Developer Operations

AWS Elastic Compute Cloud (EC2)

AWS Simple Storage Service (S3)

Amazon Elastic Compute Cloud (EC2)

Amazon
EC2

- **Resizable** compute capacity
- Complete control of computing resources
- Easy to obtain and boot new server instances

EC2 Features

- **Scale capacity** as your computing requirements change
- Pay only for capacity that **you actually use**
- Choose **Linux** or **Windows**
- Deploy across **AWS Regions** and **Availability Zones** for reliability
- Use **tags** to help manage Amazon EC2 resources

Launching an EC2 Instance via the Management Console

1. **Determine the AWS Region** in which you want to launch the EC2 instance.

2. **Launch** an EC2 instance from a pre-configured Amazon Machine Image (AMI).

3. **Choose an instance type** based on CPU, memory, storage, and network requirements.

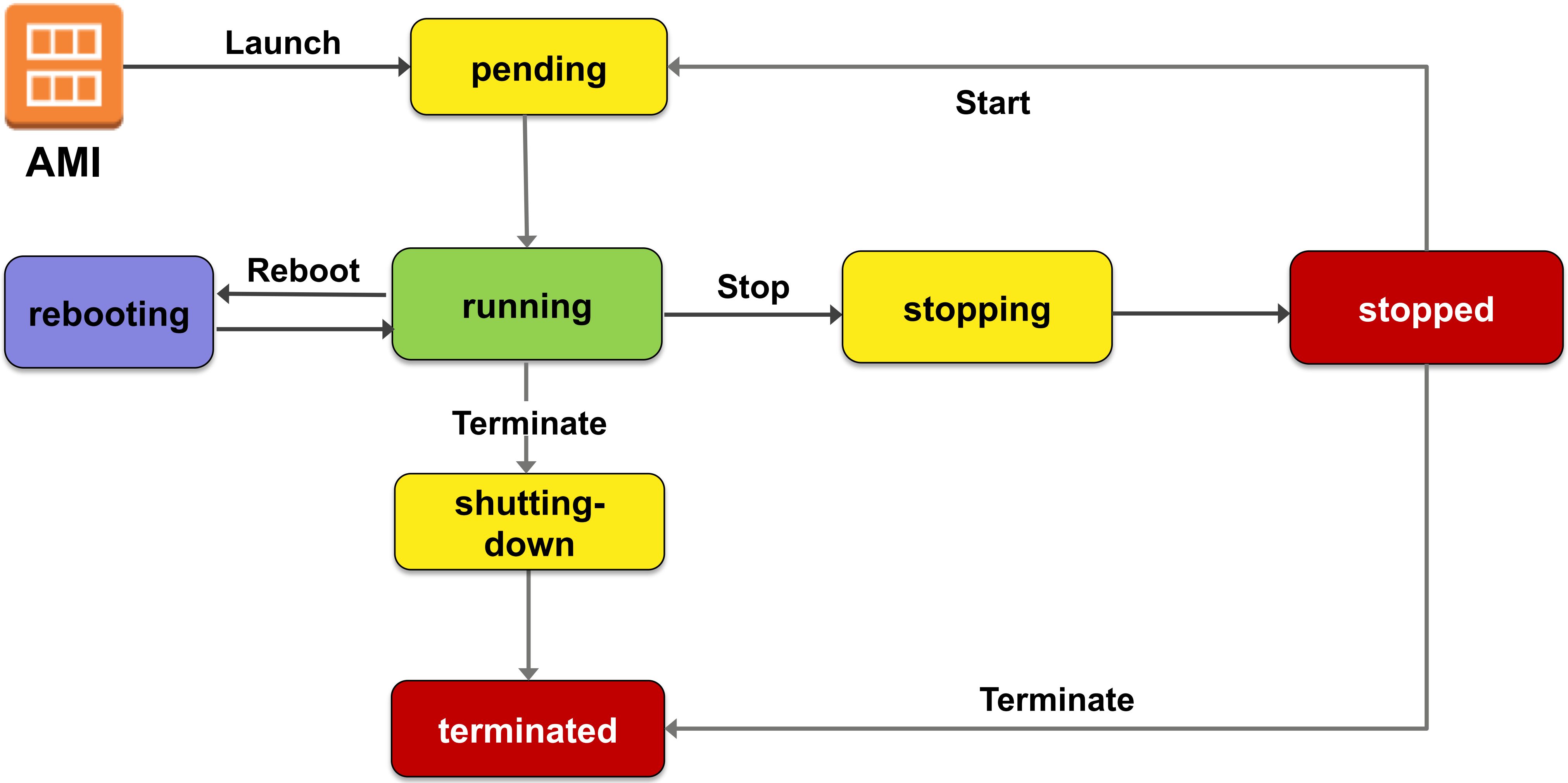
4. **Configure** network, IP address, security groups, storage volume, tags, and key pair.

