

Backups & Restores

- [Full Server Restore \(Dedi or Virtual\)](#)
- [Commvault Backups](#)
- [Bacula Restores](#)
- [Commvault Restores](#)
- [Permissions Restores](#)
- [Bacula Bextract](#)
- [Bacula Backups](#)

Full Server Restore (Dedi or Virtual)

Full Server Restores

Commvault

Bare-Metal Server

https://kb.ukfast.net/Linux_Server_Replacement_Procedure

1. Reinstall the server to clean OS, or if needed rack up a new server, have networks configure the IP and then run through autoinstaller.
2. Once server is installed, and configured as intended, we need to [install the commvault agent](#) to allow for commserve to reach the server.
I'd also advise looking to perform a 'readiness check' via Commgui.

Client servers > SID >

3. Once commvault agent is installed and confirmed to be working, you can look to being the restore process. The first step of the restore process is to run a restore of only the below directories, without using the 'unconditional overwrite' option:

```
/usr/lib64
/usr/lib
```

4. Once the above initial restore has been completed, we can then look to perform the main restore. This needs to be run with the below directories excluded:

```
/opt/commvault
/proc
/etc/sysconfig/network-scripts
/etc/fstab
/etc/mtab
/etc/udev.d
/boot
```

```
/usr/lib64
/usr/lib
```

voila that *should* be it, however, these sort of things never seem to go as planned, so I'd advise reviewing the aforementioned KB articles for further information if needed.

Side note, you also NEED to make sure that any required cleanup is done, ie making sure all appropriate software is installed on sid (threatmon etc), and that any old SID/servers are appropriately removed.

Virtual Machines

VPC

=====

=====

Bacula

Commvault Backups

=====

=====

Installing Commvault Agent on Linux server

https://kb.ukfast.net/Installing_Commvault_Agent_on_Linux_Client

https://kb.ukfast.net/Launching/Installing_Commvault

1. Ensure that the **Commvault ports** are open on the client firewall:

```
telnet 81.201.136.241 8400
telnet 81.201.136.241 8600
```

2. SSH into the Client server you wish to install the Commvault agent on. Review the backup requirements (File only/File & MySQL etc) and run the appropriate command in the terminal.

Run the following command and complete the wizard

```
bash <(curl -s http://80.244.178.135/Linux/CommVault/installer-script/commvault_install.sh)
```

If the MA is v10, use the v10 download command as v10 MA's do not support v11 clients

UPDATE THE FLAGS : -clientname idxxxxx -clienthost idxxxxx.cvbackup.ukfast

v11 all packages **UPDATE ID's**

```
mkdir CommvaultInstallMedia && cd CommvaultInstallMedia && wget
http://80.244.178.135/Commvault/v11/CommCell_F9F70_Main/Linux_X86_64_Custom_Seed_Package_FR20-
MR27_F9F70/Custom_Seed_Package_FR20-MR27_F9F70.tar && tar -xvf Custom_Seed_Package_FR20-
MR27_F9F70.tar && cd ./pkg && ./cvpkgadd -silent -clientname idXXXXX -clienthost
idXXXXX.cvbackup.ukfast && history -d $(history | tail -n 1) && cd ~ && rm -rf
CommvaultInstallMedia/;
```

v10 download command:

```
rm -rf CommvaultInstallMedia && mkdir CommvaultInstallMedia && cd CommvaultInstallMedia && wget
```

3. Confirm the agent is installed and running by the following Commands:

```
simpana list
```

```
simpana status
```

Check that the install media has been removed from the server and the history to ensure no evidence of the command run.

```
history
```

Alongside the wget command will be a "line number". Run the following to delete this line:

```
history -d LINE-NUMBER
```

```
=====
```

Troubleshooting Commvault Backup Issues

commvault has a rather useful 'readiness check' which basically just checks the backup system for any errors:

CommGUI > Client Computers > ctrl+f > SID > right click > all tasks > check readiness

Bacula Restores

Useful Bacula commands

Watch restore job:

```
watch -n1 'echo "status client client=\srv-IP_IP_IP_IP\" | bconsole'
```

Check backup client status via bconsole:

```
status client
```

```
=====
```

Bacula File Restore

1. SSH onto backup server.

2. Initiate a restore

Run the following commands to initiate a restore using the backup jobID:

Bconsole > restore > 3 > enter Job IDs, comma seperated.

