

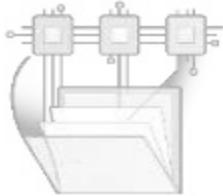
AWS Elastic Block Store

Persistent block level storage volumes for
use with Amazon EC2 instances in the
AWS Cloud

Russell Kole

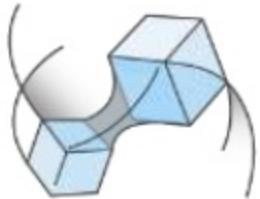
Solution Architect - AWS

AWS Storage is a Platform



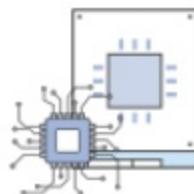
Amazon EFS

File



Amazon EBS

Block



Amazon EC2
Instance Store



Amazon S3



Amazon Glacier

Object

Data Transfer



Internet/VPN



Amazon S3
Transfer
Acceleration



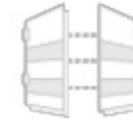
Amazon
CloudFront



AWS Direct
Connect



Amazon
Kinesis
Firehose



AWS Storage
Gateway



AWS
Snowball



ISV
Connectors

AWS Storage Choices



Amazon **S3**

Durable object storage
for all types of data



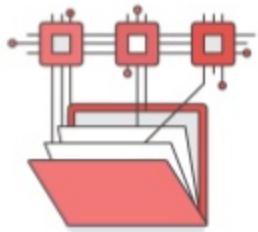
Amazon **EBS**

Block storage for use
with Amazon EC2



Amazon **Glacier**

Archival storage
for infrequently
accessed data



Amazon **EFS**

Simple, scalable, and
reliable file storage

Amazon EBS

Highly available block storage for all types of data



Internet-scale storage
Grow without limits



Built-in redundancy
Designed for
99.999% availability



**Low price per GB
per month**
No commitment
No up-front cost



**Benefit from AWS's
massive security
investments**

Amazon EBS use cases

- Persistent block storage for [Amazon EC2](#)
- **Boot Volumes**
- **Enterprise Applications** – Oracle, SAP, Microsoft Exchange, and Microsoft Sharepoint
- **Relational Databases** – SQL, Oracle, MySQL, and PostgreSQL
- **NoSQL** – Cassandra, MongoDB, and Couchbase.
- **Big Data** – Hadoop/EMR, Kafka/stream processing, Splunk/logs, ETL, Data Warehousing, Media Streaming
- **File** system for an instance - NTFS, ExtFS etc.
- **Dev and Test Environments**

AWS EBS Features

Durable And Available

Designed for five 9's reliability

Redundant storage across multiple devices within an AZ

Designed for an annual failure rate (AFR) of between 0.1% - 0.2%,

Performance

Low-latency SSD

Consistent I/O Performance

Stripe multiple volumes for higher I/O performance

Backup

Point-in-time Snapshots

Copy snapshots across AZ and Regions

Scalable

Unlimited capacity when you need it

Easily scale up and down

Secure

Identity and Access Policies
Encryption

2015 - 2016 Recap

Larger & Faster EBS SSD Volumes

- 16 TB, up to 20,000 IOPS

Performance Improvements

- Eliminating pre-warming of new EBS volumes
- Eliminating performance penalty while snapshotting

Cross-Region Encrypted Snapshot Copy

New Throughput-Intensive Magnetic Volumes

- ST1 (max 500 MB/sec) - ~\$0.045
- SC1 (max 250 MB/sec) - ~\$0.025

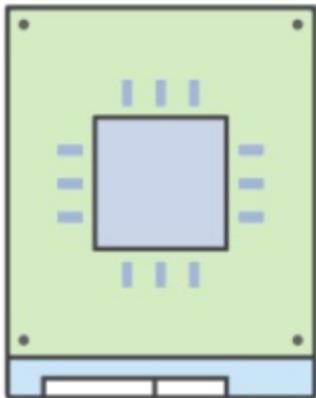
GP2 Burst Bucket Metrics

Volume Types

	Solid State Drives (SSD)		Hard Disk Drives (HDD)	
Volume Type	General Purpose SSD (gp2)*	Provisioned IOPS SSD (io1)	Throughput Optimized HDD (st1)	Cold HDD (sc1)
Use Cases	Boot volumes, low-latency interactive apps, dev & test	I/O-intensive NoSQL and relational databases	Big data, data warehouses, log processing	Colder data requiring fewer scans per day
Volume Size	1 GB - 16 TB	4 GB - 16 TB	500 GB - 16 TB	500 GB - 16 TB
Max IOPS*/Volume	10,000	20,000	500	250
Max Throughput/Volume	160 MB/s	320 MB/s	500 MB/s	250 MB/s
Price	\$0.10/GB-month	\$0.125/GB-month \$0.065/provisioned IOPS	\$0.045/GB-month	\$0.025/GB-month
Dominant Performance Attribute	IOPS	IOPS	MB/s	MB/s

*io1/gp2 based on 16K I/O size, st1/sc1 based on 1 MB I/O size

EBS Volume Types: IOPS



gp2

General Purpose SSD

Baseline: 3 IOPS per GB up to 10,000

Burst: 3,000 IOPS (for volumes up to 1 TB)

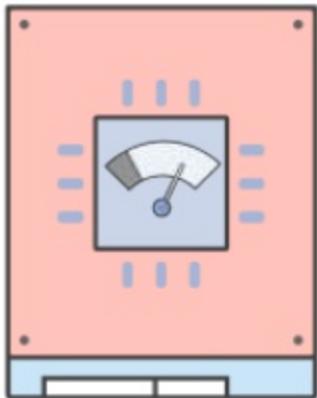
Throughput: 160 MB/s

Latency: Single-digit ms

Capacity: 1 GB to 16 TB

Great for boot volumes, low latency applications and bursty databases

EBS Volume Types: IOPS



io1

Provisioned IOPS SSD

Baseline: 100 to 20,000 IOPS

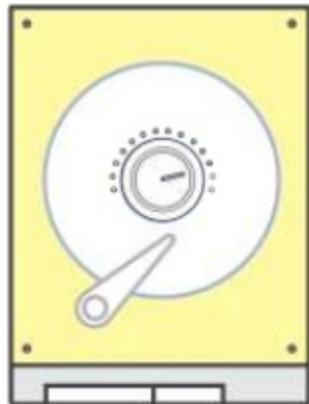
Throughput: 320 MB/s

Latency: Single-digit ms

Capacity: 4 GB to 16 TB

Ideal for critical applications and databases with sustained IOPS

EBS Volume Types: Throughput



st1

Throughput
Optimized HDD

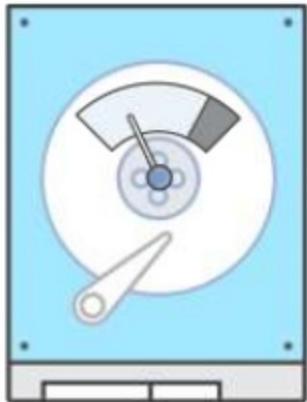
Baseline: 40 MB/s per TB up to 500 MB/s

Burst: 250 MB/s per TB up to 500 MB/s

Capacity: 500 GB to 16 TB

Ideal for large block, high throughput sequential workloads

EBS Volume Types: Throughput



sc1

Cold HDD

Baseline: 12 MB/s per TB up to 192 MB/s

Burst: 80 MB/s per TB up to 250 MB/s

Capacity: 500 GB to 16 TB

Ideal for sequential throughput workloads such as logging and backup

Customer Example SSD (GP2): Crowdstrike

Cassandra on EBS vs. Ephemeral storage

“Amazon EBS offered the performance we needed, at a third of the cost of the SSD-backed instance storage.”