Amazon Machine Image (AMI) Details

An AMI includes the following:

- A template for the root volume for the instance (for example, an operating system, an application server, and applications).
- Launch permissions that control which AWS accounts can use the AMI to launch instances.
- A block device mapping that specifies the volumes to attach to the instance when it is launched.

Instance Lifecycle



Instance Metadata

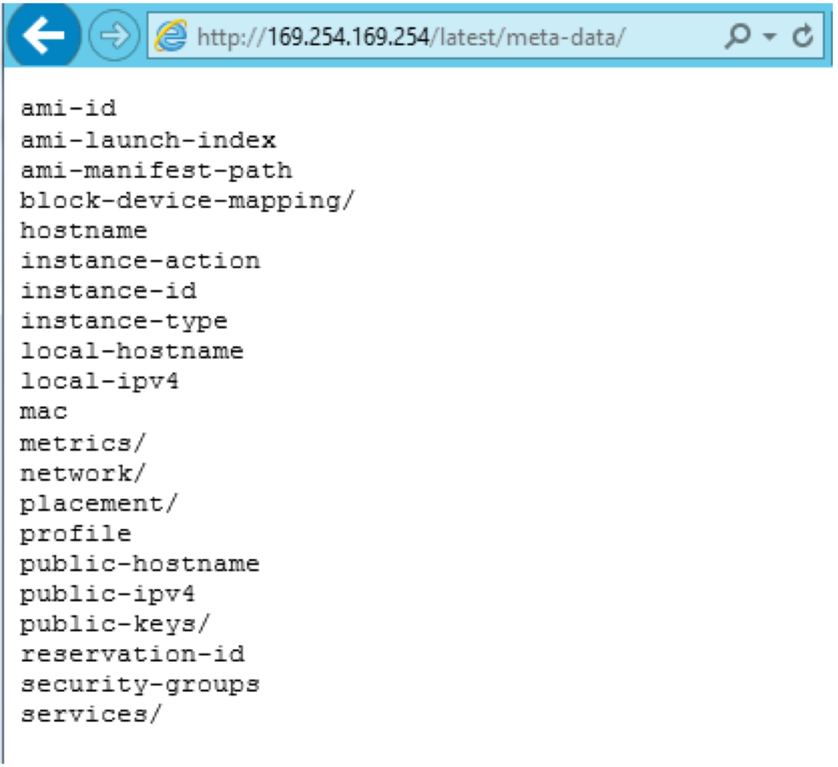
Is **data** about an **instance**.

Can be used to **configure or manage** a running instance.

Retrieving Instance Metadata

To view all categories of instance metadata from within a running instance, use the following URI:

```
http://169.254.169.254/latest/meta-data/
```



http://169.254.169.254/latest/meta-data/

```
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
hostname
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
```

On a Linux instance, you can use:

```
$ curl http://169.254.169.254/latest/meta-data/
$ GET http://169.254.169.254/latest/meta-data/
```

Instance User Data

Can be passed to the instance **at launch**.

Can be used to perform common **automated configuration tasks**.

Runs scripts after the instance starts.

Adding User Data

You can specify user data when launching an instance.

User data can be:

- Linux script
- Windows batch or PowerShell scripts

User data scripts run once per instance by default.

User Data Example Linux

```
#!/bin/sh
yum -y install httpd
systemctl enable httpd
service httpd start
```

User data shell scripts must start with the #! characters and the path to the interpreter you want to read the script.

Install Apache web server
Enable the web server
Start the web server

Amazon Simple Storage Service (S3)

Amazon S3

- Storage for the Internet
- Natively online, HTTP access
- Highly scalable, reliable, fast and durable