3. Mark files to be restored

Once the directory tree has been built we need to mark the /var/lib/mysql directory, and initiate the restore:

```
mark /path/to/file
```

Run 'done' once all required files have been marked

```
done
```

Select option 9 (where):

```
9
```

Once here, you need to enter the directory we're restoring TO:

```
/root/restore_TICKETNUMBER
```

```
=====
=====
```

Bacula DB Restore

[https://kb.ukfast.net/Restore MySQL from Bacula](https://kb.ukfast.net/Restore_MySQL_from_Bacula)

1. SSH onto backup server.

2. Initiate a restore

Run the following commands to initiate a restore using the backup jobID:

```
Bconsole > restore > 3 > enter Job IDs, comma seperated.
```

3. Mark files to be restored

Once the directory tree has been built we need to mark the /var/lib/mysql directory, and initiate the restore:

```
mark /var/lib/mysql
```

Run 'done' once all required files have been marked

```
done
```

Select option 9 (where):

```
9
```

Once here, you need to enter the directory we're restoring to (on client server):

```
/root/restore_TICKETNUMBER
```

4A. Starting 2nd MySQL Instance

Once MySQL has been restored onto client server, we then need to start a 2nd instance of mysql so that we can dump the required databases.

The below command starts the 2nd MySQL instance (You need to replace /mnt/mysql with the path we've restored MySQL to; in my example this is /root/restore_TICKETNUMBER).

```
/usr/sbin/mysqld --socket=/tmp/mysql2.sock --datadir=/mnt/mysql --skip-networking --pid-file=/tmp/mysql2.pid --user=mysql --skip-grant-tables
```

Running this command will take over your session, meaning that you'll have to leave this running and open a fresh SSH session.

If you're encountering errors when attempting to start the 2nd instance, it would be worth having a google of the errors. If you're still having issues, please see below:

4B. Troubleshooting restore MySQL startup issues

If you're unable to start the 2nd MySQL instance after troubleshooting, there's 2 main options:

1. Delete the restored content on the client server and start the restore again
2. If this still doesn't work, you can try to start the 2nd MySQL instance using `innodb_force_recovery`.

There are 6 levels of force recovery options, see [here](#) for full details.

```
/usr/sbin/mysqld --socket=/tmp/mysql2.sock --datadir=/mnt/mysql --skip-networking --pid-file=/tmp/mysql2.pid --user=mysql --skip-grant-tables --innodb_force_recovery=X
```

You'll need to replace 'x' on the above command with your chosen level, I'd advise starting with level 1 and moving up until the MySQL instance is started. Anything above level 4 can cause permanent data corruption, so it's ideal if we can avoid this.

If `innodb_force_recovery` is used, please note down the level used and tell the client about this.

<https://dev.mysql.com/doc/refman/8.0/en/forcing-innodb-recovery.html>

If errors are being shown regarding the existing MySQL configuration, you can attempt to launch the 2nd instance with the `--no-defaults` flag, this essentially tells MySQL to launch with the default settings :

```
/usr/sbin/mysqld --no-defaults --socket=/tmp/mysql2.sock --  
datadir=/home/restore_4450884/mysql --skip-networking --pid-file=/tmp/mysql2.pid --  
user=mysql --skip-grant-tables
```

5. Dumping a database

Now that we've got the restored MySQL instance running, we can look to validate the data we require is present, and dump out the database.