Goal: 1 million writes per second on 60 nodes with EBS

Customer Example: Crowdstrike, cont.

Used to believe they could never run Cassandra on EBS:

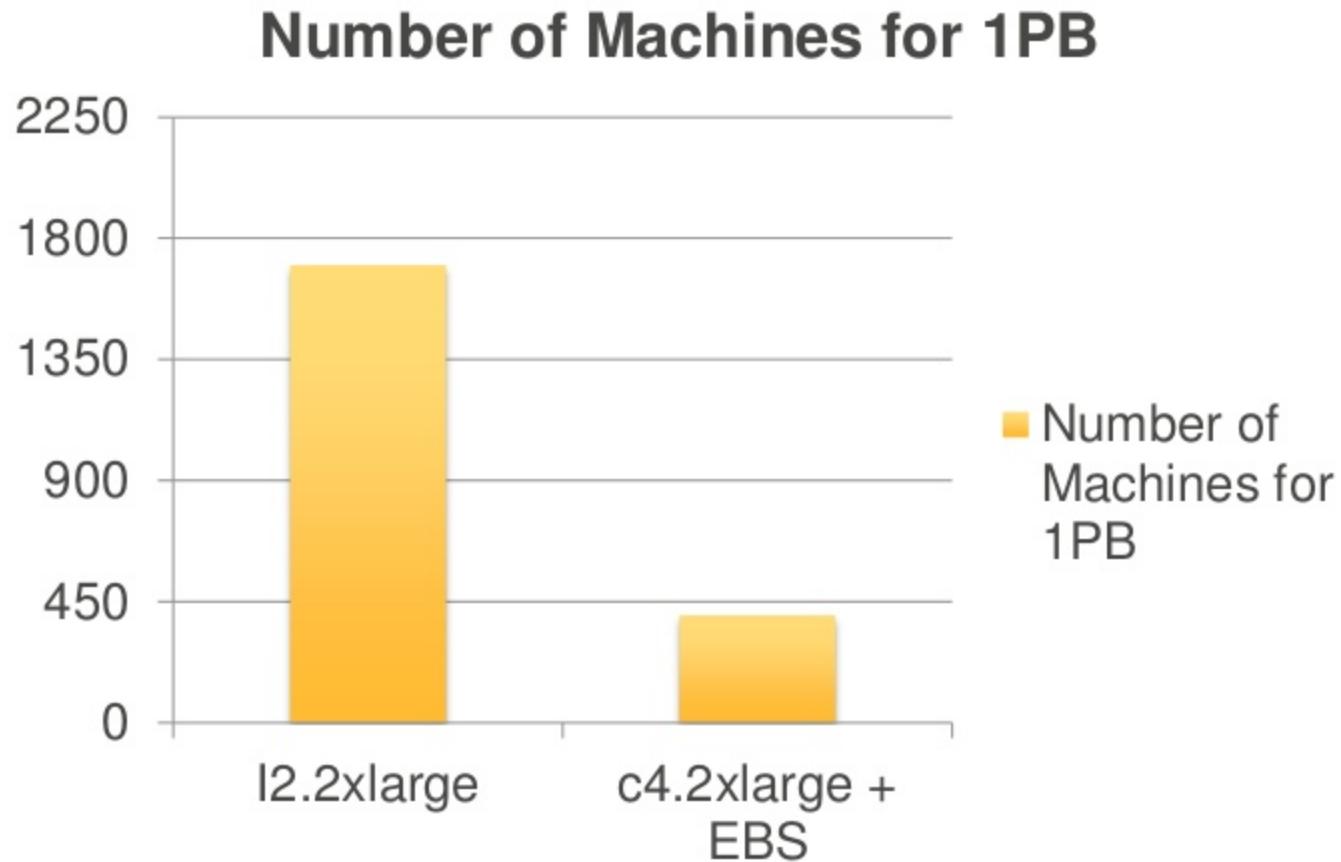
- Noisy Neighbor (jitter)
- Single point of failure in a region
- Too expensive
- Bad Volumes (spin up ten, run tests, and pick the best one)

Customer Example: Crowdstrike, cont.

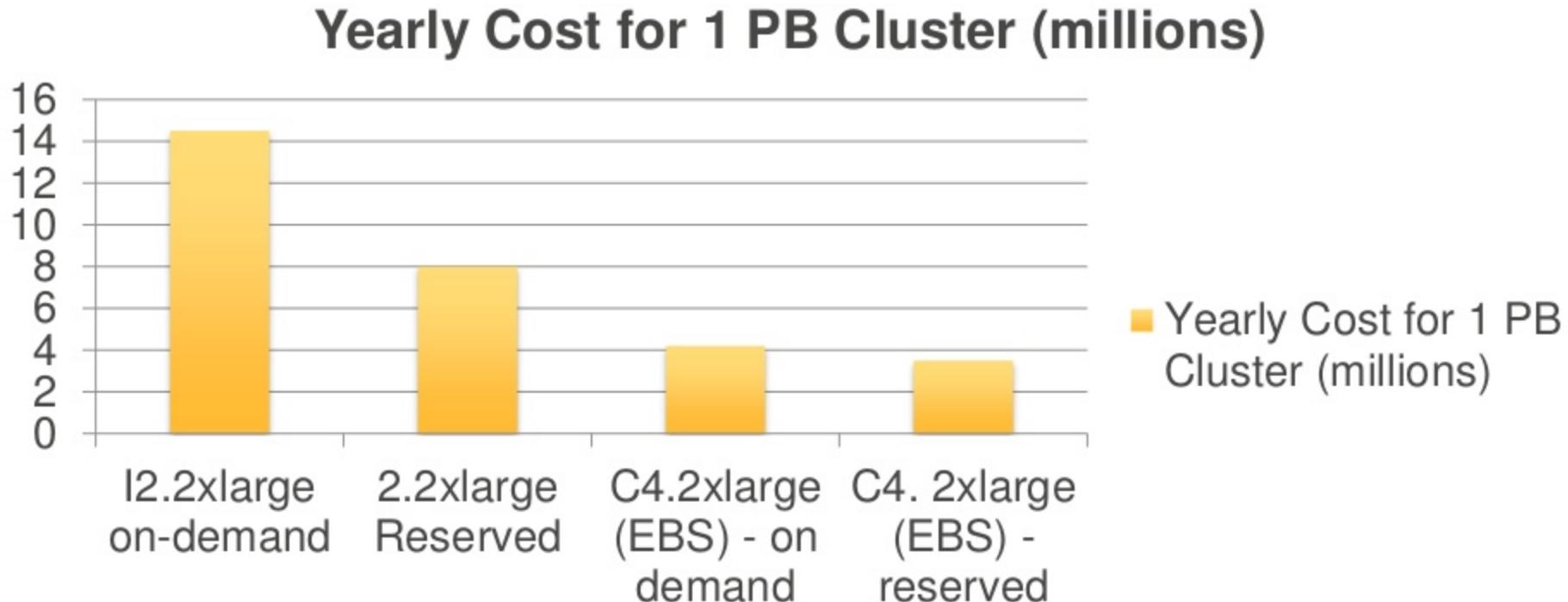
Disadvantages to running Cassandra on ephemeral storage

