



VM-based Setup on IBM Cloud

Virsec Security Platform 2.6

www.virsec.com
support@virsec.com



Contents

| | |
|------------------------------------|----|
| About this Document | 3 |
| 1. IBM Cloud - VM Creation..... | 3 |
| 2. Extending /var filesystem | 6 |
| 3. Installation..... | 10 |

About this Document

This document provides steps for the IBM Cloud environment setup. It also provides information about Docker and Docker Compose installation on Virtual Server instances.

The pre-requisites for VSP installation:

1. Virtual Server instances with:
 - a. Docker
 - b. Docker Compose
 - c. Network Security Group with the list of ports utilized by VSP components. Refer to the VM Installation Checklist for information on the ports

1. IBM Cloud - VM Creation

1. Access the IBM cloud Dashboard: <https://cloud.ibm.com> using valid credentials
2. Navigate to **VPC Infrastructure > Virtual Server Instances > Create Instance**



3. Provide **Instance Name**, select **Resource Group** and add **Tags**

Details

Name

Resource group ⓘ

Default ▾

[View all resource groups](#)

Tags ⓘ

customer:virsec X

- Configure Security Groups to allow ports **22**, **80** and **443**. The snapshot is intended for reference only, since it allows the ports for any source IP Address

| | | | | |
|-----|-----|-----------|---------------|---|
| TCP | Any | 0.0.0.0/0 | Ports 80-80 | ↓ |
| TCP | Any | 0.0.0.0/0 | Ports 443-443 | ↓ |
| TCP | Any | 0.0.0.0/0 | Ports 22-22 | ↓ |

- Click **Create virtual server** to launch the Instance

Virtual server instance \$0.455/hr

- 8 vCPUs
- 32 GiB RAM
- 300 GB instance storage
- 16 Gbps

Image \$0.120/hr

- Red Hat Enterprise Linux

Boot volume \$0.012/hr

- 100 GB

Network interface provided

Apply a code

Subtotal \$428.58

Sustained usage discount -\$32.26

Total estimated cost \$396.32/mo

[Create virtual server](#)

[Get sample API call](#)

[Add to estimate](#)

- After a successful instance creation, navigate to **Virtual Server Instances** and search for the newly created instance

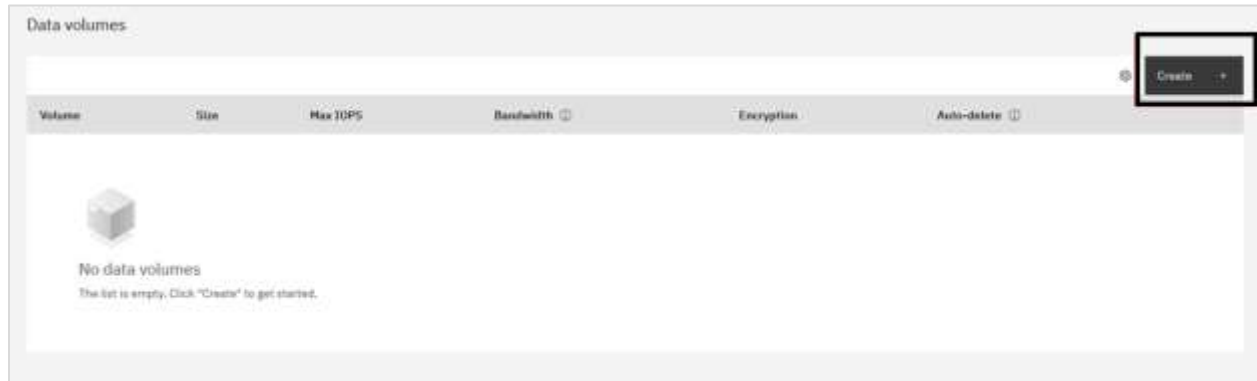
Virtual server instances for VPC

Region: Frankfurt

| Name | Status | Resource group | Virtual Private Cloud | Profile | Reserved IP | Floating IP |
|--------|---------|----------------|-----------------------|-----------|-------------|-------------|
| vpc-vm | Running | | | bx28-8x32 | | |

2. Extending /var filesystem

1. Create a new Virtual Server with an additional drive or add a new drive after creation
 - a. Add a new drive at VM creation



- b. Provide the required values. Click **Create**

Create data volume

Volume details

Name:

Resource group: Location:

Throughput: Auto-delete: Disabled

Size:
Enter a number between 10 and 10000 GB.

Profile:

Encryption:

- c. The newly created drive is now listed

| Volume | Size | Max IOPS | Bandwidth | Encryption | Auto-delete |
|--------|--------|----------|-----------|------------------|-------------|
| drive1 | 300 GB | 3000 | 393 Mbps | Provider managed | Enabled |

