosts

In the screenshot above you can see that the target host does not have a 5th disk. Thus, the 4 volumes `/sde/[1356]` in orange on the the right panel will not be created. We could choose to exclude another disk but assume that `sde` is fine. You can also see that we swapped `sdb` and `sdc`, because the new `sdc` is bigger and we want to give this extra space to the `/bigone` volume.

Adjusting Partitions and Volumes Sizes

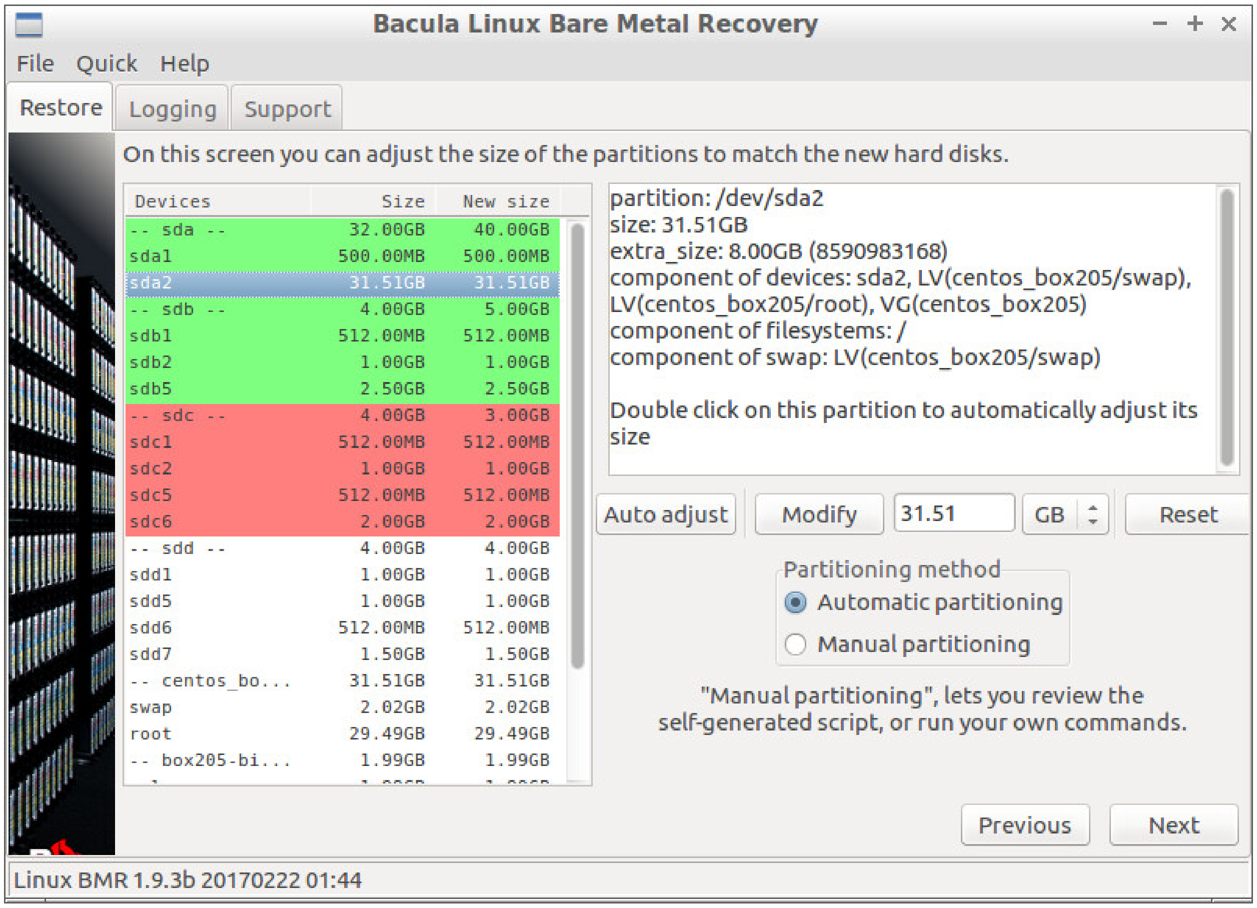
The same partitions and volumes as backed up will be created on the target host. You cannot add or remove any of them. If your disks are bigger or smaller than the source disks then you will have to choose which partitions or volumes will grow or shrink.

In the next screenshot you can see, on the left side, all partitions and volumes, and on the the right side the information related to the selected object on the left. For now, ignore the *Partitioning method* part on the bottom right that is described at the end of this section.

The areas in green have some extra space that can be assigned to the partitions and *Logical Volumes*. The areas in red will require some of their components to be shrunk.

`sda` is shown in green, used to have a size of 32GB on the source system and can now take advantage of 8 more GB. We can distribute this extra space between its partitions `sda1` and `sda2`.

To do that you must select the first partition `sda1`, then type the new size in the field at the right of the "Modify" button and click "Modify". You will see the new size allocated to the partition. Then select the other partition and do the



On this screen you can adjust the size of the partitions to match the new hard disks.

```
Devices            Size     New size
-- sda --         32.00GB   40.00GB
sda1             500.00MB  500.00MB
sda2              31.51GB   31.51GB
-- sdb --          4.00GB    5.00GB
sdb1             512.00MB  512.00MB
sdb2               1.00GB    1.00GB
sdb5               2.50GB    2.50GB
-- sdc --          4.00GB    3.00GB
sdc1             512.00MB  512.00MB
sdc2               1.00GB    1.00GB
sdc5             512.00MB  512.00MB
sdc6               2.00GB    2.00GB
-- sdd --          4.00GB    4.00GB
sdd1               1.00GB    1.00GB
sdd5               1.00GB    1.00GB
sdd6             512.00MB  512.00MB
sdd7               1.50GB    1.50GB
-- centos_bo...   31.51GB   31.51GB
swap               2.02GB    2.02GB
root              29.49GB   29.49GB
-- box205-bi...    1.99GB    1.99GB
```

partition: /dev/sda2
size: 31.51GB
extra_size: 8.00GB (8590983168)
component of devices: sda2, LV(centos_box205/swap), LV(centos_box205/root), VG(centos_box205)
component of filesystems: /
component of swap: LV(centos_box205/swap)

Double click on this partition to automatically adjust its size

Partitioning method
[● Automatic partitioning]
○ Manual partitioning

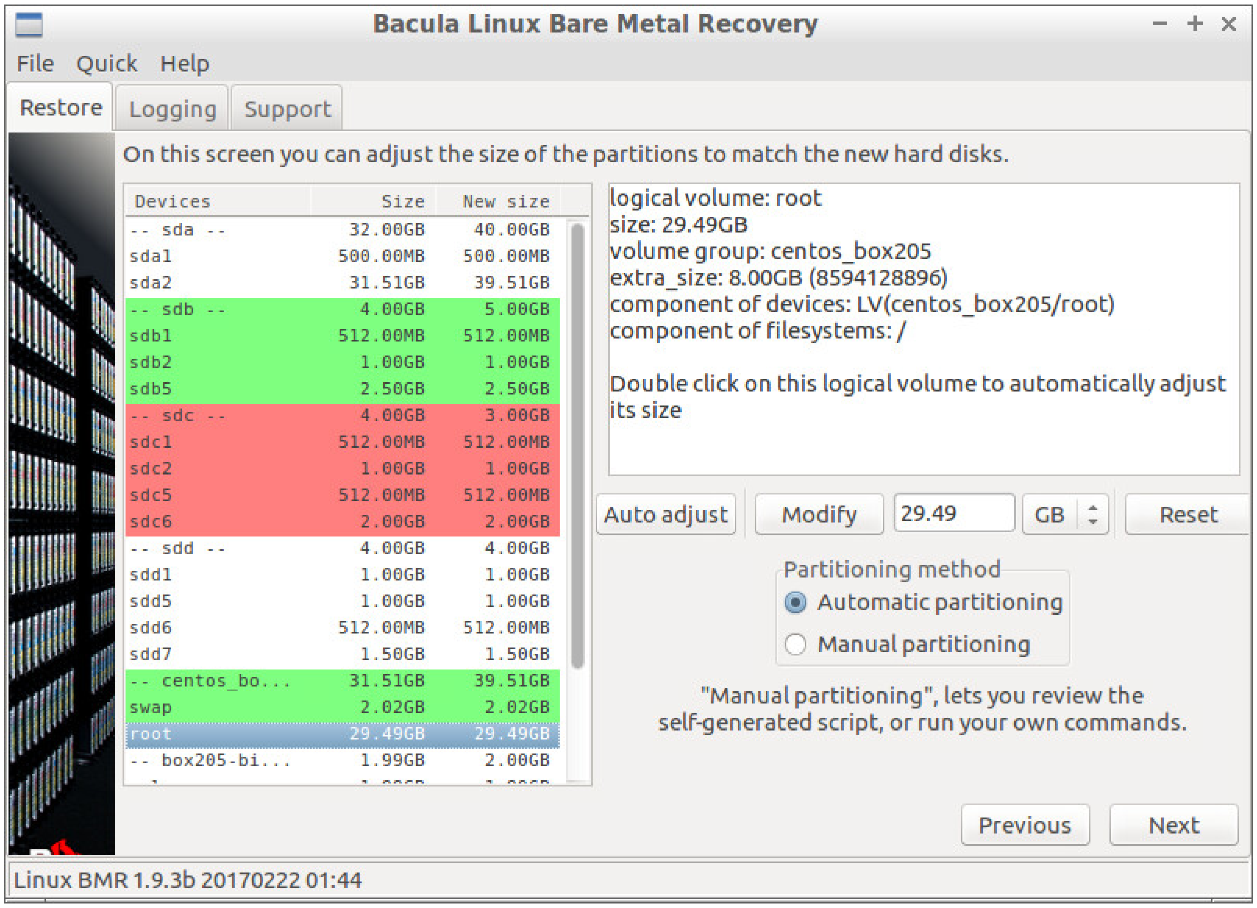
"Manual partitioning", lets you review the self-generated script, or run your own commands.