- Fewer EC2 instances are offering local instance storage
- There is no data persistence if you stop/start the EC2 instance to resize
- I2's are expensive, especially when you need three of them per node for the replication
- You can't snapshot the data using EBS snapshot
- No EBS volume monitoring

Customer Example: Crowdstrike



Customer Example: Crowdstrike



Customer Example: Crowdstrike

Crowdstrike Today:

- In the past 12 months, zero Amazon EBS-related failures
- Thousands of GP2 data volumes (~2PB of data)
- Transitioning all systems to Amazon EBS root drives
- Moved all data stores to EBS

Benefits of EBS:

- Use EBS volume monitoring
- Schedule snapshots for consistent backups
- Stop/start and resize
- Half the cost (using reserved pricing comparison)

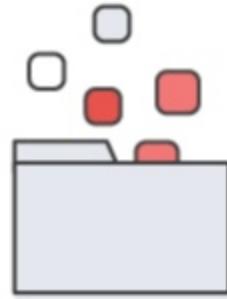
Introducing



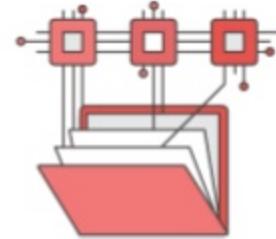
Amazon Elastic File System (EFS)

Agenda

1. Overview of Amazon EFS
2. Examples of Amazon EFS use cases
3. Review of Amazon EFS technical concepts
4. Walk through experience of creating a file system
5. Discuss file system security mechanisms
6. Review the Amazon EFS performance model
7. Explore the Amazon EFS regional availability and durability model

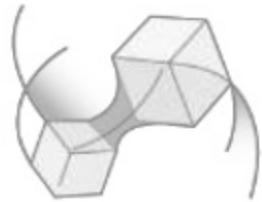


AWS Storage is a Platform

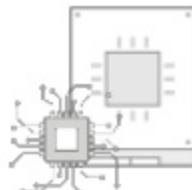


Amazon EFS

File



Amazon EBS



Amazon EC2
Instance Store

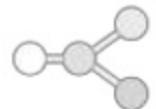


Amazon S3



Amazon Glacier

Data Transfer



Internet/VPN



AWS Direct
Connect



Amazon
CloudFront



S3 Transfer
Acceleration



ISV
Connectors



Storage
Gateway



AWS
Snowball



Amazon
Kinesis
Firehose

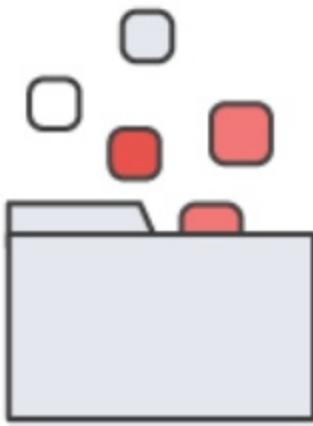
What is Amazon EFS?

- A fully managed file system for Amazon EC2 instances
- A file system interface with file system access semantics that works with standard operating system APIs
- Sharable across thousands of instances
- Designed to grow elastically to petabyte scale
- Built for performance across a wide variety of workloads
- Highly available and durable
- Strong consistency

We focused on changing the game



① Amazon EFS is Simple



Fully managed

- No hardware, network, file layer
- Create a scalable file system in seconds!

Seamless integration with existing tools and apps

- NFS v4.1—widespread, open
- Standard file system access semantics
- Works with standard OS file system APIs

Simple pricing = simple forecasting

② Amazon EFS is Elastic



File systems grow and shrink automatically as you add and remove files

No need to provision storage capacity or performance

You pay only for the storage space you use, with no minimum fee

3

Amazon EFS is Scalable



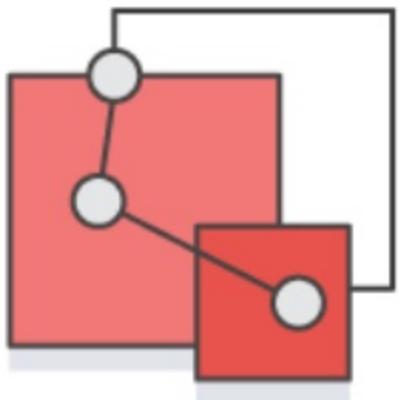
File systems can grow to petabyte scale

Throughput and IOPS scale automatically as file systems grow

Consistent low latencies regardless of file system size

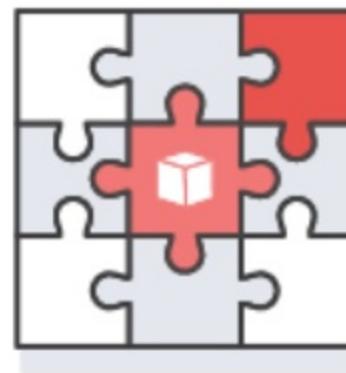
Support for thousands of concurrent NFS connections

Highly Durable and Highly Available



- Designed to sustain AZ offline conditions
- Resources aggregated across multiple AZ's
- Superior to traditional NAS availability models
- Appropriate for Production / Tier 0 applications

Use Cases



Example use cases

Big Data Analytics

Media Workflow Processing

Web Serving

Content Management

Home Directories

Use case: Enterprise Applications

“ We are growing by leaps and bounds, and our core offering is all about better support delivery. During the course of developing our next-generation internal support system, we never wanted to worry about scale again, yet we had existing architectural commitments that meant a distributed file solution was required. We chose Amazon EFS because it was the only option available that scaled both capacity and performance – without the up-front payments or the management overhead of traditional models. Now we can just focus on supporting our customers to help them support their customers. **”**



- Sam Caldwell, Sr. Systems Engineer

Use case: Enterprise Applications

“ Arcesium is a financial services SaaS platform that requires resilient, secure, and scalable file storage. Amazon EFS offers us a powerful way to operate and scale file storage for our Amazon EC2 instances and is just the kind of solution we need to build out our platform quickly and without compromising quality. **”**

- Guarav Suri, CEO



Arcesium

Use case: Web Serving/Content Management

“ Seeking Alpha subscribers rely on us for accurate trading information. Amazon EFS helps us manage the thousands of web pages on our site. It also helps us manage the traffic spikes generated by financial market events, and ensure that every bit of content requested during these spikes is quickly and reliably delivered to our subscribers. **”**

- Asi Segal, CTO

Seeking Alpha^α

Use case: Web Serving/Content Management

“

Our customers are the Internet. They build over 80% of the world's websites using our PHP technology.