Connect to restored MySQL instance:

```
mysql -S /tmp/mysql2.sock
```

Check that the required database is present:

```
SHOW DATABASES;
```

Once we've confirmed that the databases we need is present, we can look to dump out the database into a file:

Run the following command, substituting 'databasename' with the name of your required database, and also updating 'database_restoreTICKETNUMBER' with the appropriate ticket number.

```
mysqldump -S /tmp/mysql2.sock databasename --events --triggers --routines >  
/root/database_restoreTICKETNUMBER.sql
```

Other options for dumping a database/s

All data and databases :

```
mysqldump -S /tmp/mysql2.sock --all-databases --events --triggers --routines >
/root/database_restoreTICKETNUMBER.sql
```

To dump several but not all databases (substitute databasename1,databasename2 etc) :

```
mysqldump -S /tmp/mysql2.sock --databases databasename1 databasename2 --events --triggers --
routines > /root/database_restoreTICKETNUMBER.sql
```

For a specific table only (substitute databasename and tablename)

```
mysqldump -S /tmp/mysql2.sock databasename tablename > /root/database_restoreTICKETNUMBER.sql
```

If you're encountering errors when attempting to dump the required data, you can use the -f flag to ignore errors.

6. Cleaning up

Once the dump is complete, terminate the 2nd instance:

```
mysqladmin -S /tmp/mysql2.sock shutdown
```

Remove the restored /var/lib/mysql directory.

```
rm -rf /root/restore_TICKETNUMBER
```

Commvault Restores

Commvault Restores

Commvault restore types:

[MySQL Restores](#)

File level restores

[VPC Restores](#)

=====

MySQL Restores

There are 2 types of database restore we can look to perform through commvault, depending on the type of backups in use.

If the client has MySQL level backups, see here

If the client only had file-level backups, see here

=====

MySQL level backups

Starting the restore job

Commgui > Client Servers > Ctrl+F > Search SID > Double Click MySQL > Double Click SID

Right click > All tasks > browse and restore > Untick Recover > Select Required Date using end time

Select Required Database from the list > Recover Selected

Post-restore action

Replaying bin logs

```
cp -a DBNAME DBNAME.sql
```

Check disk space before replaying

```
mysqlbinlog -v --database="DBNAME" RESTOREPATH/BINFILENAME.* >> RESTOREPATH/DBNAME.sql
```

Once the bin logs have been replayed, we need to remove USE statements:

```
grep -i ^USE DBNAME  
sed -i '/^[uU][sS][eE] /d' DBNAME
```

Replaying bin logs to a certain time:

https://kb.ukfast.net/MySQL#Bin_Logs

Once the bin logs have been replayed, we can look to remove the bin logs and older db copy

Restoring a specific table from DB level backups

Identify the line number of the desired table in the dump file:

```
grep -in 'dumping data for' dumpfile.sql
```

Next, we need to remove the lines before and after the desired table

```
sed -i '10611,28206d' m2meandemcom.sql
```

The 2 numbers here are the line number of the next table (shown in the original grep command) and the final tables line number (also in the dump)

Next we do the same for the table line numbers before the one we need:

```
sed -i '51,10581d' m2meandemcom.sql
```

```
=====
```

File level database restore

=====

=====

File Level Restores

=====

=====

eCloud VPC Restores

Commvault VM Level Backup Restore:?

CommGUI > Client Computers > ctrl+f > VM ID/VPC Instance ID > right click > browse and restore > Full Virtual Machine > Select End time (if required) > Browse > Select VM > Restore full VM > Select Restore DataStore, change name to VMID_TICKETNUMBER> Go

Vmware > search for restore VM (VMID_Ticket)

Edit settings (along the top)

1. Disable Networking

vSphere > VM > Edit Settings > Network Adapter > Disable 'Connect at power on'

2. Power on restore VM

vSphere > VM > Actions > Power > Power On

3. Add swing disk

Confirm how much space you will require for a swing disk to be used to move data over to the live VM.

vsphere > VM > Edit Settings > Add New > Hard Disk > OK

Once the disk has been added in vSphere, you'll need to configure this on the server.

Use `mkfs` to initialise the disk filesystem.