Linux BMR 1.9.3b 20170222 01:44

Fig. 10: Disk Resizing sda

same. To go faster you can give it all the remaining space by clicking the “Auto adjust” button. When all the extra space is allocated the disk and its partitions are not highlighted anymore.

Often you don’t want to divide the extra space and just give it all to one partition or volume. To go faster, double click on the partition that must grow (or shrink) and all the extra space is added (or lowered) to (or from) this partition or volume.



On this screen you can adjust the size of the partitions to match the new hard disks.

```
Devices              Size   New size
-- sda --         32.00GB    40.00GB
sda1             500.00MB   500.00MB
sda2              31.51GB    39.51GB
-- sdb --          4.00GB     5.00GB
sdb1             512.00MB   512.00MB
sdb2               1.00GB     1.00GB
sdb5               2.50GB     2.50GB
-- sdc --          4.00GB     3.00GB
sdc1             512.00MB   512.00MB
sdc2               1.00GB     1.00GB
sdc5             512.00MB   512.00MB
sdc6               2.00GB     2.00GB
-- sdd --          4.00GB     4.00GB
sdd1               1.00GB     1.00GB
sdd5               1.00GB     1.00GB
sdd6             512.00MB   512.00MB
sdd7               1.50GB     1.50GB
-- centos_bo...   31.51GB    39.51GB
swap               2.02GB     2.02GB
root              29.49GB    29.49GB
-- box205-bi...    1.99GB     2.00GB
```

logical volume: root
size: 29.49GB
volume group: centos_box205
extra_size: 8.00GB (8594128896)
component of devices: LV(centos_box205/root)
component of filesystems: /

Double click on this logical volume to automatically adjust its size

"Manual partitioning", lets you review the self-generated script, or run your own commands.

Linux BMR 1.9.3b 20170222 01:44

Fig. 11: Disk Resizing Root

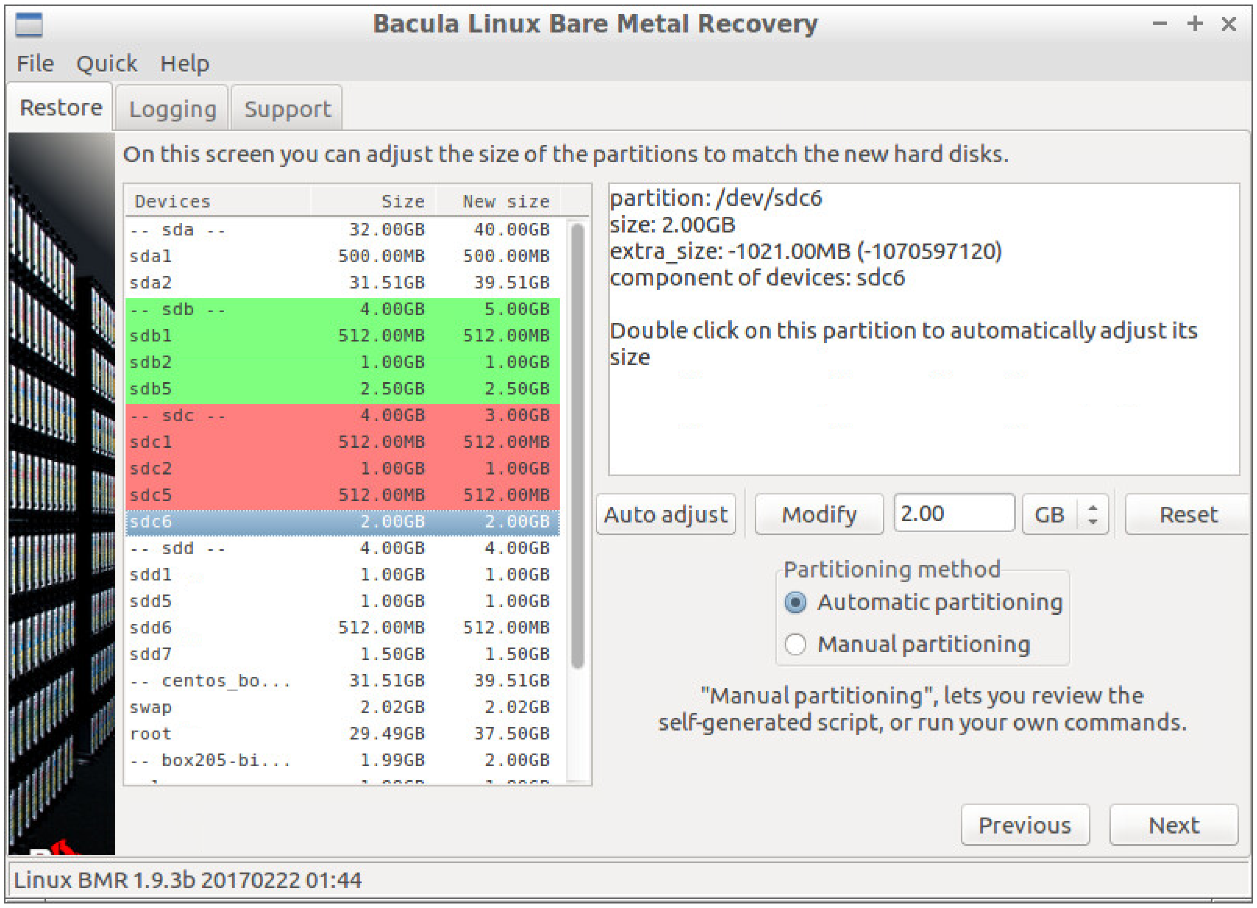
This is what we did with partition `sda2`. The `sda` disk is not green anymore, but the “Volume Group centos_box205” has become green. This *Volume Group* that got some extra space from the enlargement of `sda2` has 2 *Logical Volumes* that hold a `swap` area and the `root` filesystem. As we don’t want to increase the swap space, we double-click the `root` volume to give it all the space. See the result in the next screenshot below.

Now we handle the `sdc` disk that is smaller on the new system. We must shrink one ore more partitions to save 1GB. Double clicking on `sdc1` would return an error because the partition is too small to take all space difference on its own. We choose to double click and shrink `sdc6` instead.

Finally we see that `sdc6` has lost 1GB.

And finally we increase the size of `sdb2` and give more space to the `/bigone` filesystem (not visible in screenshot because that filesystem is at the bottom of the list).

Note: Most of the sizes here are approximate because tools used to create the partitions and `LVM` objects depend on their own alignment rules and granularity constraints. To make it work, we let the tool make the adjustments on the last partition of the disk or the last Logical Volume of the Volume Group. This means that the last object will be a few KB or MB smaller than expected.