Our testing has helped us conclude that the only logical way to share files in the cloud is on Amazon EFS. There's no easier way to get out of the file server management business and focus on web development.

- Boaz Ziniman, Sr. Dir Cloud Strategy

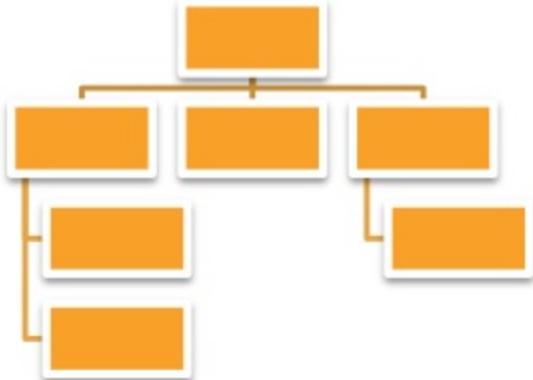


The PHP Company

”

Diving in

What is a file system?



The primary resource in Amazon EFS

Where you store files and directories

Can create multiple file systems per account

How to access a file system from an instance

You “mount” a file system on an Amazon EC2 instance (standard command) — the file system appears like a local set of directories and files

An NFSv4.1 client is standard on Linux distributions

```
mount -t nfs4 -o nfsvers=4.1  
    [file system DNS name] :/  
    /[user's target directory]
```

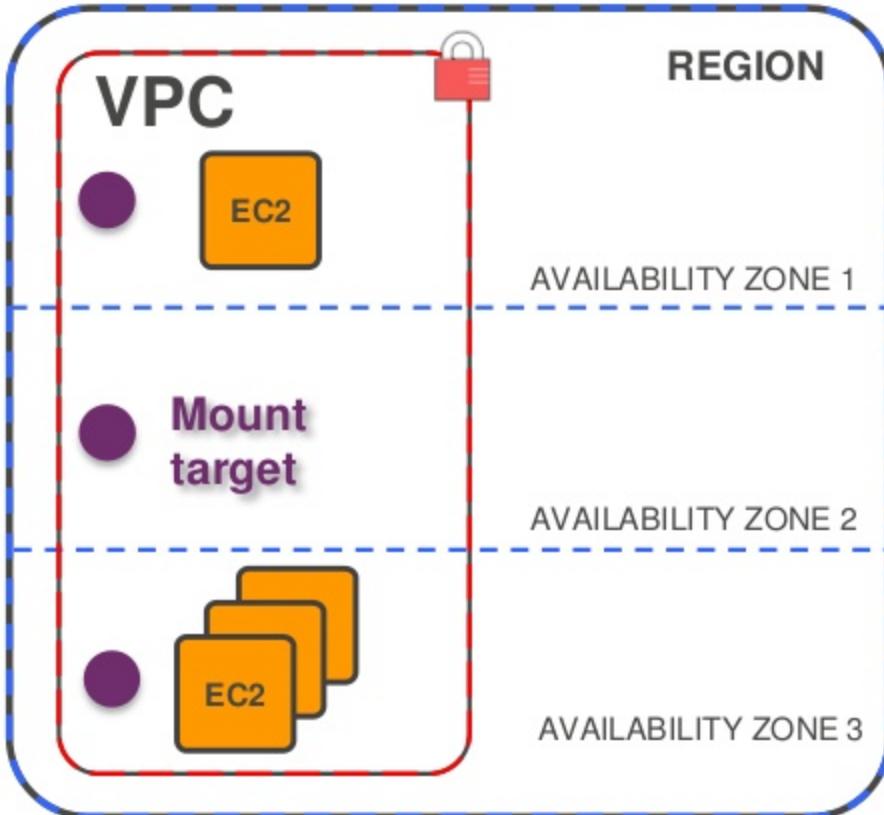


What is a mount target?

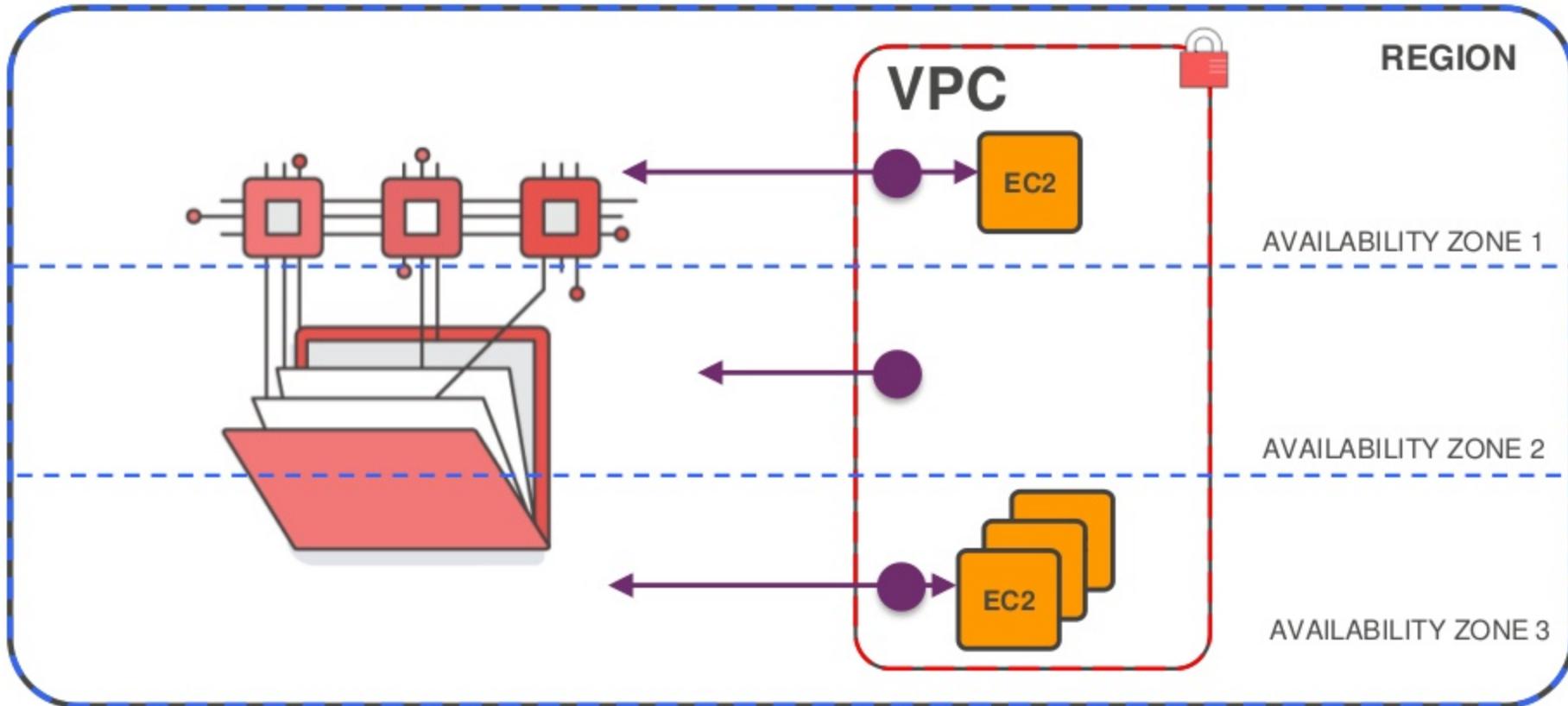
To access your file system from instances in an Amazon VPC, you create *mount targets* in the VPC