`mkfs /dev/sdX`

Mount the disk once configured.

```
mount /dev/sdX /mnt
```

4. Move requested data onto disk

5. Unmount the disk from restore VM

'Once the data has been moved onto the disk, unmount the disk from the server filesystem:

```
umount /mnt
```

And, unmount the disk from the restore VM within vSphere

```
vSphere > Edit Settings > X symbol next to disk we've added. Don't click 'Delete files from datastore'.
```

5. Transfer the disk to the live VM

Mount disk on live VM in vSphere

```
Edit Settings > Add New Device > Existing Hard Disk > Datastores > Restores > Identify restore VM > Add restore swing disk
```

Mount the disk on live server filesystem

```
mount /dev/sdX /mnt
```

6. Move file away from swing disk, into the required location for the client

7. Clean up

Unmount the disk from the live VM

```
umount /dev/diskname
```

Vsphere > edit settings > remove disk > Select 'delete disk from datastore'

Delete the restore VM

vSphere > Restore VM > Actions > Delete from disk > yes

Removing Restore

Once the client has finished with the restore, the disk will need to be unmounted from the server and the restore VM can be deleted.

SSH onto client VPC > df -h > check for /restore > umount /restore

Once the disk has been unmounted:

Vmware > Client VM > Edit Settings > Restore Disk > Remove (X on right side)

Now delete the Restore VM:

Vmware > restore VM > Power Off > Actions > Delete From Disk

Attach disk

Access restored server

Lsblk - find device name

Mkfs /dev/(device name)

Mount /dev/(devicename) /mnt

Main server > edit settings > attach device > existing disk > find disk

On server:

Mount /dev/(devicename) /mnt

Move restored files onto server

Permissions Restores

=====
=====

Client chowned their server? did you chown a clients server? Never fear, fact is here.

https://kb.ukfast.net/Linux_file_permissions

=====
=====

We can use the set/get fact command set for restoring server file permissions. TLDR; getfact can be used to generate a list of file permissions that can then be implemented onto a selection of files using the setfact command.

The below command would be for creating a permissions.acl file (containing all permissions on a set of files) in /

```
getfact -Rp / > permissions.acl
```

The below command is for using the file created above to set permissions:

```
setfact --restore=permissions.acl
```

Bacula Bextract

Bextract

1. Get list of volumes from TechDB

To get an idea of what files you'll need, you need to examine the TechDB Backups tab. For each backup, a line will be printed which starts "Volume name(s):" - this should be in the following format:

```
srv-7813711322-3889
srv-7813711322-3882
srv-7813711322-3875
srv-7813711322-3868
srv-7813711322-3861
srv-7813711322-3854
srv-7813711322-3847
srv-7813711322-3891
```

2. Create bootstrap file

This is a text file containing a list of volumes for Bacula to read from, in the correct order, it's format should be as follows (oldest to newest):

```
Volume=srv-78_137_113_22-3847
Volume=srv-78_137_113_22-3854
Volume=srv-78_137_113_22-3861
Volume=srv-78_137_113_22-3868
Volume=srv-78_137_113_22-3875
Volume=srv-78_137_113_22-3882
Volume=srv-78_137_113_22-3889
Volume=srv-78_137_113_22-3891
```

There's a little cheat to make this step easier which is the below command that should be run from within the backups directory on backups server (typically /home/bacula/clientIP_IP_IP_IP):

```
ls -latr srv* | awk '{print $9}' | sed 's/\(.*\) /Volume=\1/' > bootstrap.bsr
```

3. Create includes file

The next thing to create is a file that describes which files you want to extract from the backups, this is just another text file, containing the paths to the files you want to restore, separated by linebreaks, e.g:

```
/path/to/required/files  
/path2/to/required/files
```

Note: This file is not needed if you need to restore everything within a backup file.

4. Create the output directory (on backup server)

Create a directory for the extract command to output the backup contents to. This needs to be on the backup server as we'll look to transfer the files to the client later on.

5. Running the command

There are 2 sets of commands here, differing for PyBaculaV2 and none-pybac backup server. (Most we deal with now have been upgraded to PyBacV2).

The only real difference between the 2 options is the storage type that's specified. For PyBacV2, you'll need to specify the device prefix, whereas for none-pybac, you'll need to specify the storage prefix.

