EBS stands for **Amazon Elastic Block Store**. It is a storage service provided by **Amazon Web Services (AWS)** that offers **block-level storage volumes** for use with **EC2 (Elastic Compute Cloud)** instances.

Key Features of EBS:

- **Durable** and **persistent storage**: Data remains even after the EC2 instance is stopped or terminated (unless it's a temporary volume).
- **Block-level storage**: Works like a hard drive attached to your EC2 instance.
- **Customizable**: Choose storage type (e.g., SSD, HDD) based on performance and cost.
- **Snapshot support**: Take backups of volumes and store them in S3.
- **Encrypted storage**: Provides data-at-rest encryption.
- **Scalable**: You can resize volumes and increase performance as needed.

Types of EBS Volumes:

1. **gp3 (General Purpose SSD)** – Balanced price and performance.
2. **io2/io1 (Provisioned IOPS SSD)** – High-performance SSD for critical applications.
3. **st1 (Throughput Optimized HDD)** – For large, sequential workloads (like big data).
4. **sc1 (Cold HDD)** – Lowest cost, for infrequent access.

Common Use Cases:

- Hosting databases (e.g., MySQL, MongoDB)
- Storing application data
- Boot volumes for EC2 instances
- Storing logs or backups

Would you like an example of how to create and attach an EBS volume to an EC2 instance?

🔹 1. What is IOPS?

IOPS = Input/Output Operations Per Second

➤ Meaning:

It measures **how many read/write operations** your storage can handle **per second**.

- It's about **how many times** your system can **access small chunks** of data quickly.
- Important for **random, small transactions** (e.g., databases, web apps).

🧠 Think of it like:

Imagine a librarian who gives you **small pages** from books.
IOPS = how many pages they can hand you per second.

📌 Example:

- If an EBS volume has **3,000 IOPS**, it can perform **3,000 read/write operations** per second.
- Useful for apps that read/write small data **very frequently**, like **MySQL**, **PostgreSQL**, etc.

🔹 2. What is Throughput?

Throughput = How much total data is transferred per second, measured in **MB/s** (megabytes per second).

➤ Meaning:

It's about **how fast big files** can be transferred — i.e., the **volume of data** processed in a second.

- Good for **large, sequential data** (e.g., streaming video, big data processing).

🧠 Think of it like:

A water pipe.
Throughput = how wide the pipe is, i.e., how much water (data) flows through **each second**.

📌 Example:

- If your EBS has **500 MB/s throughput**, it can transfer **500 megabytes of data** per second.
- Useful for **data lakes**, **log processing**, **media streaming**.

🔹 3. What is Size?

Size = How much **total storage space** you allocate to your volume. Measured in **GiB** (Gibibytes).

➤ Meaning:

It's the **capacity** of your disk, like a pen drive or hard disk.

- The larger the size, the more files or data you can store.

🧠 Think of it like:

A shelf.
Size = how many books it can store.

📌 Example:

- A 100 GiB EBS volume can store up to **100 gibibytes** of data.
- 1 GiB = 1,073,741,824 bytes (slightly more than 1 GB)

🧩 Putting It All Together

Let's take an EBS volume with:

- **Size** = 100 GiB
- **IOPS** = 3,000
- **Throughput** = 250 MB/s

This means:

- You can store 100 GiB of files
- You can read/write **up to 3,000 small chunks** of data per second
- You can transfer **up to 250 megabytes** of large files per second

---

💡 Summary

Term	Unit	What It Measures	Best For
IOPS	operations/sec	Speed of small, random reads/writes	Databases, OLTP apps
Throughput	MB/sec	Total data transfer rate (big files)	Big data, media, analytics
Size	GiB	Total storage capacity	File storage, logs, backups

🔸 1. Private IP

✅ What it is:

- An IP address **used within a private network**.
- Assigned automatically to every EC2 instance.
- **Not accessible from the internet**.

🧠 Example:

Think of it like your **house address inside a gated community**. Only neighbors (inside network) can reach you.

📌 Use Cases:

- Communication between EC2 instances in the **same VPC or subnet**.
- **Database servers**, backend systems.

🧪 Example:

`10.0.0.25` or `172.31.0.5` (typical private IP ranges)

🔸 2. Public IP

✅ What it is:

- An IP address that is **accessible from the internet**.
- Assigned automatically when you launch an EC2 instance with **auto-assign public IP = enabled**.
- **Temporary**: changes when the instance stops/starts.

🧠 Example:

Like your **main road address** that people on the internet can use to reach your house.

📌 Use Cases:

- Connecting to EC2 via **SSH from your laptop**
- Hosting websites, APIs that are accessed publicly

🔸 3. Elastic IP

✅ What it is:

- A **static public IP address** provided by AWS.
- **Does not change** when you stop/start the EC2 instance.
- You can **detach and reattach** it to any EC2 instance in your account.

🧠 Example:

Like buying a **permanent phone number** for your business. Even if you change your phone (instance), you keep the number (IP).

📌 Use Cases:

- Applications that need a **stable IP** for DNS, APIs, firewalls, etc.
- Failover setups (move IP to another instance if one fails)

🧩 Summary Table

IP Type	Internet Accessible?	Changes on Stop/Start?	Use Case
Private IP	❌ No	❌ No	Internal communication
Public IP	✅ Yes	✅ Yes	Temporary internet access
Elastic IP	✅ Yes	❌ No	Static IP for public-facing services

Would you like a visual diagram comparing all three in an AWS VPC setup?