A mount target is an NFS endpoint in your VPC

A mount target has an IP address and a DNS name you use in your mount command



How does it all fit together?

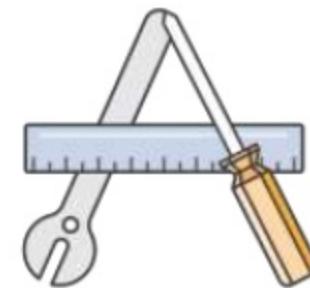


There are three ways to set up and manage a file system

AWS Management Console

AWS Command Line Interface (CLI)

AWS Software Development Kit (SDK)



The AWS Management Console, CLI, and SDK each allow you to perform a variety of management tasks

Create a file system

Create and manage mount targets

Tag a file system

Delete a file system

View details on file systems in your AWS account

Setting up and mounting a file system takes under a minute

1. Create a file system
2. Create a mount target in each Availability Zone from which you want to access the file system
3. Enable the NFS client on your instances
4. Run the mount command

Mounting your file system

1. Open an SSH client and connect to your EC2 instance. (find out how to [connect](#))
2. Create a new directory on your EC2 instance, such as "efs".

- `sudo mkdir efs`

3. Mount your file system using the DNS name. The following command looks up your EC2 instance's Availability Zone (AZ) using the EC2 instance metadata URI 169.254.169.254, then mounts the file system using the DNS name for that AZ. ([what is EC2 instance metadata?](#))

- `sudo mount -t nfs4 -o nfsvers=4.1 $(curl -s http://169.254.169.254/latest/meta-data/placement/availability-zone).fs-20877e69.efs.us-east-1.amazonaws.com:/ efs`

If you are unable to connect, please see our [troubleshooting documentation](#).

DNS names

X

You can mount your file system from EC2 instances in your VPC using the DNS name(s) below. A file system has a unique DNS name for each Availability Zone in which you have created a mount target.

Availability Zone	DNS name
us-east-1e	us-east-1e.fs-20877e69.efs.us-east-1.amazonaws.com
us-east-1d	us-east-1d.fs-20877e69.efs.us-east-1.amazonaws.com
us-east-1b	us-east-1b.fs-20877e69.efs.us-east-1.amazonaws.com
us-east-1c	us-east-1c.fs-20877e69.efs.us-east-1.amazonaws.com

Close

Petabyte-scale file system available for use

```
[ec2-user@ip-172-31-43-133 ec2-user]$ df -H
Filesystem           Size  Used Avail Use% Mounted on
/dev/xvda1            8.4G  1.2G  7.1G  15% /
devtmpfs              513M   58k  513M   1% /dev
tmpfs                 523M     0  523M   0% /dev/shm
us-west-2b.fs-1a4aa7b3.efs.us-west-2.amazonaws.com:/  9.3E    0  9.3E   0% /home/ec2-user/efs
[ec2-user@ip-172-31-43-133 ec2-user]$
```

Don't worry—We only bill for the space you use ☺

Security



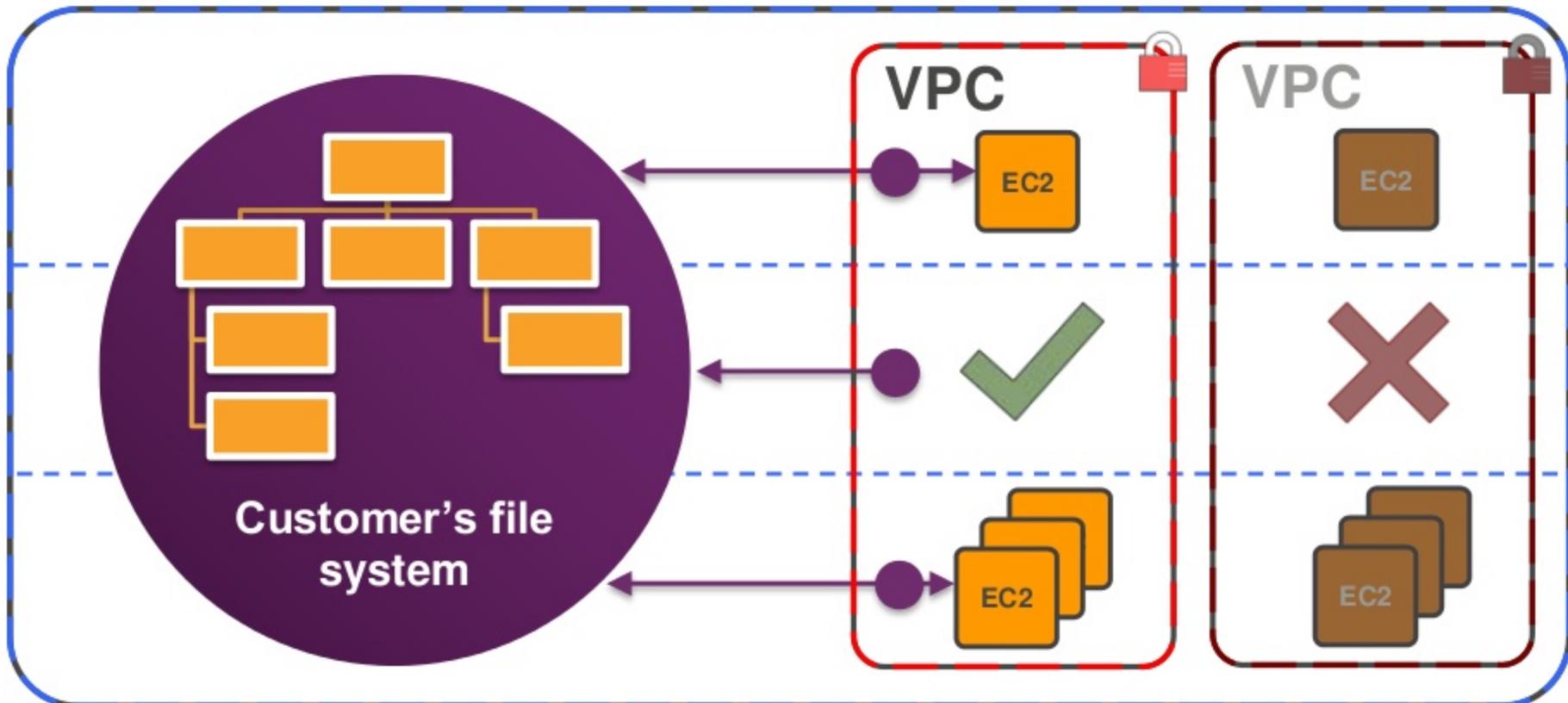
Security controls

Control network traffic to and from file systems by using VPC security groups and network ACLs