On this screen you can adjust the size of the partitions to match the new hard disks.

```
Devices                 Size   New size
-- sda --            32.00GB    40.00GB
sda1                500.00MB   500.00MB
sda2                 31.51GB    39.51GB
-- sdb --             4.00GB     5.00GB
sdb1                512.00MB   512.00MB
sdb2                  1.00GB     1.00GB
sdb5                  2.50GB     2.50GB
-- sdc --             4.00GB     3.00GB
sdc1                512.00MB   512.00MB
sdc2                  1.00GB     1.00GB
sdc5                512.00MB   512.00MB
sdc6                  2.00GB     2.00GB
-- sdd --             4.00GB     4.00GB
sdd1                  1.00GB     1.00GB
sdd5                  1.00GB     1.00GB
sdd6                512.00MB   512.00MB
sdd7                  1.50GB     1.50GB
-- centos_bo...      31.51GB    39.51GB
swap                  2.02GB     2.02GB
root                 29.49GB    37.50GB
-- box205-bi...       1.99GB     2.00GB
```

partition: /dev/sdc6
size: 2.00GB
extra_size: -1021.00MB (-1070597120)
component of devices: sdc6

Double click on this partition to automatically adjust its size

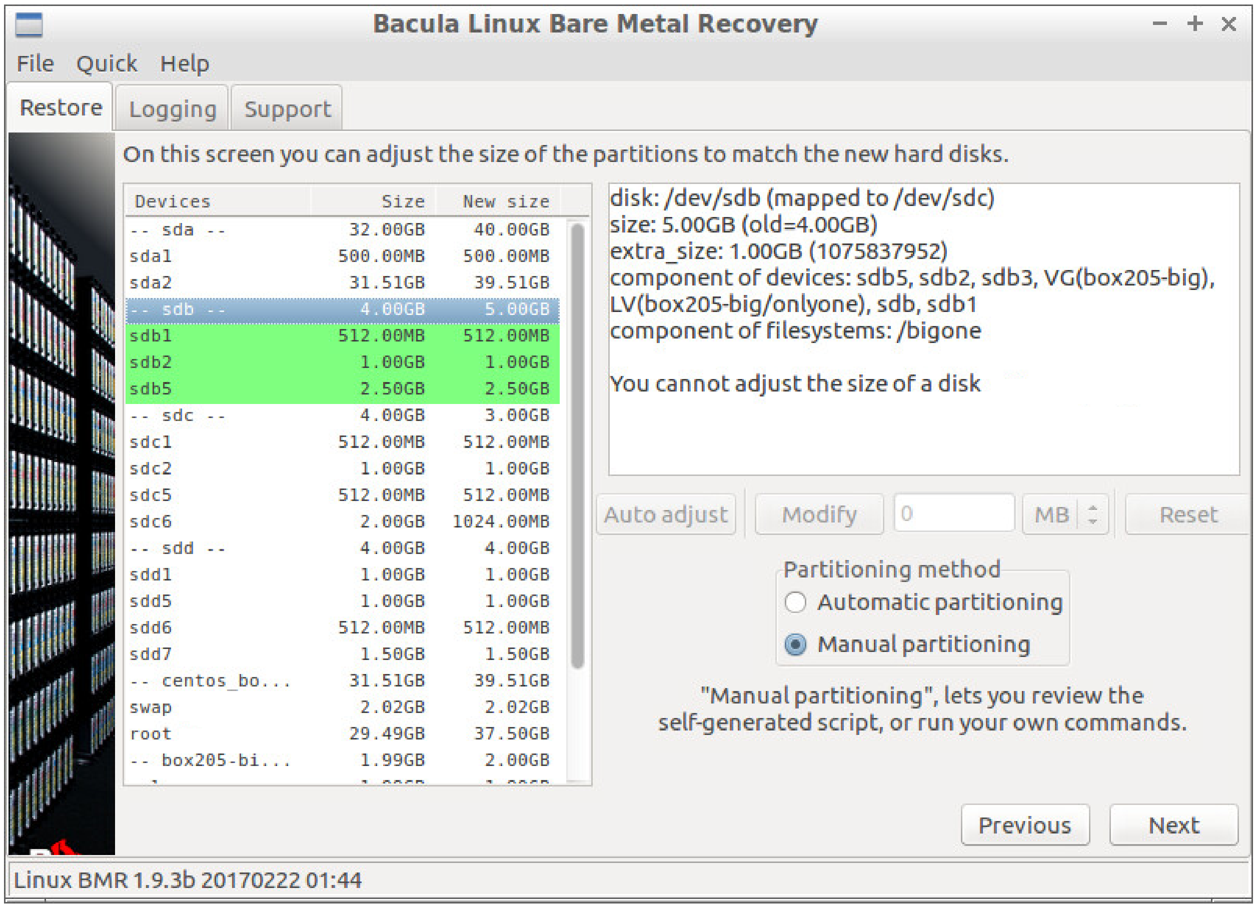
2.00 GB

Partitioning method
[✓ Automatic partitioning]
Manual partitioning

"Manual partitioning", lets you review the self-generated script, or run your own commands.

Linux BMR 1.9.3b 20170222 01:44

Fig. 12: Disk Resizing sdc6



On this screen you can adjust the size of the partitions to match the new hard disks.

```
Devices               Size    New size
-- sda --           32.00GB    40.00GB
sda1               500.00MB   500.00MB
sda2                31.51GB    39.51GB
-- sdb --            4.00GB     5.00GB
sdb1               512.00MB   512.00MB
sdb2                 1.00GB     1.00GB
sdb5                 2.50GB     2.50GB
-- sdc --            4.00GB     3.00GB
sdc1               512.00MB   512.00MB
sdc2                 1.00GB     1.00GB
sdc5               512.00MB   512.00MB
sdc6                 2.00GB  1024.00MB
-- sdd --            4.00GB     4.00GB
sdd1                 1.00GB     1.00GB
sdd5                 1.00GB     1.00GB
sdd6               512.00MB   512.00MB
sdd7                 1.50GB     1.50GB
-- centos_bo...     31.51GB    39.51GB
swap                 2.02GB     2.02GB
root                29.49GB    37.50GB
-- box205-bi...      1.99GB     2.00GB
```

```
disk: /dev/sdb (mapped to /dev/sdc)
size: 5.00GB (old=4.00GB)
extra_size: 1.00GB (1075837952)
component of devices: sdb5, sdb2, sdb3, VG(box205-big),
LV(box205-big/onlyone), sdb, sdb1
component of filesystems: /bigone

You cannot adjust the size of a disk
```

Partitioning method
( ) Automatic partitioning
[✓ Manual partitioning]

"Manual partitioning", lets you review the self-generated script, or run your own commands.

Linux BMR 1.9.3b 20170222 01:44

Fig. 13: Disk Resizing Final

For your information, the program displays an entire disk or Volume Group in green or red if the difference between the expected size and the available size is more than 4MB.

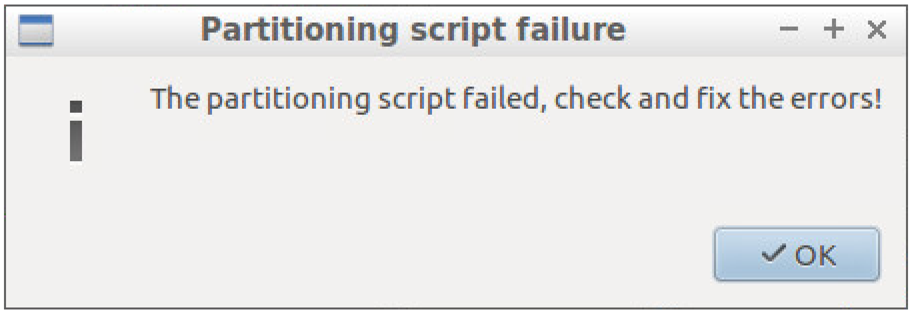
Even though there should never be any green or red areas left, the program will not complain about it and will try to have the partitioning and `LVM` tools do the adjustment on the last partition or Logical Volume. If the adjustment cannot be done (for example because there is no more space to create any partition or *Logical Volume* the partitioning will fail.

If you need precise sizing you can tweak the partitioning script yourself, which is described in the next section.

Partitioning

Finally we are done with the resizing and we can create the partitions and volumes. The BMR tool generates a shell script using the information coming from the source system and the choices made in the last two screens.

If you have selected "Automatic partitioning" and the script causes an error (see example below) you can read this section and fix the problem yourself, or contact the **Bacula Systems** Support Team.



The partitioning script failed, check and fix the errors!

Fig. 14: Partitioning Failure

If the generated script ran without error then the BMR tool skips the manual partitioning screen. Continue to read *Volume Mapping*.

If you are not interested in creating your own scripts or how to tweak the auto generated script, you can skip this section.

The script `mkpart-auto-generated.sh` in the list is the one that the BMR has generated. The two other scripts are here as a reminder that it is possible to write your own scripts in advance and store them in the `/opt/bacula/rescue/scripts` directory of your source host. Such custom scripts will be shown in the list if their names start with `mkpart-`.

You can edit and run the scripts using the "Edit script" button on the right. Their output is displayed in the small window at the bottom. The "Help" button provides information about what you have to do, how you can do it, and all the details about the source system.

Before leaving this screen, the BMR program reminds you that it is going to restore the data, and that the directory `/target` must be mounted and ready to receive the data.