PyBaculaV2:

View files included in backup (with includes.txt taken into account):

```
bls -b bootstrap.bsr -i includes.txt -pv device-IP_IP_IP_IP
```

Initiate bextract:

```
bextract -b bootstrap.bsr -i includes.txt -pv storage-IP_IP_IP_IP /path/to/restore/to
```

None-PyBac

View files included in backup (with includes.txt taken into account):

```
bls -b bootstrap.bsr -i includes.txt -pv storage-IP_IP_IP_IP
```

Initiate bextract:

```
bextract -b bootstrap.bsr -i includes.txt -pv storage-IP_IP_IP_IP /path/to/output/
```

6. Transfer files to client-server

Feel free to use your preferred file transfer method for this step, for the below I've detailed how to achieve this using rsync.

A. Compress the file before transfer

[tar guide](#)

```
tar -czf zipfilename.tar.gz filetoarchive
```

B. Transfer the files

```
rsync -r -e "ssh -p2020" /path/to/files/on/backup/server  
'root@[client:server:BAC:address]':/client/server/restore/path
```

C. Uncompress files on client-server

(extracts into current working directory)

```
tar -xzf archivename.tar.gz
```

Bacula Backups

Backup Server Setup (via automation)

USEFUL LINKS:

https://kb.ukfast.net/Backup_Server_Install

<https://linuxinstaller-man4.devops.ukfast.co.uk/dashboard/>

<https://linuxinstaller-man5.devops.ukfast.co.uk/dashboard/>

<https://awx.devops.ukfast.co.uk/#/home>

All steps here are done within TechDB - specifically on the SID page for the backup server, unless otherwise specified.

If this is a reinstall then you first need to schedule maintenance and task DC to replace the disks

```
server type: "backup server"  
server subtype: "Bacula v9 Server"  
Status: Awaiting Installation  
Role: "Backup Server"  
OS type: Ubuntu 20.04 x86_64
```

Add the OS > Software Installed > Select Ubuntu 20.04 x86_64 and Right Click > Add

Also remove any OS that is already present (Only for reinstalls)

get networks to assign you an IP Address and fill in the Hostname (not needed for reinstall)

Hardware > Configuration > Model > This needs to be set to the chassis (Most will be R320 - won't need changing for a reinstall unless chassis has been swapped)

The server must have a rack switch AND a backup switch connection (backup switch MUST be Gigabit)

Set the vlan to the installer vlan for the relevant datacentre: [MAN4: 516, MAN5: 316]

Click Save Edits
Click Configure Switch
Click Reset Server OS
deselect firewall information - this is for shared backup servers, these are not Firewalled

Select Save Edits

Open the virtual console in DRAC

Click Install/Re-install OS to start automation, you can watch the steps in automation history and use DRAC too
Reboot server

When server is booting hit f2 to go to system setup then System BIOS > Boot Settings > Change from BIOS to UEFI, can then exit this menu and server will reboot
While its booting hit f11 to go into the Boot Manager > UEFI Boot menu > NIC 1
After that then automation will do the rest for you (hopefully)

=====
=====

Bacula Client Setup

=====
=====

Troubleshooting Bacula

Bacula Networking

BAC Address

Firstly, check the client server has a bacula IPv6 address.

```
ip a
```

Will look something like the following (note the 'BAC' present around the middle of this address)

```
2a02:22d0:bac:0:3617:ebff:fef1:1f8
```

If the server does not have a v6 address, we need to check the backup NIC config and then try to restart the backup NIC.

IPv6 Network Restrictions

Ensure that the ip6tables service isn't running:

```
Systemctl status ip6tables
```

Check the ruleset:

```
ip6tables -S
```

Add required rules for Bacula IPv6 networking over ports 9102 and 9103:

```
ip6tables -I INPUT -i eth1/em2 -j ACCEPT
ip6tables -I OUTPUT -p tcp --dport 9102 -j ACCEPT
ip6tables -I OUTPUT -p tcp --dport 9103 -j ACCEPT
service ip6tables save && service ip6tables restart
```

Bacula-fd Configuration