EBS

24 December 2024 14:04

Throughput mean data transfer speed ----(copy past/ Downloads speed)

EBC

SSD HDD

In AWS, **EBS Volume Types** are categorized based on performance and cost, allowing users to select storage optimized for their workload requirements. Below is a detailed explanation of each volume type you listed:

1. General Purpose SSD (gp2):
- **Purpose**: Balances price and performance for a wide variety of workloads.
- **Performance**:
  - Provides baseline performance of 3 IOPS per GB.
  - Can burst up to 3,000 IOPS for volumes under 1 TB.
- **Use Cases**:
  - System boot volumes.
  - Low-latency interactive applications.
  - Development and testing environments.
- **Capacity**: 1 GiB to 16 TiB.
- **Throughput**: Up to 250 MiB/s.

2. General Purpose SSD (gp3):
- **Purpose**: Next-generation general-purpose SSD offering predictable performance and lower cost compared to gp2.
- **Performance**:
  - Delivers a baseline performance of 3,000 IOPS and 125 MiB/s.
  - Performance can be provisioned up to 16,000 IOPS and 1,000 MiB/s, independent of volume size.
- **Use Cases**:
  - Databases, boot volumes, and applications requiring consistent performance.
- **Capacity**: 1 GiB to 16 TiB.
- **Throughput**: Up to 1,000 MiB/s.
- **Key Benefit**: More cost-effective and flexible compared to gp2.

3. Provisioned IOPS SSD (io1):
- **Purpose**: High-performance SSD for mission-critical workloads that require low-latency and high IOPS.
- **Performance**:
  - Provision IOPS from 100 to 64,000 (max 50 IOPS per GiB for volumes up to 64 TiB).
  - Designed for latency-sensitive workloads.
- **Use Cases**:
  - Large-scale databases (e.g., Oracle, SQL Server, MySQL).
  - Applications requiring high transactional workloads.
- **Capacity**: 4 GiB to 16 TiB.
- **Key Benefit**: Customizable performance for demanding workloads.

4. Provisioned IOPS SSD (io2):
- **Purpose**: Enhanced version of io1 with higher durability and better performance.
- **Performance**:
  - Provision up to 64,000 IOPS (max 500 IOPS per GiB for volumes up to 64 TiB).
  - 99.999% durability (compared to io1's 99.9%).
- **Use Cases**:
  - Critical applications requiring very high durability and performance.
  - Enterprise-grade databases and ERP systems.
- **Capacity**: 4 GiB to 64 TiB.
- **Key Benefit**: Higher durability and efficiency for high-performance workloads.

5. Cold HDD (sc1):
- **Purpose**: Low-cost, infrequent access storage for data that is rarely accessed.
- **Performance**:
  - Baseline throughput of 12 MiB/s per TiB.
  - Max throughput of 250 MiB/s.
- **Use Cases**:

    - Cold data storage.
    - Log storage or backups that are rarely accessed.
- **Capacity**: 125 GiB to 16 TiB.
- **Key Benefit**: Cost-efficient storage for rarely accessed data.

6. Throughput Optimized HDD (st1):
- **Purpose**: Optimized for workloads requiring high throughput rather than IOPS.
- **Performance**:
    - Baseline throughput of 40 MiB/s per TiB.
    - Max throughput of 500 MiB/s.
- **Use Cases**:
    - Big data processing (e.g., Hadoop).
    - Data warehouses.
    - Streaming workloads requiring high throughput.
- **Capacity**: 125 GiB to 16 TiB.
- **Key Benefit**: Cost-effective for high-throughput workloads.

7. Magnetic (Standard):
- **Purpose**: Legacy storage option for workloads requiring cost-effective, low-performance storage.
- **Performance**:
    - Average IOPS of 100.
    - Suitable for sequential read/write operations.
- **Use Cases**:
    - Archive storage.
    - Non-critical workloads or boot volumes for older instance types.
- **Capacity**: Up to 1 TiB.
- **Key Benefit**: Simple and cost-effective, but mostly replaced by newer options.

Comparison Summary:

Volume Type	Max IOPS	Max Throughput	Durability	Cost	Use Case
General Purpose SSD (gp2)	16,000	250 MiB/s	Standard	Moderate	Balanced workloads
General Purpose SSD (gp3)	16,000	1,000 MiB/s	Standard	Low	Databases, boot volumes
Provisioned IOPS SSD (io1)	64,000	1,000 MiB/s	High	High	Critical databases
Provisioned IOPS SSD (io2)	64,000	1,000 MiB/s	Very High	High	Enterprise applications
Cold HDD (sc1)	~500	250 MiB/s	Standard	Very Low	Rarely accessed data
Throughput Optimized HDD (st1)	~500	500 MiB/s	Standard	Low	Streaming, big data
Magnetic (Standard)	~100	Low	Standard	Very Low	Legacy, basic storage

Notes:
- **IOPS (Input/Output Operations Per Second)**: Measures the performance of a volume in handling read/write operations.
- **Durability**: Higher durability reduces the risk of data loss.
- **Throughput**: Indicates the data transfer rate, often critical for workloads like streaming or big data.
- **Cost Optimization**: Match the volume type to your workload to avoid unnecessary costs.

Let me know if you need further clarification!

Step of how to Attach volume EBS

- Create a instance Lunch the instance
- Go to EBS Volume
- Create the Volume
    - Select the Volume types
        - General Purpose SSD (gp2)
        - General Purpose SSD (gp3)
        - Provisioned IOPS SSD (io1)
        - Provisioned IOPS SSD (io2) --------- (input output per sec.)
        - Cold HDD (Sc1)
        - Throughput Optimized HDD (st1)
        - Magnetic (standard)
    - Select Size (GiB)
    - Select IOPS
    - Select throughput ----------(data transfer speed MB per sec.)

