Developer Operations

AWS Elastic Compute Cloud (EC2)

AWS Simple Storage Service (S3)

Amazon Elastic Compute Cloud (EC2)

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

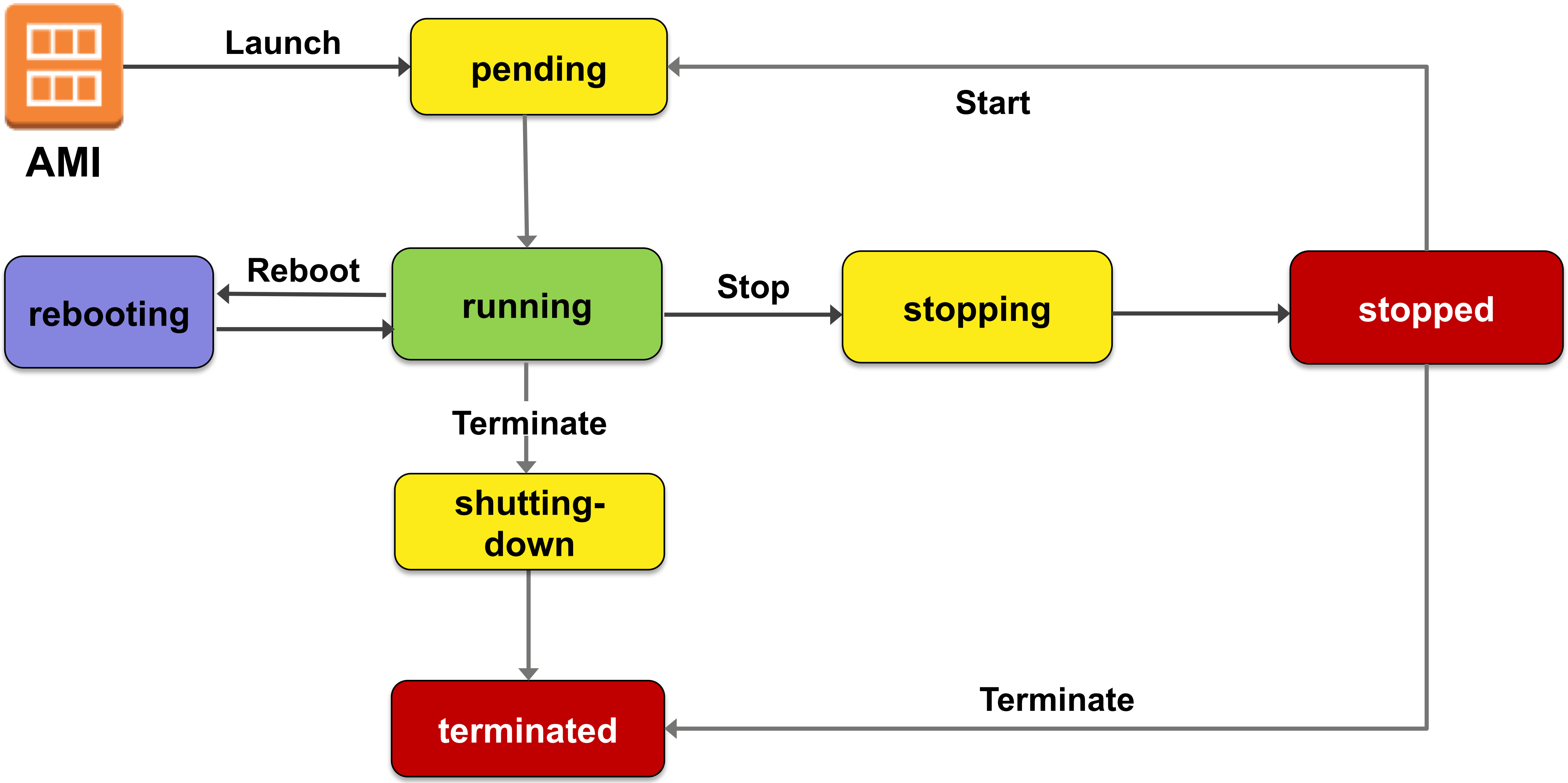
EC2 Features

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

Launching an EC2 Instance

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)
   - AMI contains root volume **template** and various other features such as permissions on who can use it
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

