Migrate Instances to the New Nimbus

Back to:

[Get Started with the New Nimbus](#)

⚠ We are accepting applications to use the new Nimbus, and all new and existing users should apply at the [application portal](#). These will be processed and access will be granted from **01 Apr 2020 .**

Please see the [Dates and Links](#) section of the [Get Started with the New Nimbus](#) page for further details.

With the new Nimbus hardware online, users are required to convert their existing instances to the new flavours to work on the new compute nodes. This conversion will only work for instances running newer operating systems. Older instances will need to be deleted and rebuilt from scratch. You can read [instructions](#) to help with recreating your instance.

Other elements within your Nimbus project (private networks, storage volumes, etc) do not need to be converted.

⚠ Any snapshots of an instance root volume taken prior to conversion will also no longer be usable.

Determine Operating System (OS) Version

1. Determine what distribution and version of Linux your instance is running. Look for a file called **/etc/os-release**:

```
ubuntu@ubuntu-18:~$ cat /etc/os-release
NAME="Ubuntu"
VERSION="18.04.3 LTS (Bionic Beaver)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 18.04.3 LTS"
VERSION_ID="18.04"
...
```

2. Look for the fields **NAME and VERSION_ID.**

    If they correspond to any of the following values, then your instance can be migrated.

NAME	VERSION_ID	Operating System	Support
Ubuntu	18.04	Ubuntu 18.04	Can be migrated with full CPU support
Ubuntu	16.04	Ubuntu 16.04	Can be migrated with limited CPU support
CentOS Linux	7	CentOS 7	Can be migrated with full CPU support

If the values do not match anything in the table above, you cannot migrate your instance directly to the new Nimbus hardware. Follow the [instructions](#) to recreate your instance and preserve any data volume you have attached.

Update Linux Kernel

Ubuntu

1. If you are running **Ubuntu 18.04** or **Ubuntu 16.04**, update all available packages using **apt-get** (including the kernel), then restart the instance:

```
sudo apt-get update
sudo apt-get -y dist-upgrade
sudo reboot
```

2. If prompted while performing the upgrade, select **keep the local version currently installed** (you may be asked multiple times).
3. After the instance has restarted, upgrade the kernel to the HWE (Hardware Enablement) kernel:

```
sudo apt-get -y install linux-generic-hwe-$(lsb_release -rs)
```

Again, if prompted while performing the install, select **keep the local version currently installed** (you may be asked multiple times).
4. Once the HWE kernel is installed, shut the instance down:

```
sudo shutdown -h now
```

CentOS

1. If you are running CentOS 7, update all packages, in particular the kernel.
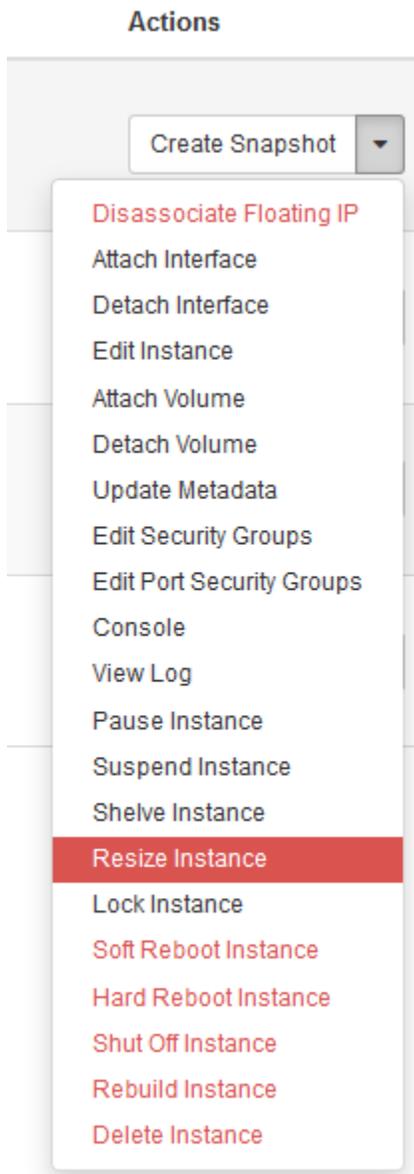2. Shut the instance down.

```
sudo yum makecache
sudo yum update -y
sudo shutdown -h now
```

Resize Instance

Once the instance is shut down, log in to the Nimbus dashboard, then

1. Go to **Compute** > **Instances.**
2. Confirm the instance is shut down.

3. From the drop-down menu on the right of the instance, select **Resize Instance**:

4. On the dialog window, resize the instance. The **Old Flavor** field will be set to the current flavour of your instance; you need to set the **New Flavor** field the new flavour that corresponds to your old flavor.

Old Flavor	New Flavor
m2.small	n3.1c4r
m2.medium	n3.2c8r
m2.large	n3.4c16r
m2.xlarge	n3.8c32r
m2.jumbo	n3.16c64r

5. After setting the new flavour, click **Resize**. Your instance goes through a resizing process and when finished the Status will be **Confirm or Revert Resize/Migrate**.
6. Click **Confirm Resize/Migrate** to the right of the instance, to confirm the resize:

Confirm CPU Visibility

Once completed, your instance will be ready to be started back up.

1. Use SSH to connect to your instance.
2. Confirm that your instance can see the new CPUs by running **lscpu** :

```
[centos@centos-7 ~]$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                1
On-line CPU(s) list:   0
Thread(s) per core:    1
Core(s) per socket:    1
Socket(s):             1
NUMA node(s):          1
Vendor ID:             AuthenticAMD
CPU family:            23
Model:                 1
Model name:            AMD EPYC Processor (with IBPB)
Stepping:              2
CPU MHz:               2345.592
BogoMIPS:              4691.18
Virtualization:        AMD-V
Hypervisor vendor:     KVM
Virtualization type:   full
L0 cache:              0K
L0 cache:              677K
NUMA node0 CPU(s):     0
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr
pge mca cmov pat pse36 clflush mmx fxsr sse sse2 syscall nx mmxext
fxsr_opt pdpe1gb rdtscp lm art rep_good nopl xtopology extd_apicid amd_dcm
eagerfpu pni pclmulqdq ssse3 fma cx16 sse4_1 sse4_2 x2apic movbe popcnt
tsc_deadline_timer aes xsave avx f16c rdrand hypervisor lahf_lm cmp_legacy
svm cr8_legacy abm sse4a misalignsse 3dnowprefetch osvw topoext
perfctr_core retpoline_amd ssbd ibpb vmmcall fsgsbase tsc_adjust bmi1 avx2
smep bmi2 rdseed adx smap clflushopt clwb sha_ni xsaveopt xsavec xgetbv1
arat
```

IMPORTANT: For **Ubuntu 16** instances, the above command does not work. As mentioned previously, this means the instance does not support the new Epyc features. It will still run on the new compute nodes, however.