- Select Availability Zone
- Click the cleck the Create Volume

- After create the Volume click the Actions
- and select the Attach Volume
  - Provide the default Volume ID
  - Default Availability Zone
  - Select the Running Instance show available in Availability Zone
  - Provide the device name
  - Click the Attach Volume
- Go to the terminal

Command

- lsblk
  - display information about block devices. Block devices include hard drives, SSDs, USB drives, and partitions
  - Syntax lsblk [option]
    - Option
      - lsblk -a -------- (-a: show all devices, including empty ones)
      - lsblk -l -------- (-l : display output in list format)
      - lsblk -f -------- (-f : show the file system type UUIDs)

Example of output

-
```
root@ip-172-31-6-109:/# lsblk
NAME     MAJ:MIN   RM  SIZE RO TYPE MOUNTPOINTS
loop0      7:0      0 26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1      7:1      0 73.9M  1 loop /snap/core22/1663
loop2      7:2      0 38.8M  1 loop /snap/snapd/21759
xvda     202:0      0    8G  0 disk
├─xvda1  202:1      0    7G  0 part /
├─xvda14 202:14     0    4M  0 part
├─xvda15 202:15     0  106M  0 part /boot/efi
└─xvda16 259:0      0  913M  0 part /boot
xvdbd    202:14080  0  160G  0 disk
```

Explanation of Output:

- **NAME**: The name of the block device (e.g., xvda, xvdf).
- **MAJ:MIN**: Major and minor device numbers.
- **RM**: Indicates whether the device is removable (1 for removable, 0 otherwise).
- **SIZE**: The size of the device or partition.
- **RO**: Indicates whether the device is read-only (1 for read-only, 0 otherwise).
- **TYPE**: The type of device (e.g., disk, part for partition, rom for CD-ROM).
- **MOUNTPOINT**: The directory where the device is mounted.

- mkfs.ext4 /dev/xvdbd
  - is used to format a block device (in this case, /dev/xvdbd) with the **ext4** file system on a Linux system, such as one running on an AWS EC2 instance

    Breakdown of the Command:
    - **mkfs.ext4**: Creates an ext4 file system on the specified device.
    - **/dev/xvdbd**: Specifies the block device to be formatted. This might represent an attached EBS volume or another storage device.

-
```
root@ip-172-31-6-109:/home/ubuntu# mkfs.ext4 /dev/xvdbd
mke2fs 1.47.0 (5-Feb-2023)
/dev/xvdbd contains a ext4 file system
        created on Thu Jan  2 07:17:42 2025
Proceed anyway? (y,N) y
Creating filesystem with 41943040 4k blocks and 10485760 inodes
Filesystem UUID: 8981afee-3ce6-4a1d-ad75-5d070f0e744e
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424, 20480000, 23887872

Allocating group tables: done
Writing inode tables: done
Creating journal (262144 blocks): done
Writing superblocks and filesystem accounting information: done
```

  - mkdir /test
  - mount /dev/xvdbd /test

```
root@ip-172-31-6-109:/# lsblk
NAME     MAJ:MIN   RM  SIZE RO TYPE MOUNTPOINTS
loop0      7:0      0 26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1      7:1      0 73.9M  1 loop /snap/core22/1663
loop2      7:2      0 38.8M  1 loop /snap/snapd/21759
xvda     202:0      0    8G  0 disk
├─xvda1  202:1      0    7G  0 part /
├─xvda14 202:14     0    4M  0 part
├─xvda15 202:15     0  106M  0 part /boot/efi
└─xvda16 259:0      0  913M  0 part /boot
xvdbd    202:14080  0  160G  0 disk /test
root@ip-172-31-6-109:/#
```

- mountpoint /test
  - The EBS volume are allot to the test directory

```
root@ip-172-31-6-109:/# mountpoint /test/
/test/ is a mountpoint
root@ip-172-31-6-109:/#
```

- umount /test
  - Remove the mount

```
root@ip-172-31-6-109:/# umount /test/
root@ip-172-31-6-109:/# lsblk
NAME     MAJ:MIN   RM  SIZE RO TYPE MOUNTPOINTS
loop0      7:0      0 26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1      7:1      0 73.9M  1 loop /snap/core22/1663
loop2      7:2      0 38.8M  1 loop /snap/snapd/21759
xvda     202:0      0    8G  0 disk
├─xvda1  202:1      0    7G  0 part /
├─xvda14 202:14     0    4M  0 part
├─xvda15 202:15     0  106M  0 part /boot/efi
└─xvda16 259:0      0  913M  0 part /boot
xvdbd    202:14080  0  160G  0 disk
root@ip-172-31-6-109:/#
```

- file -s /dev/xvdbd

- how to increase the size of EBS volume
  - Select the volume click the actions
  - Click Modify volume
    - Change the size increase in GB
    - And click modify
    - After increase the volume
    - Go to the terminal
      - Write the command
        - resize2fs /dev/xvdbd ---( )
        - df -h --- (Use df -h to confirm the file system has been resized)
        - fdisk /dev/xvdbd
          - n ----(new partition)
          - p -----(primary partition)
          - w -----(save)
          - q -----(quit)
          - mkfs.ext4 dev/xvdbd1 -----(make a file system)
          - mkdir ebs
          - mount /dev/xvdbd1 ebs
          - df -h
          - mountpoint
          - umount ---------(remove the mount)