Instance Lifecycle



AMI

Launch

pending

Start

Reboot

rebooting

running

Stop

stopping

stopped

Terminate

shutting-down

terminated

Terminate

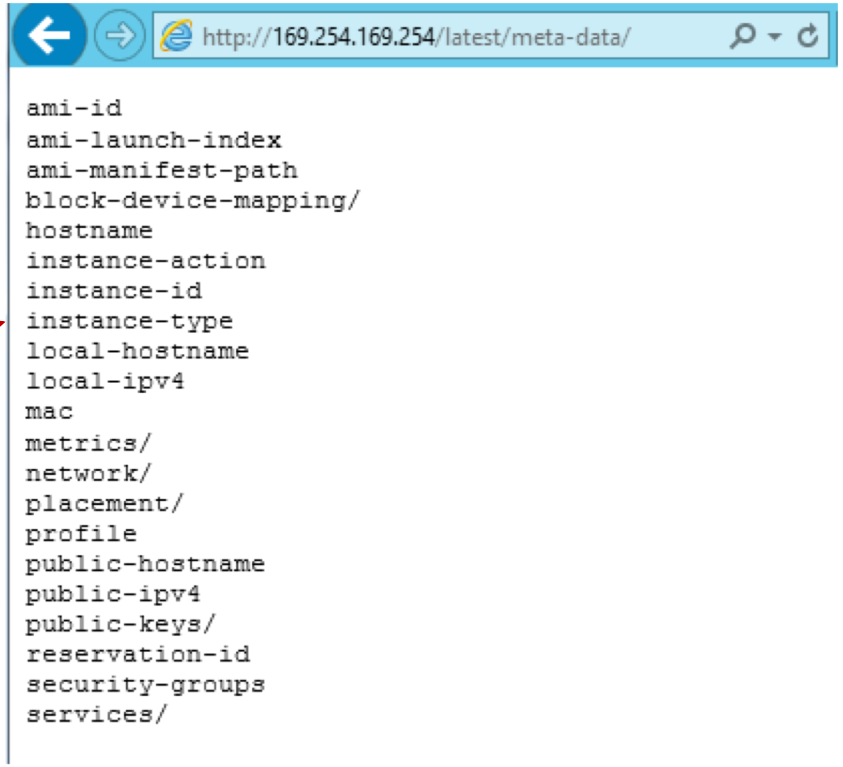
Instance Metadata

- Data about an instance. Can be used to help manage a running instance.
- To view all categories of instance metadata from within a running instance, use the following URI:

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

- 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/
```



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/
```

Instance User Data

- You can (optionally) specify User Data when launching an instance.
- User data can be a Linux (or Windows) shell script
  - Which can in turn run Python or any other code
- This script is be passed to the instance at launch
- Runs immediately after the instance starts
- Can be used to perform common automated configuration tasks.

User Data Example (Linux)

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

Specify path to interpreter you want to read the script

Install Apache web server

Enable the web server

Start the web server

Amazon Simple Storage Service (S3)

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

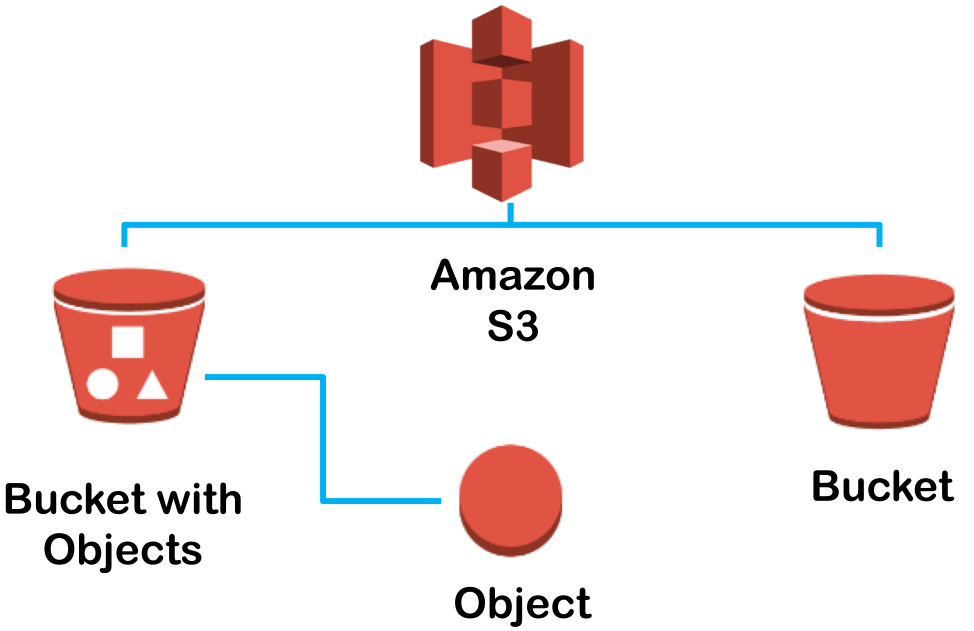
S3

- 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 data
- Highly scalable, reliable, fast, and inexpensive
- Optional server-side or client-side encryption
- Access logs
- REST and SOAP interfaces

Common Use Scenarios

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

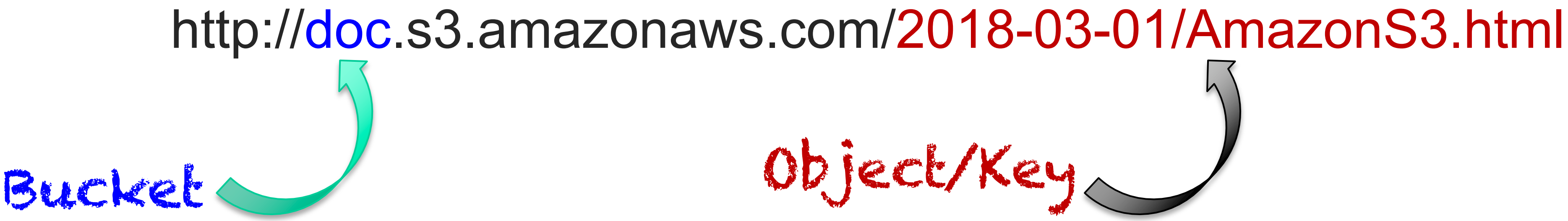
S3 Concepts



- 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/2018-03-01/AmazonS3.html

Bucket

Object/Key