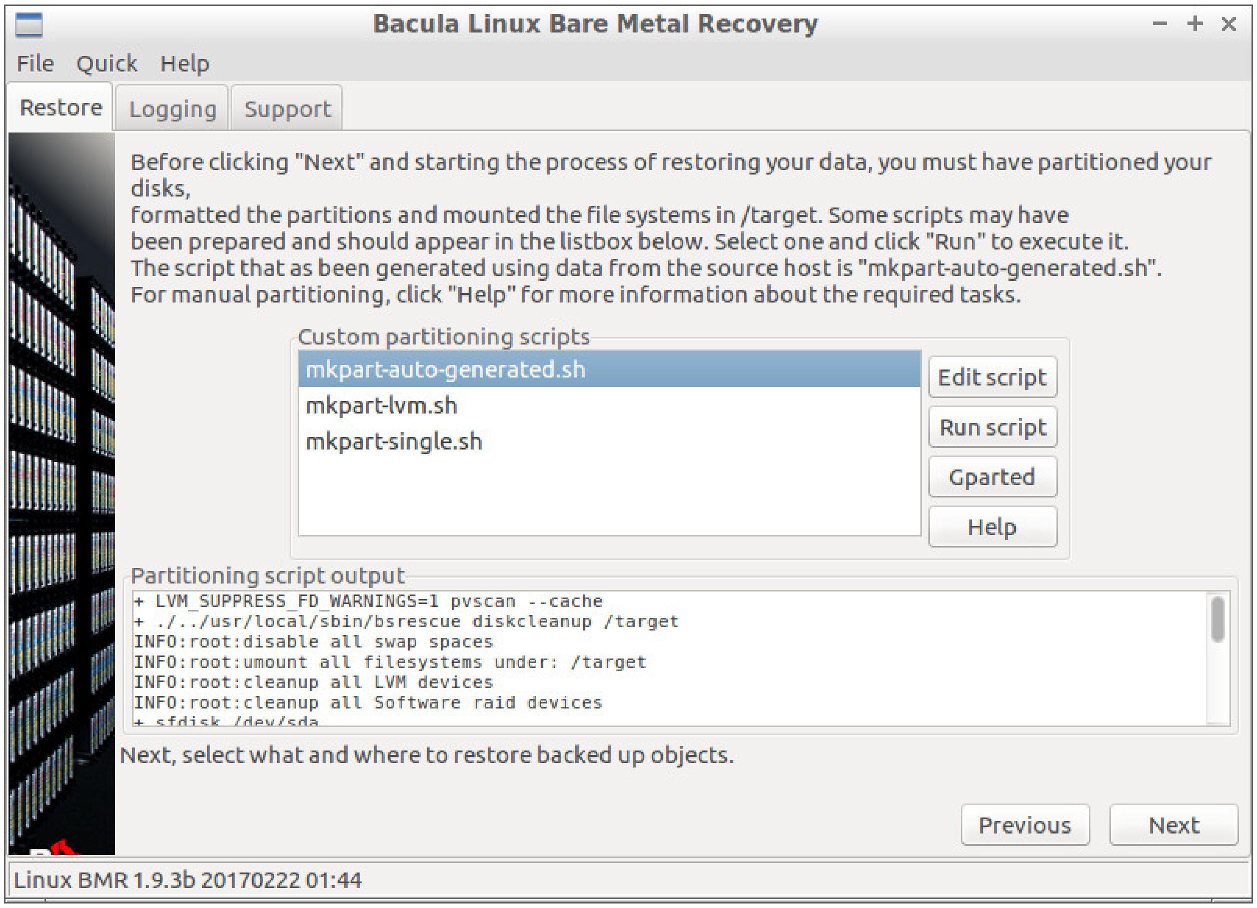
Volume Mapping

We are one step away from restoring the data and the purpose of the next screen is to match the volumes that have been backed up from the source host with the volumes that have been created by the Partitioning process on the target host.

In our example, we had no 5th disk to map the disk `sde`. As seen in the figure above, the program will not restore data in the directories that were stored on the missing disk – this is why it says *None* in the "Restore to" column. But we are in the process of redirecting the data in directory `/sde/1` to directory `/target/bigone` which we know to be big enough to fit the extra data.

You can double click on any volume of the source system and change the directory on the target system.

When done, click "Next" to start restoring your data. A progress bar will keep you informed about the status of the restore.



Before clicking "Next" and starting the process of restoring your data, you must have partitioned your disks,
formatted the partitions and mounted the file systems in /target. Some scripts may have
been prepared and should appear in the listbox below. Select one and click "Run" to execute it.
The script that as been generated using data from the source host is "mkpart-auto-generated.sh".
For manual partitioning, click "Help" for more information about the required tasks.

Custom partitioning scripts

- mkpart-auto-generated.sh
- mkpart-lvm.sh
- mkpart-single.sh

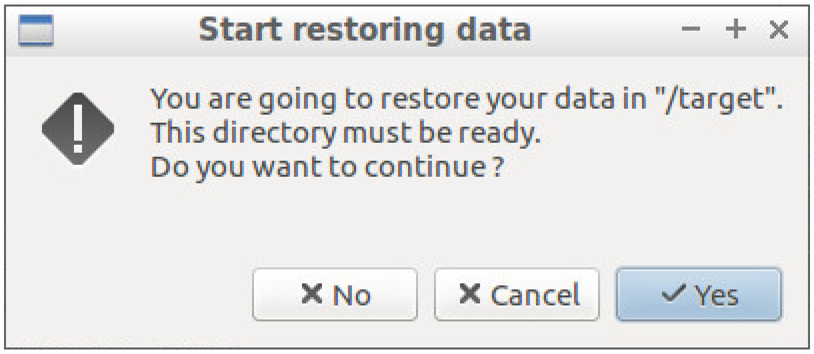
Partitioning script output

```
+ LVM_SUPPRESS_FD_WARNINGS=1 pvscan --cache
+ ./../usr/local/sbin/bsrescue diskcleanup /target
INFO:root:disable all swap spaces
INFO:root:umount all filesystems under: /target
INFO:root:cleanup all LVM devices
INFO:root:cleanup all Software raid devices
+ sfdisk /dev/sda
```

Next, select what and where to restore backed up objects.

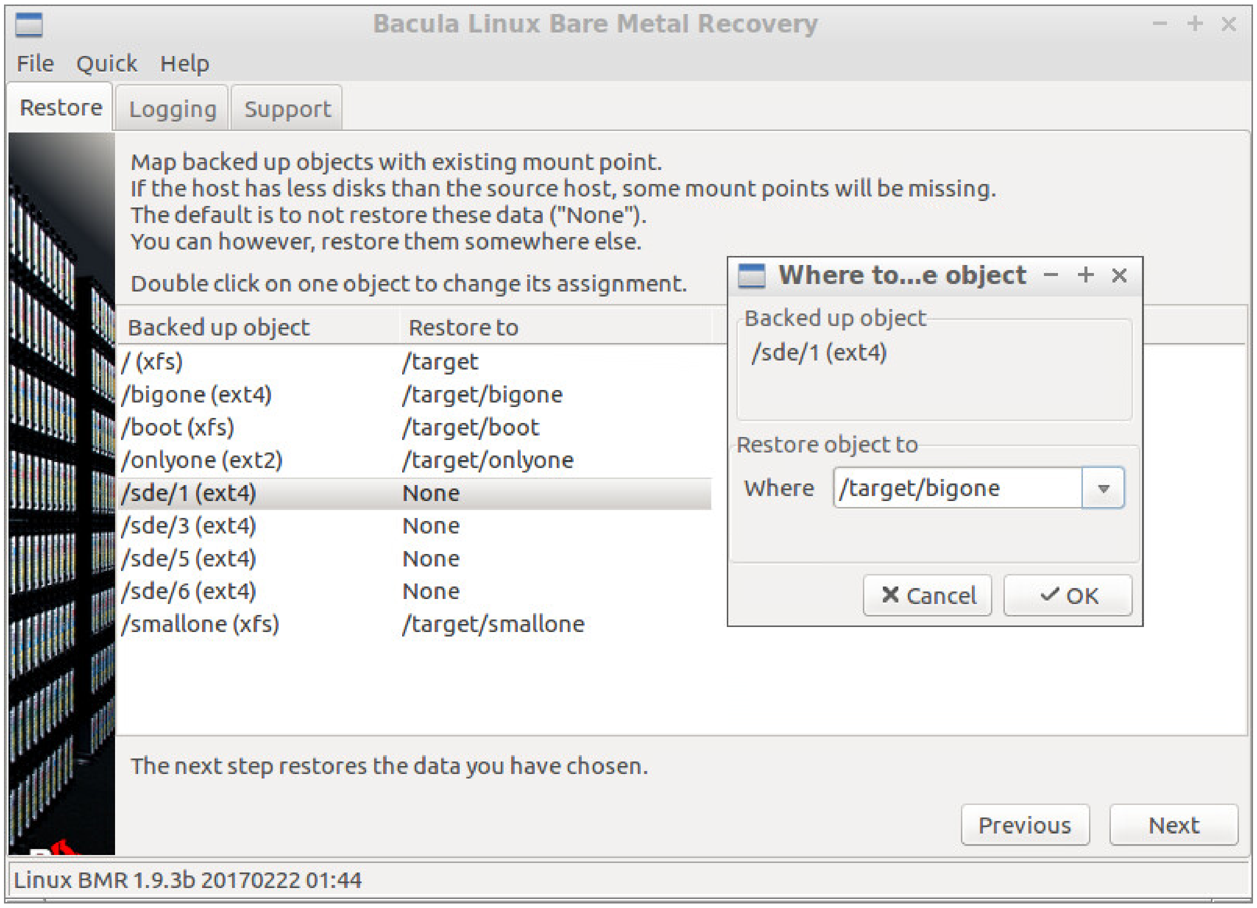
Linux BMR 1.9.3b 20170222 01:44

Fig. 15: Manual Partitioning



You are going to restore your data in "/target".
This directory must be ready.
Do you want to continue ?