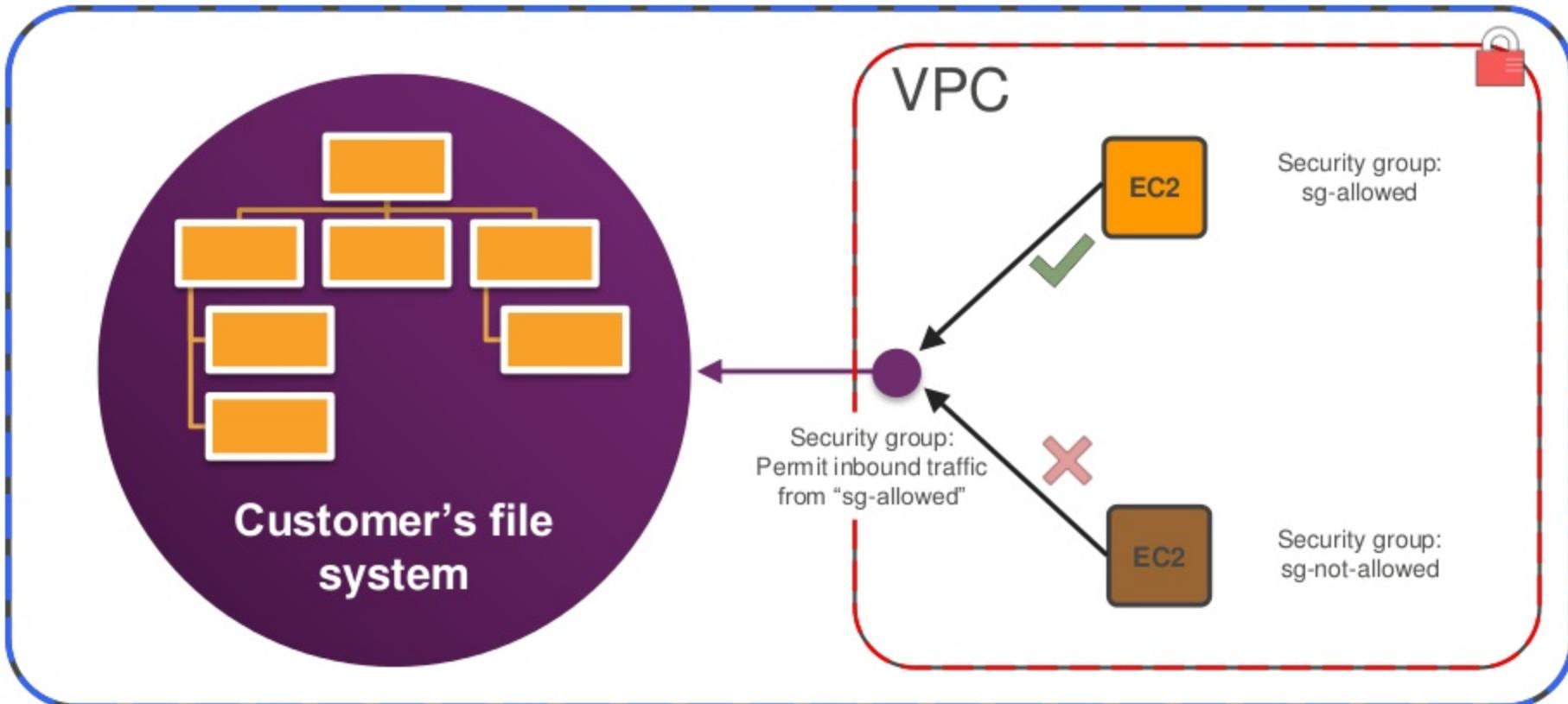
Manage file and directory access by using standard Linux directory and file-level permissions

Grant administrative access (via API) to file systems by using AWS Identity and Access Management (IAM)

Only EC2 instances in the VPC you specify can access your Amazon EFS file system



Security groups control which instances in your VPC can connect to your mount targets



Amazon EFS supports the POSIX permissions model

Set file/directory permissions to specify read-write-execute permissions for users and groups

```
drwxr-xr-x  4 root      root      4096 Feb  5 22:37 .
dr-xr-xr-x 25 root      root      4096 Feb  5 22:20 ..
drwxr-xr-x  2 mike      mike     4096 Feb  4 01:18 mike
```

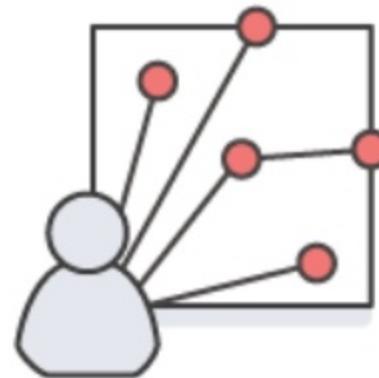
Integration with AWS IAM provides administrative security



Use IAM policies to control who can use the administrative APIs to create, manage, and delete file systems

Amazon EFS supports *action-level* and *resource-level* permissions

Performance



EFS provides a throughput bursting model that scales as a file system grows

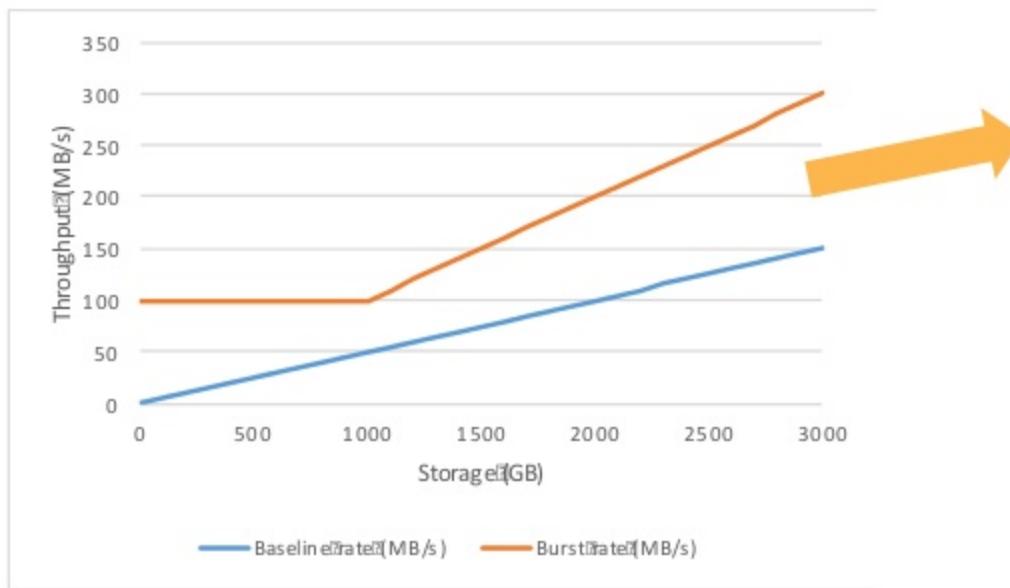
As a file system gets larger, it needs access to more throughput