2. Log in into the VM and execute the command below to find the additional drive:

a. `fdisk -l`

```
[root@rhel-var ~]# fdisk -l

Disk /dev/vda: 107.4 GB, 107374182400 bytes, 209715200 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000cf734

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1    *          2048     1050623     524288   83   Linux
/dev/vda2                1050624    209715199    104332288   8e   Linux LVM

Disk /dev/vdb: 300.0 GB, 300000002048 bytes, 73242188 sectors
Units = sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/vdc: 0 MB, 391168 bytes, 764 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/vdd: 0 MB, 45056 bytes, 88 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/Vo1Group-lv_root: 106.3 GB, 106321412096 bytes, 207659008 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/Vo1Group-lv_swap: 511 MB, 511705088 bytes, 999424 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

[root@rhel-var ~]#
```

b. Partition the disk using the command:

i. `fdisk /dev/vdb`

```
[root@act-cms-vm-2 ~]# fdisk /dev/vdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x3b93e8d7.

Command (m for help): █
```

ii. Use option `'n'` to create the partition

iii. Use `'w'` command to save

- iv. Use the commands depicted in the screenshot below:

```
[root@rhel-var ~]# fdisk /dev/vdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x2299b078.

Command (m for help): n ← Type 'n'
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p ← Type 'p'
Partition number (1-4, default 1):
First sector (256-73242187, default 256):
Using default value 256
Last sector, +sectors or +size{K,M,G} (256-73242187, default 73242187):
Using default value 73242187
Partition 1 of type Linux and of size 279.4 GiB is set

Command (m for help): t ← Type 't'
Selected partition 1
Hex code (type L to list all codes): 8e ← Type '8e'
Changed type of partition 'Linux' to 'Linux LVM'

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@rhel-var ~]#
```

- v. Press “Enter” to choose default values on sectors

3. Execute the command below to check the mounted disk:

- a. `lsblk`

```
[root@rhel-var ~]# lsblk
NAME                MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
vda                  252:0    0  100G  0 disk
├─vda1                252:1    0   512M  0 part /boot
└─vda2                252:2    0  99.5G  0 part
   └─VolGroup-lv_root 253:0    0   99G   0 lvm  /
      └─VolGroup-lv_swap 253:1    0  488M  0 lvm
vdb                  252:16   0  279.4G  0 disk
└─vdb1                252:17   0  279.4G  0 part
vdc                  252:32   0    382K  0 disk
vdd                  252:48   0    44K   0 disk
[root@rhel-var ~]#
```


4. Modify the file system to `ext4` using the command:

- a. `mkfs -t ext4 /dev/vdb1`

```
[root@rhel-var ~]# mkfs -t ext4 /dev/vdb1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
18317312 inodes, 73241932 blocks
3662096 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2220883968
2236 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000, 23887872, 71663616

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

[root@rhel-var ~]#
```

5. Rename `var` directory using the command:

- a. `mv /var /var2`

```
[root@rhel-var ~]# mv /var /var2
[root@rhel-var ~]# ls /
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var2
[root@rhel-var ~]#
```

6. Create a new directory for `var` using the command:

- a. `mkdir /var`

```
[root@rhel-var ~]# mkdir /var
[root@rhel-var ~]# ls /
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var var2
[root@rhel-var ~]#
```

7. Mount created partition with `/var` using the command:

- a. `mount /dev/vdb1 /var`

```
[root@rhel-var ~]# mount /dev/vdb1 /var
[root@rhel-var ~]# df -h
Filesystem              Size  Used Avail Use% Mounted on
devtmpfs                 16G   0    16G   0% /dev
tmpfs                    16G   0    16G   0% /dev/shm
tmpfs                    16G  17M   16G   1% /run
tmpfs                    16G   0    16G   0% /sys/fs/cgroup
/dev/mapper/VolGroup-lv_root 99G  1.5G  98G   2% /
/dev/vda1                 509M  282M  228M  56% /boot
tmpfs                    3.2G   0    3.2G   0% /run/user/0
/dev/vdb1                 275G   65M  261G   1% /var
[root@rhel-var ~]#
```

- b. Verification: Execute the command:
 - i. `df -h`
8. Move files from the backup directory to /var directory:
 - a. `mv /var2/* /var/`

```
[root@rhel-var ~]# mv /var2/* /var/
mv: cannot remove '/var2/tmp': Device or resource busy
[root@rhel-var ~]#
```

- b. Ignore the above error since temporary files cannot be removed
9. Create a `fstab` entry to make this change permanent:
 - a. Open the file `/etc/fstab`

```
[root@rhel-var ~]# vi /etc/fstab
[root@rhel-var ~]# .....
```

- b. Add the entry below and save the file

```
/dev/vdb1      /var      ext4      defaults      0 0
```

```
#
# /etc/fstab
# Created by anaconda on Thu Jun  2 12:31:44 2022
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/VolGroup-lv_root /                xfs     defaults      0 0
UUID=f3537874-5ccf-42eb-9a5a-81116970fdec /boot            xfs     defaults,nodev 0 0
/dev/vdb1      /var      ext4      defaults      0 0
~
```

- c. Reboot the VM to verify the changes

3. Installation

1. Log in to the two newly created machines using SSH
2. Install Docker and Docker Compose using the commands below:
 - a. `sudo su`
 - b. `subscription-manager repos --enable=rhel-7-server-rpms`
 - c. `subscription-manager repos --enable=rhel-7-server-extras-rpms`

- d. `subscription-manager repos --enable=rhel-7-server-optional-rpms`
 - e. `sudo yum install -y`
`https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm`
 - f. `sudo yum install -y yum-utils device-mapper-persistent-data lvm2`
 - g. `sudo yum-config-manager --add-repo`
`https://download.docker.com/linux/centos/docker-ce.repo`
 - h. `sudo yum install docker-ce`
 - i. `sudo systemctl enable --now docker.service`
 - j. `sudo curl -L`
`"https://github.com/docker/compose/releases/download/1.29.2/docker-`
`compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose`
 - k. `sudo chmod +x /usr/local/bin/docker-compose`
 - l. `mv /usr/local/bin/docker-compose /usr/bin/`
3. **Verification:** Execute the commands below to verify the Docker and Docker Compose versions:
- a. `docker version`
 - b. `docker-compose version`

-- END OF DOCUMENT --