Fig. 16: `/target` Must be Ready to get the Data



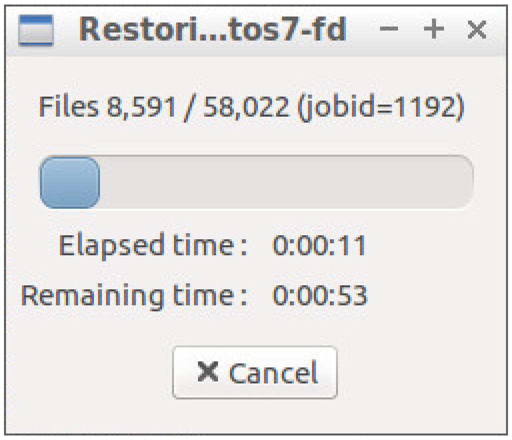
Map backed up objects with existing mount point.
If the host has less disks than the source host, some mount points will be missing.
The default is to not restore these data ("None").
You can however, restore them somewhere else.

Double click on one object to change its assignment.

Backed up object	Restore to
/ (xfs)	/target
/bigone (ext4)	/target/bigone
/boot (xfs)	/target/boot
/onlyone (ext2)	/target/onlyone
/sde/1 (ext4)	None
/sde/3 (ext4)	None
/sde/5 (ext4)	None
/sde/6 (ext4)	None
/smallone (xfs)	/target/smallone

The next step restores the data you have chosen.

Fig. 17: Volume Mapping



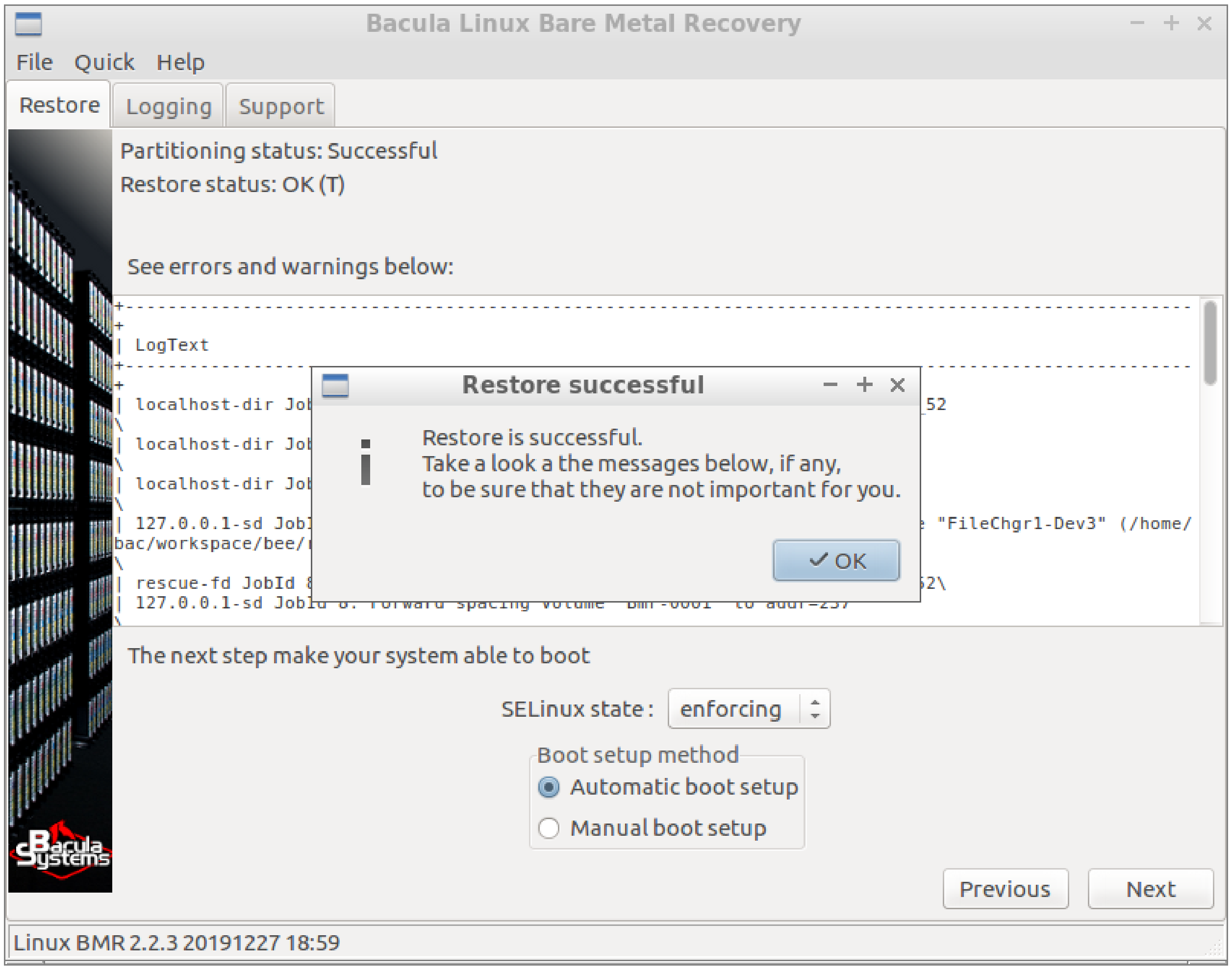
Files 8,591 / 58,022 (jobid=1192)

Elapsed time : 0:00:11
Remaining time : 0:00:53

Fig. 18: Restore Gauge

5.6 Restore and Boot Loader Configuration

Once restore of the data is complete, the BMR program invites you to look at the joblog of the restore job. Even if a successful restore would be good news, the log should be reviewed for warnings or other minor errors. The same screen in figure *After Restore* asks to choose between automatic and manual configuration of the boot loader.



Partitioning status: Successful
Restore status: OK (T)

See errors and warnings below:

```
+----------------------------------------------------------------------------------------------------
+
| LogText
+----------------------------------------------------------------------------------------------------
+
| localhost-dir Jo                                                                  52
\
| localhost-dir Jo
\
| localhost-dir Jo
\
| 127.0.0.1-sd Job                                                     "FileChgr1-Dev3" (/home/
bac/workspace/bee/
\
| rescue-fd JobId                                                                   2\
| 127.0.0.1-sd JobId 8: Forward spacing Volume "bmr-0001" to addr=237
\
```

Restore is successful.
Take a look a the messages below, if any,
to be sure that they are not important for you.