Many file workloads are spiky, with peak throughput well above average levels



Amazon EFS scalable bursting model is designed to make performance available when you need it

Throughput bursting model based on earning and spending “bursting credits”

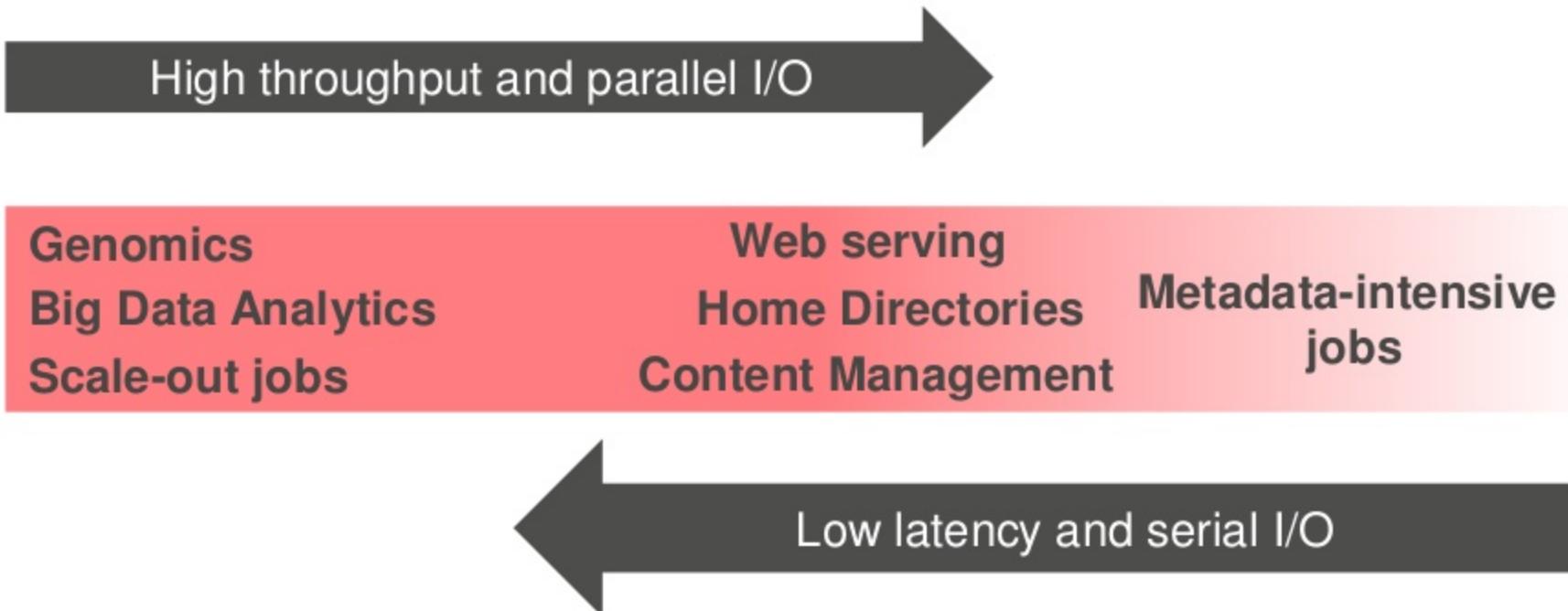


- File systems earn credits at a “baseline rate” of 0.05 MB/s per GB stored and use credits by performing file system operations; file systems can drive throughput at “baseline rate” indefinitely
- File systems with a positive bursting credit balance can “burst” to higher levels for periods of time: 100 MB/s for file systems 1TB or smaller, 100 MB/s per TB for file systems larger than 1TB
- New file systems start with a full credit balance

Bursting model examples

File system size	Read/write throughput
A <u>1 TB</u> EFS file system can...	<ul style="list-style-type: none">• Drive up to 50 MB/s continuously <i>or</i>• Burst to 100 MB/s for up to 12 hours each day*
A <u>10 TB</u> EFS file system can...	<ul style="list-style-type: none">• Drive up to 500 MB/s continuously <i>or</i>• Burst to 1 GB/s for up to 12 hours each day*
A <u>100 GB</u> EFS file system can...	<ul style="list-style-type: none">• Drive up to 5 MB/s continuously <i>or</i>• Burst to 100 MB/s for up to 72 minutes each day*

Amazon EFS is designed for wide spectrum of use cases



Two performance modes designed to support this broad spectrum of use cases

Default: Recommended for most use cases

General Purpose mode

Optimized for latency-sensitive applications and general-purpose file-based workloads – this mode is the best option for the majority of use cases

Max I/O mode

Can scale to higher levels of aggregate throughput with a tradeoff of slightly higher latencies for file operations

Use CloudWatch to determine whether your application may benefit from “Max I/O” mode; if not, you’ll get the best performance in “General Purpose” mode

CloudWatch metrics provide visibility into file system performance

View bursting credit balance and determine throughput permitted by the bursting model

Monitor I/O metrics to understand the throughput your file system is driving

Determine whether your workload may benefit from “Max I/O” mode



Regional availability and durability

In what regions can I use Amazon EFS?