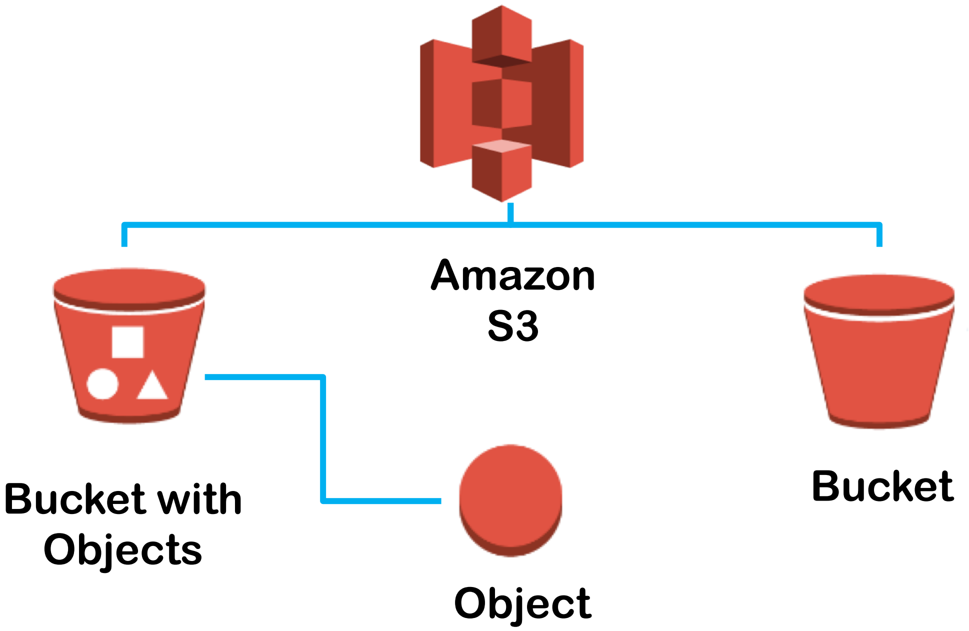
S3

- Can store an **unlimited number of objects** in a bucket
- Objects can be **up to 5 TB**; no bucket size limit
- Designed for **99.999999999%** durability and **99.99%** availability of objects over a given year
- Can use **HTTP/S** endpoints to store and retrieve any amount of data, at any time, from anywhere on the web
- Is highly scalable, reliable, fast, and inexpensive
- Can use optional server-side **encryption** using AWS or customer-managed provided client-side encryption
- Auditing is provided by access logs
- Provides standards-based **REST** and SOAP interfaces

Common Use Scenarios

- Storage and backup
- Application file hosting
- Media hosting
- Software delivery
- Store AMIs and snapshots

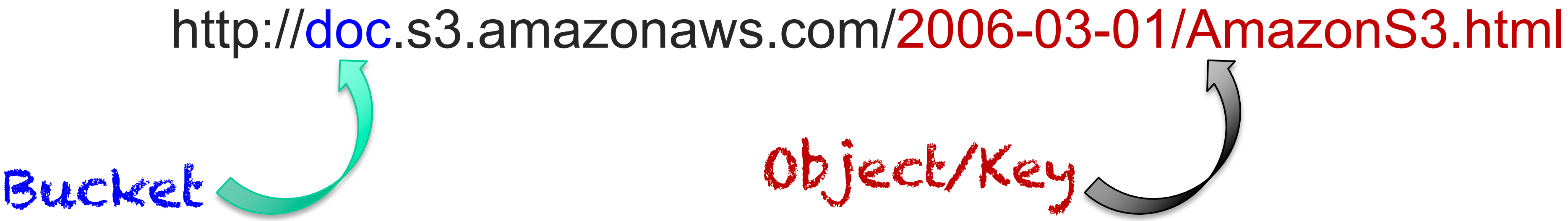
S3 Concepts



- Amazon S3 stores data as objects within **buckets**
- An object is composed of a file and optionally any **metadata** that describes that file
- You can have **up to 100 buckets** in each account
- You can **control access** to the bucket and its objects

Object Keys

An object key is the unique identifier for an object in a bucket:



http://doc.s3.amazonaws.com/2006-03-01/AmazonS3.html

Bucket

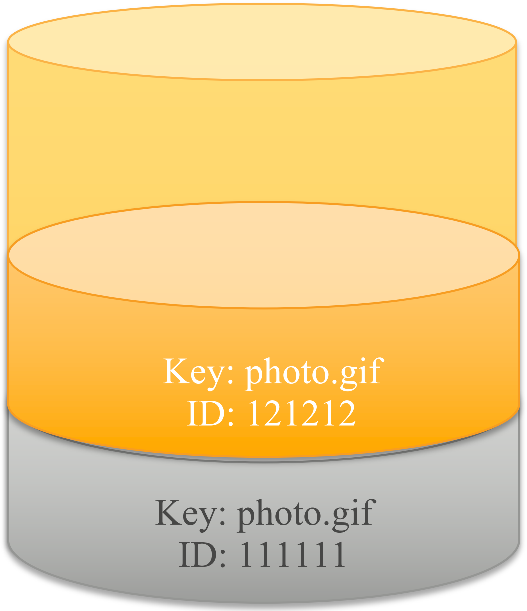
Object/Key

S3 Security

- You can **control access** to buckets and objects with:
  - Access Control Lists (ACLs)
  - Bucket policies
  - Identity and Access Management (IAM) policies
- You can upload or download data to S3 via **SSL** encrypted endpoints.
- You can **encrypt data** using AWS SDKs.

S3 Versioning

- Protects from **accidental overwrites and deletes** with no performance penalty
- Generates a **new version with every upload**
- Allows easily retrieval of deleted objects or **roll back** to previous versions
- Three states of an S3 bucket
  - Un-versioned (default)
  - Versioning-enabled
  - Versioning-suspended



Versioning Enabled