The next step make your system able to boot

SELinux state : enforcing

Boot setup method
- [✓ Automatic boot setup]
- Manual boot setup

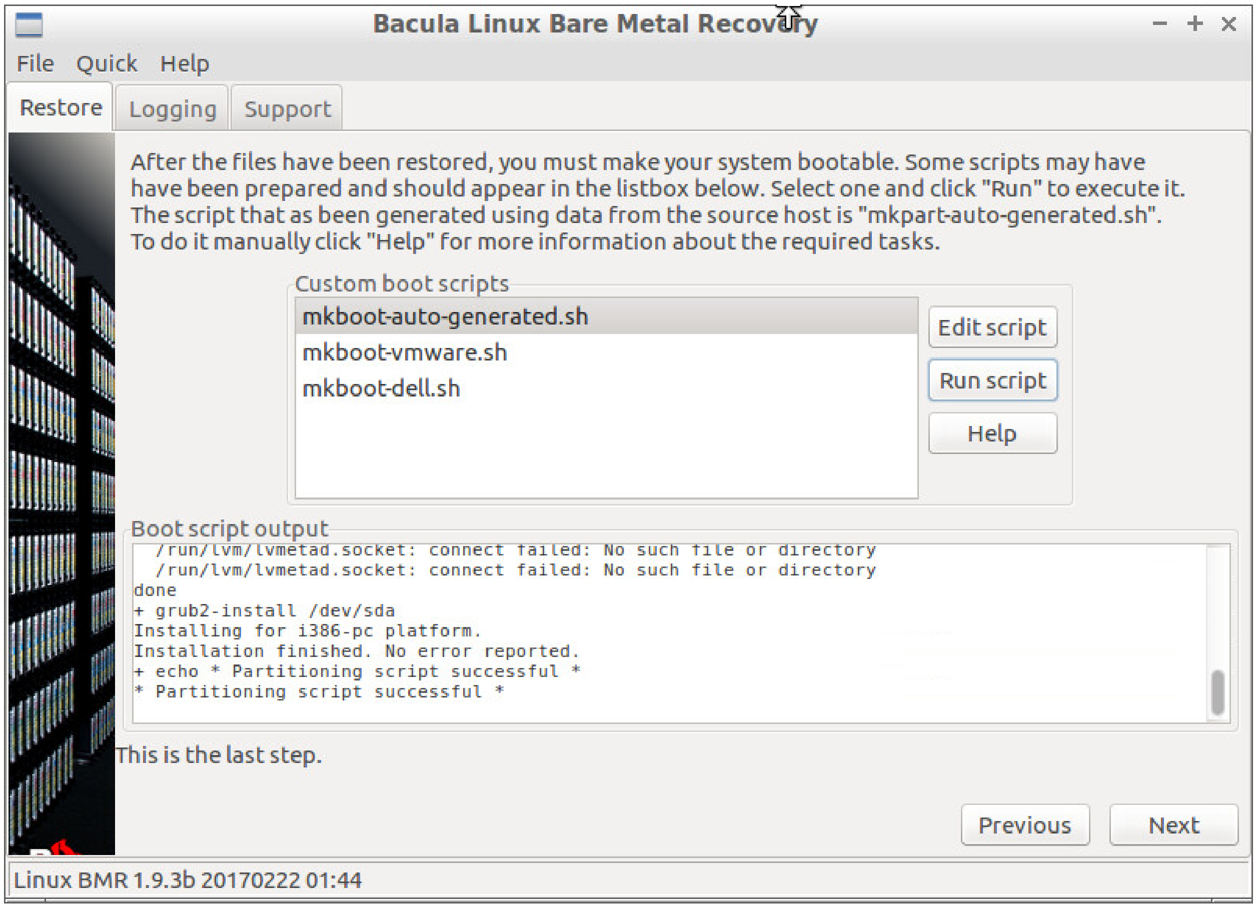
Linux BMR 2.2.3 20191227 18:59

Fig. 19: After Restore

The process is similar to the partitioning process seen before: The BMR program generates a shell script. If you choose "Manual boot setup" or if the auto-generated script has caused errors then the program enters the Boot Loader Manual Configuration screen presented in figure *Boot Loader Manual Configuration*.

You can change the SELinux mode using the combo box. This can be convenient if you have some problems at reboot time. Have a look at section *Restoring a system with SELinux enabled*.

The script `mkboot-auto-generated.sh` in the list is the one that the BMR program has generated. The two other scripts are here to remind you that you can prepare your own scripts in advance and store them in the `/opt/bacula/rescue/scripts` directory of the source host. These custom scripts will be available in the list if their names start with `mkboot-`. You can edit and run the scripts using the buttons on the right. The output is displayed in the small window at the bottom. The "Help" button provides a lot of information about what you have to do, how you can do it, and all the details about the source system.



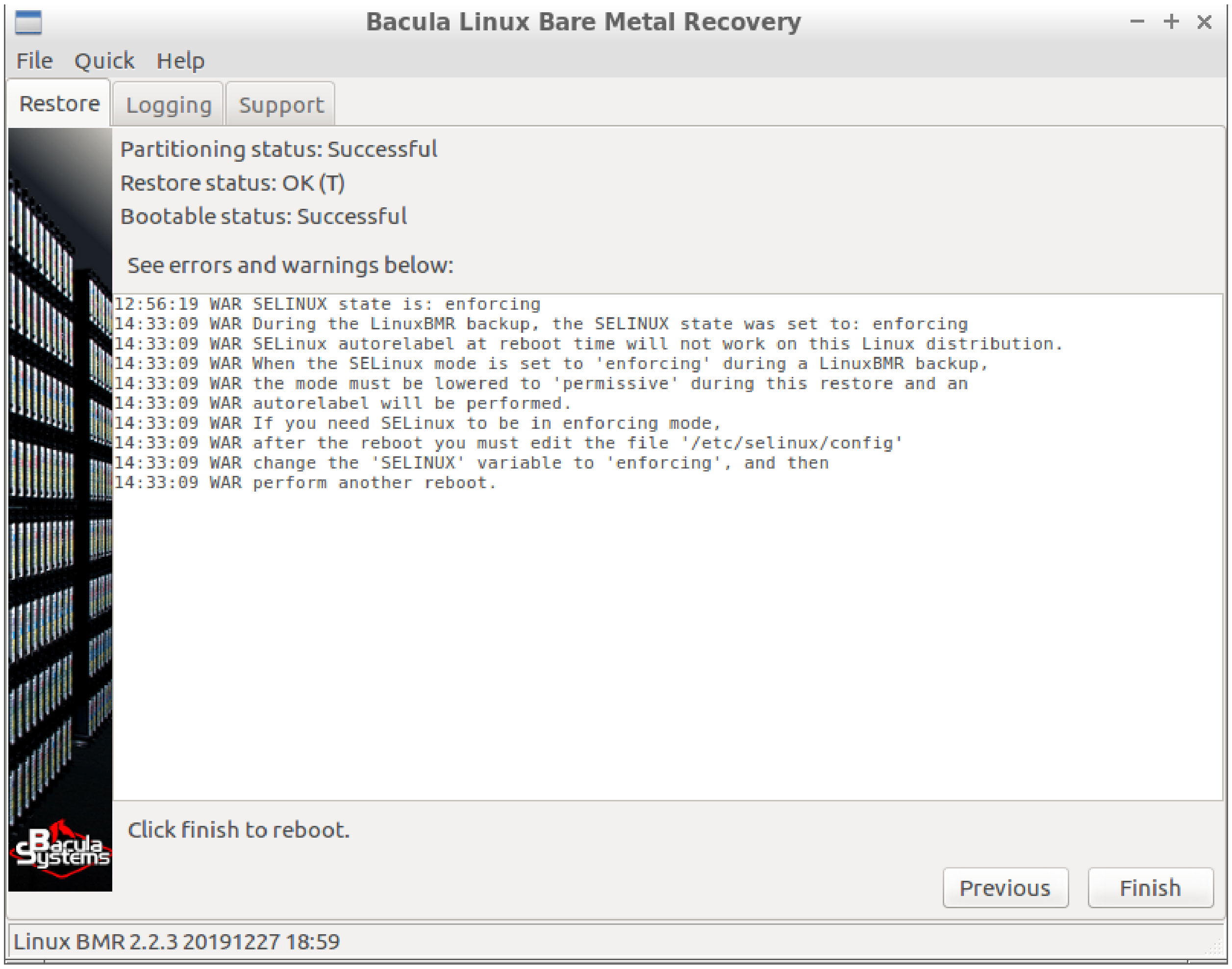
After the files have been restored, you must make your system bootable. Some scripts may have have been prepared and should appear in the listbox below. Select one and click "Run" to execute it. The script that as been generated using data from the source host is "mkpart-auto-generated.sh". To do it manually click "Help" for more information about the required tasks.

```
  /run/lvm/lvmetad.socket: connect failed: No such file or directory
  /run/lvm/lvmetad.socket: connect failed: No such file or directory
done
+ grub2-install /dev/sda
Installing for i386-pc platform.
Installation finished. No error reported.
+ echo * Partitioning script successful *
* Partitioning script successful *
```

This is the last step.

Fig. 20: Boot Loader Manual Configuration

5.7 BMR Final



Partitioning status: Successful
Restore status: OK (T)
Bootable status: Successful

See errors and warnings below:

```
12:56:19 WAR SELINUX state is: enforcing
14:33:09 WAR During the LinuxBMR backup, the SELINUX state was set to: enforcing
14:33:09 WAR SELinux autorelabel at reboot time will not work on this Linux distribution.
14:33:09 WAR When the SELinux mode is set to 'enforcing' during a LinuxBMR backup,
14:33:09 WAR the mode must be lowered to 'permissive' during this restore and an
14:33:09 WAR autorelabel will be performed.
14:33:09 WAR If you need SELinux to be in enforcing mode,
14:33:09 WAR after the reboot you must edit the file '/etc/selinux/config'
14:33:09 WAR change the 'SELINUX' variable to 'enforcing', and then
14:33:09 WAR perform another reboot.
```

Click finish to reboot.

Linux BMR 2.2.3 20191227 18:59

Fig. 21: BMR Final

You have reached the final screen. It shows the status of all important operations and all events with a level or "warning" or "error" that the BMR program has encountered. Figure *BMR Final* also shows how mapped disk contents have been handled which caused some lines to be removed from the `fstab` file accordingly.

Before clicking the "Finish" button to reboot the system, you can do some interesting things:

You can inspect the restored data using any of the tools available on the Recovery system, or using a simple terminal. The restored data is in the the `/target` directory.

You can do some operations inside the restored system. Start a terminal, become root using the command `sudo su -` then type command `chroot /target` to operate in the freshly restored system like if it was your root system.

You can generate and download the report of the full BMR process in a file `bssupport.zip` that you could send to our support in case your system would not reboot or would not meet your expectations. The report can be generated in the "Support" tab described in section *The Support tab*.

5.8 Restoring a system with SELinux enabled

The BMR program can restore *SELinux* enabled systems. By default, the program creates the file `/.autorelabel` and lets the system reset all *SELinux* attributes at first reboot. The system often reboots a second time after the reset of the attributes.

Unfortunately, for unknown reasons this procedure doesn't work anymore with some recent Linux distributions when the *enforcing* mode is used. The workaround is to switch the mode to *permissive* just after the restore using the *combobox* in figure *Boot Loader Manual Configuration*, and let the system reset the attributes.