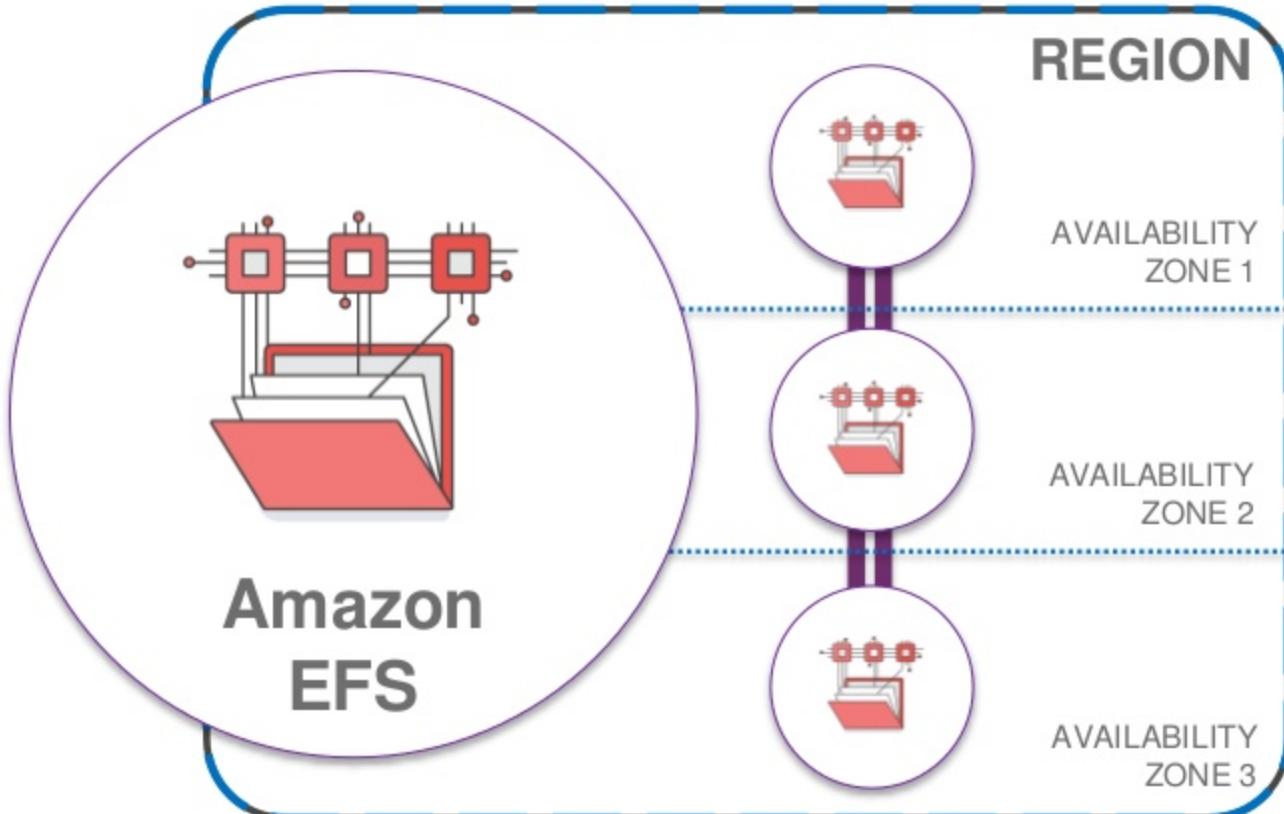
US-West-2 (Oregon)

US-East-1 (Northern Virginia)

EU-West-1 (Ireland)

Data is stored in multiple Availability Zones for high availability and durability

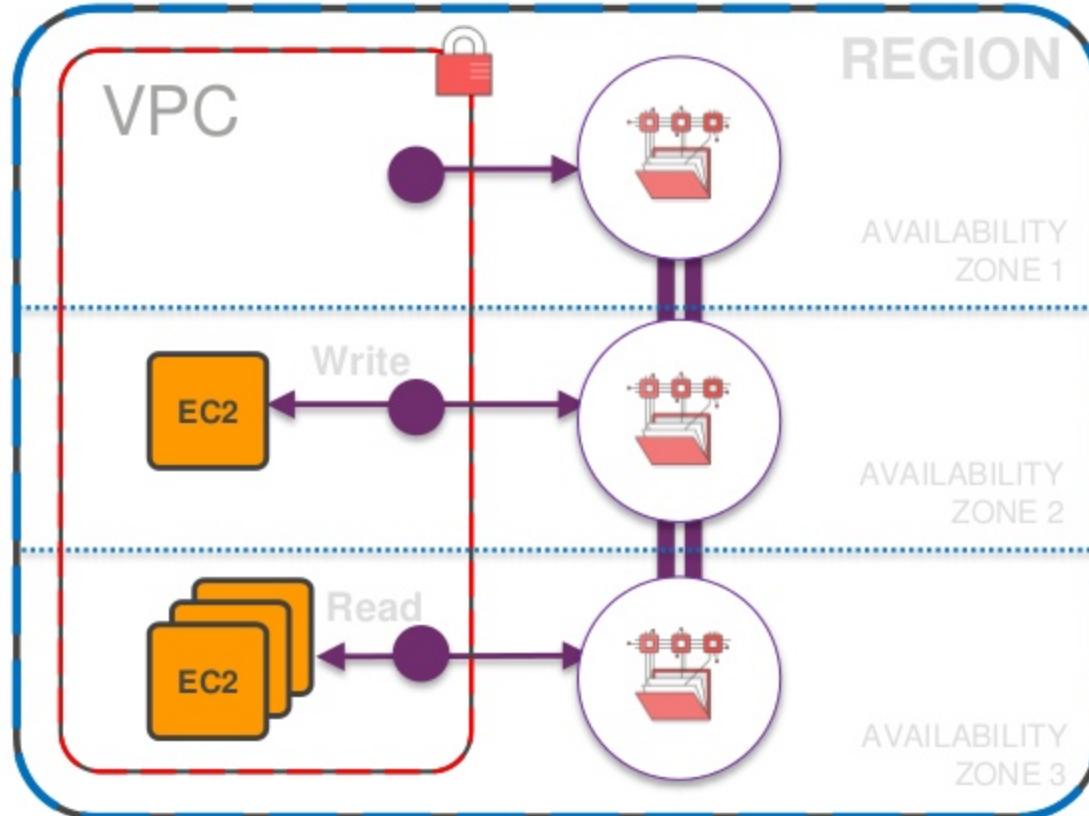
Every file system object (directory, file, and link) is redundantly stored across multiple Availability Zones in a region



Data can be accessed from any Availability Zone in the region while maintaining strong consistency

Your EC2 instances can connect to your EFS file system from any Availability Zone in a region

Reads and writes are strongly consistent in all Availability Zones



Wrapping up

Simple and predictable pricing

With EFS, you pay only for the storage space you use

- No minimum commitments or up-front fees
- No need to provision storage in advance
- No other fees, charges, or billing dimensions

EFS price: **\$0.30/GB-month** (US-East-1)

Customers eligible for the AWS Free Tier (12 months following AWS sign-up date) get 5GB free storage per month

Frequently asked questions

Q. Does Amazon EFS offer encryption of data at rest?

A. Not currently, but we will offer this feature soon.

Q. What AMIs work with Amazon EFS?

A. Amazon EFS is accessible from all Linux-based AMIs.

Q. How do I back up a file system?

A. Amazon EFS is designed to be highly durable. If you want to be able to revert to earlier versions to undo changes, you can use standard backup software.

Q. Can I mount Amazon EFS using NFSv4.0, or only NFSv4.1?

A. Amazon EFS supports either protocol, but we recommend mounting via NFSv4.1 for best performance.

Thank you