When the system is rebooted for the second time, the administrator has to login and edit the file `/etc/syslinux/selinux` to change the *SELinux* mode back to *enforcing* before rebooting one final time.

If you have problems at reboot time and you know that the *SELinux* mode is set to enforcing, then try to lower the mode to *permissive* as described above.

The BMR program detects this situation for Redhat Enterprise Linux 8 and automatically lowers the *SELinux* mode to *permissive* while configuring the *BOOT* and warns you in the log displayed just before reboot. In any case, a warning is displayed when *SELinux* mode was set to *enforcing* on the source system to get your attention about this problem.

The problem is known to show up also on Debian 10 but it is not yet handled by the BMR program, this up to you to change the *SELinux* mode in the dialog box, and follow the procedure.

5.9 More about the GUI

File Menu

The File menu shown in figure *The File Menu* gives you access to a "Root terminal" and lets you "Exit" the program or "Reboot" the system.

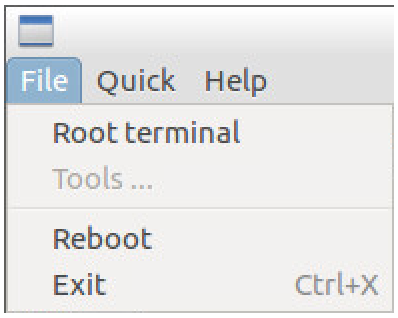


Fig. 22: The File Menu

Quick Menu

The Quick menu in figure *The Quick Menu* can open the log file of the BMR program.

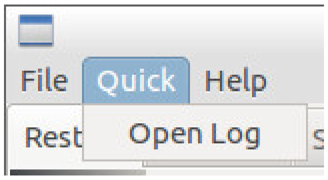
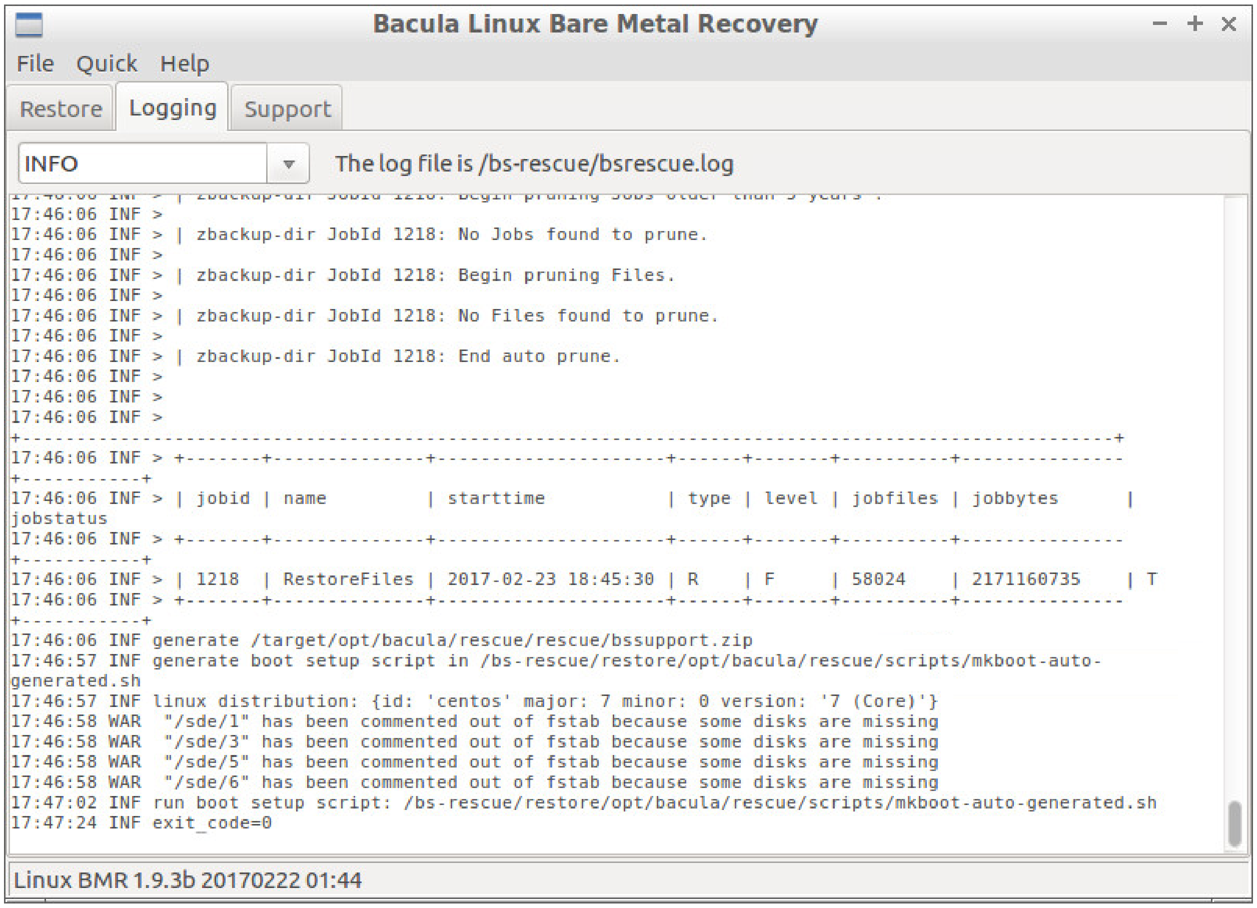


Fig. 23: The Quick Menu

Restore Tab

The restore tab has been described in the previous sections.

Logging Tab


Bacula Linux Bare Metal Recovery
File Quick Help
Restore Logging Support
INFO
The log file is /bs-rescue/bsrescue.log
Linux BMR 1.9.3b 20170222 01:44

```
17:46:06 INF >
17:46:06 INF > | zbackup-dir JobId 1218: No Jobs found to prune.
17:46:06 INF >
17:46:06 INF > | zbackup-dir JobId 1218: Begin pruning Files.
17:46:06 INF >
17:46:06 INF > | zbackup-dir JobId 1218: No Files found to prune.
17:46:06 INF >
17:46:06 INF > | zbackup-dir JobId 1218: End auto prune.
17:46:06 INF >
17:46:06 INF >
17:46:06 INF >
+-----------------------------------------------------------------------------------------------------+
17:46:06 INF > +-------+-------------+--------------------+------+-------+----------+--------------
+-----------+
17:46:06 INF > | jobid | name        | starttime          | type | level | jobfiles | jobbytes      |
jobstatus
17:46:06 INF > +-------+-------------+--------------------+------+-------+----------+--------------
+-----------+
17:46:06 INF > | 1218  | RestoreFiles | 2017-02-23 18:45:30 | R    | F     | 58024    | 2171160735    | T
17:46:06 INF > +-------+-------------+--------------------+------+-------+----------+--------------
+-----------+
17:46:06 INF generate /target/opt/bacula/rescue/rescue/bssupport.zip
17:46:57 INF generate boot setup script in /bs-rescue/restore/opt/bacula/rescue/scripts/mkboot-auto-
generated.sh
17:46:57 INF linux distribution: {id: 'centos' major: 7 minor: 0 version: '7 (Core)'}
17:46:58 WAR  "/sde/1" has been commented out of fstab because some disks are missing
17:46:58 WAR  "/sde/3" has been commented out of fstab because some disks are missing
17:46:58 WAR  "/sde/5" has been commented out of fstab because some disks are missing
17:46:58 WAR  "/sde/6" has been commented out of fstab because some disks are missing
17:47:02 INF run boot setup script: /bs-rescue/restore/opt/bacula/rescue/scripts/mkboot-auto-generated.sh
17:47:24 INF exit_code=0
```

Fig. 24: The Logging Tab

Next to the “Restore” tab is the “Logging” tab that displays all log messages of the BMR program. Messages can be filtered by level.

Support Tab

The “Support” tab presented in figure *The Support tab* allows the generation of a report file `bssuport.zip` that contains information required by our support team to assist. The file can be uploaded directly to Bacula Systems’ ticketing system (using the https protocol), or downloaded through HTTP protocol via a built-in server. When uploading, if you already have a ticket open, then supply the ticket number when sending the file to Support. If not, use any identifier that can help the support team to easily identify your report.

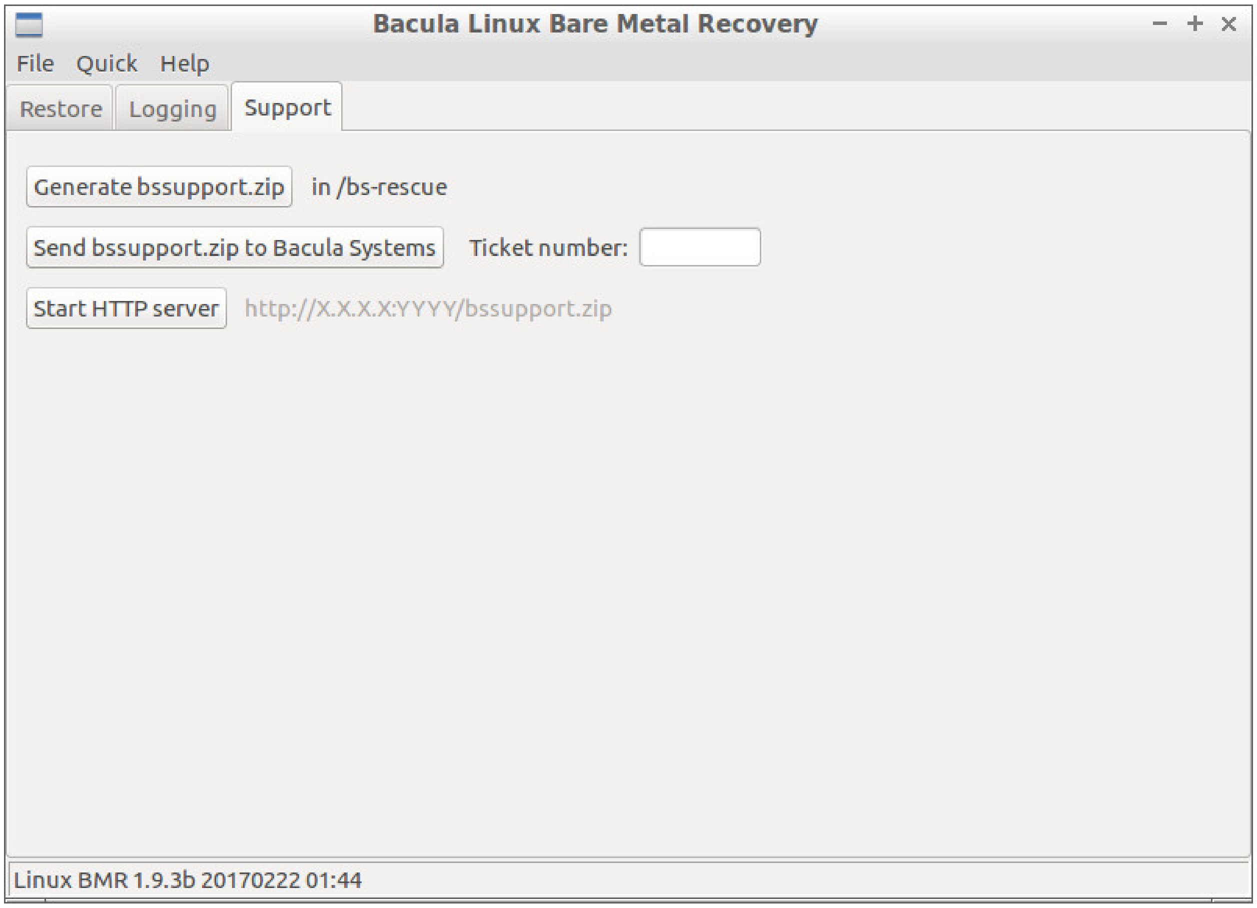
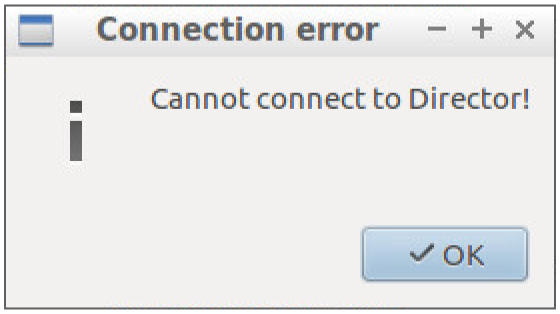

Bacula Linux Bare Metal Recovery
File
Quick
Help
Restore
Logging
Support
Generate bssupport.zip
in /bs-rescue
Send bssupport.zip to Bacula Systems
Ticket number:
Start HTTP server
http://X.X.X.X:YYYY/bssupport.zip
Linux BMR 1.9.3b 20170222 01:44

Fig. 25: The Support tab

5.10 Connection Troubleshooting

Network problems or wrong values in the form shown in figure *Bacula Configuration*, can cause different errors. Here are some hints to solve them.

If you get the error message in figure **fig:bmr-config-no-dir-connect**, then you have a problem connecting to your **Director**. You can try to `ping` or `telnet` to your **Director**, or review the Director's fields in the configuration form.

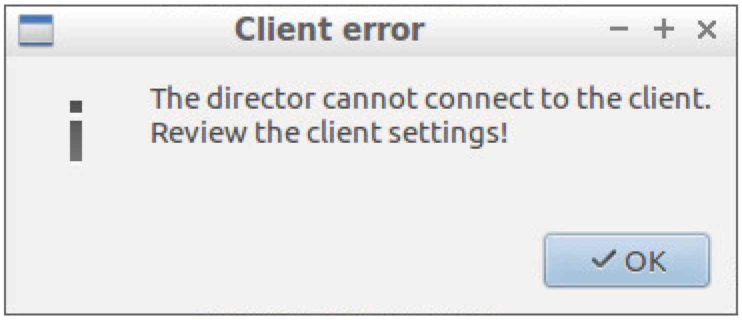


Connection error

Cannot connect to Director!

Fig. 26: Something is Wrong Connecting to the Director

If you get the error message shown in the screenshot below, then the **Director** can not reach your client. Try to `ping` or `telnet` to the client from the **Director** itself and compare the rescue client fields in the configuration form with the ones on the Director side.

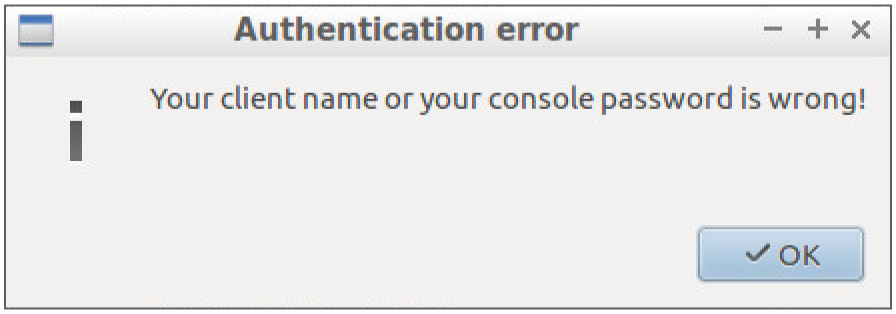


Client error

The director cannot connect to the client.
Review the client settings!

Fig. 27: Something is Wrong in the Client Configuration

If you get the error message in figure *Something is Wrong in the Console Configuration*, then the "Rescue client name" or "Rescue console password" do not match the ones used in your Console resource of the **Director**, as shown in figure **fig:cons**.



Authentication error

Your client name or your console password is wrong!

Fig. 28: Something is Wrong in the Console Configuration

6 Compatibility

- The LinuxBMR runs with **Bacula Enterprise** versions 6 and newer.
- The LinuxBMR 2.2.4 version is compatible with the Bacula-rescue.sh script version 2.4.0 available from Bacula Enterprise 12.4.2 and in the LinuxBMR Plugin download area.
- The LinuxBMR has been successfully tested on RedHat and Centos Linux 5, 6, 7, and 8, SUSE Linux Enterprise Server 11 with Grub Legacy, SUSE Linux Enterprise Server 12 with at least Service Pack 1 (btrfs is not supported), as well as Debian Linux 7, 8, 9, and 10, and Ubuntu LTS 14.04, 16.04, 18.04, and 20.04. Only 64bit versions of these distributions have been tested.
- The LinuxBMR supports `LVM`. LVM's advanced and `RAID` functionalities are not supported. *Multipath* is also not supported.
- `LVM` unallocated space is preserved and handled like an unused *Logical Volume* to preserve snapshot functionality.
- LinuxBMR supports the following filesytems: xfs, ext2, ext3, ext4. MSDOS / FAT are supported for UEFI boot partitions. zfs, btrfs and others are not.
- DOS and GPT partition tables are supported.
- Linux Software `RAID` and `LVM` on RAID will be supported in a future version.
- Partition encryption is not supported (this is planned for a future release).
- The `restart` command has limitations with plugins, as it initiates the Job from scratch rather than continuing it. Bacula determines whether a Job is restarted or continued, but using the `restart` command will result in a new Job.
- Ask our Support Team for up to date information about supported and unsupported features. The most frequently requested features will be prioritized.

7 Important Considerations

The LinuxBMR is designed to be an extensible framework, not a static one-size-fits-all solution. Unsupported or problematic features can be handled manually if needed. In particular the disk partitioning and boot manager configurations have a manual mode. The scripts generated by the BMR using information from the source system can be used as an example to help the user.

Due to the very large compatibility matrix between hardware and software that LinuxBMR should support, LinuxBMR needs careful testing and configuration on actual production-like systems before it can become part of a reliable production-ready Disaster Recovery solution. Once a combination of hard- and software (for example, DELL R620 Perc H800 with Redhat 7 64bit) is validated, the product can safely be used on all servers that are using that combination.