AWS EC2 Concepts

1. Amazon Machine Image (AMI)

An AMI is a preconfigured template for launching EC2 instances, containing the OS, software, and configurations. It serves as a blueprint for creating identical virtual servers.

Components of an AMI:

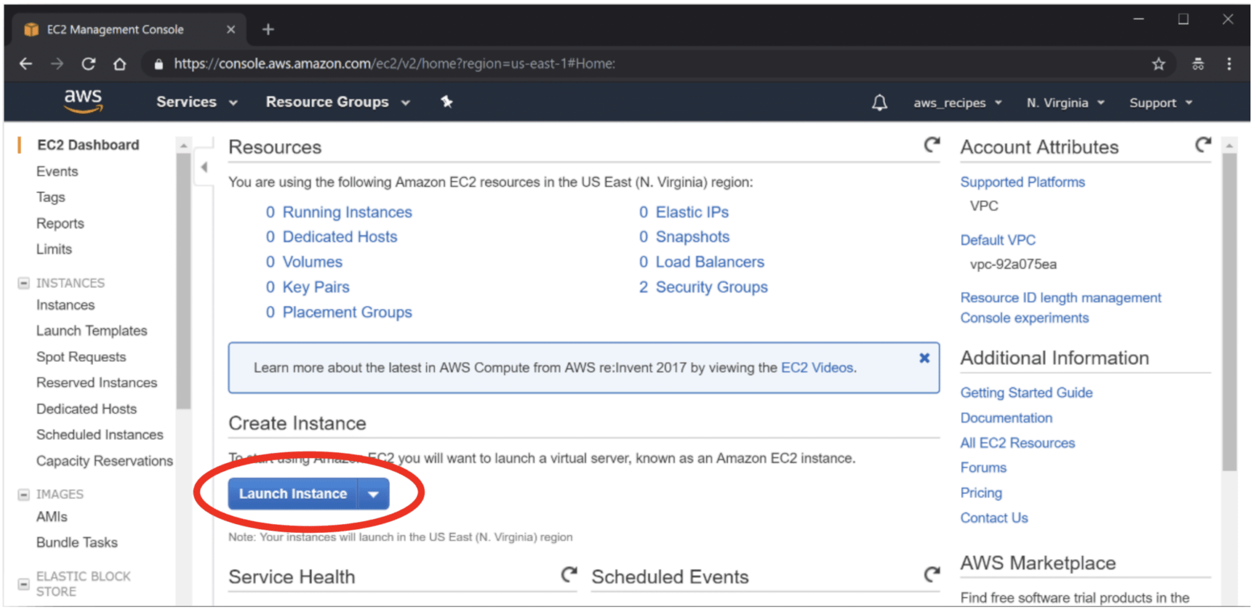
- Root volume template (OS, libraries, applications)
- Launch permissions (controls which AWS accounts can use it)
- Block device mappings (defines attached storage volumes)

Creating a Custom AMI:

1. **From an EC2 Instance:**
    - Launch an instance using a base AMI (e.g., Amazon Linux, Ubuntu).
    - Customize the instance (install software, configure settings).
    - Create an image via the EC2 console or CLI. AWS will snapshot the root volume and register it as a new AMI.
2. **Using EC2 Image Builder:**
    - AWS's automated service for building and maintaining AMIs. Define components (software), create a recipe, and let Image Builder handle the process.
3. **From Snapshots:**
    - Create an AMI from an existing EBS snapshot with the correct boot configuration.

Steps to Create an AMI from an EC2 Instance:

1. Navigate to **EC2 Dashboard → Instances**.
2. Right-click the target instance → **Image and Templates → Create Image**.
3. Configure optional settings (name, description, volume size, tags).



Resources

You are using the following Amazon EC2 resources in the US East (N. Virginia) region:

0 Running Instances	0 Elastic IPs
0 Dedicated Hosts	0 Snapshots
0 Volumes	0 Load Balancers
0 Key Pairs	2 Security Groups
0 Placement Groups	

Learn more about the latest in AWS Compute from AWS re:Invent 2017 by viewing the EC2 Videos.

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Note: Your instances will launch in the US East (N. Virginia) region

Service Health

Scheduled Events

Account Attributes

Supported Platforms
VPC

Default VPC
vpc-92a075ea

Resource ID length management
Console experiments

Additional Information

Getting Started Guide
Documentation
All EC2 Resources
Forums
Pricing
Contact Us

AWS Marketplace

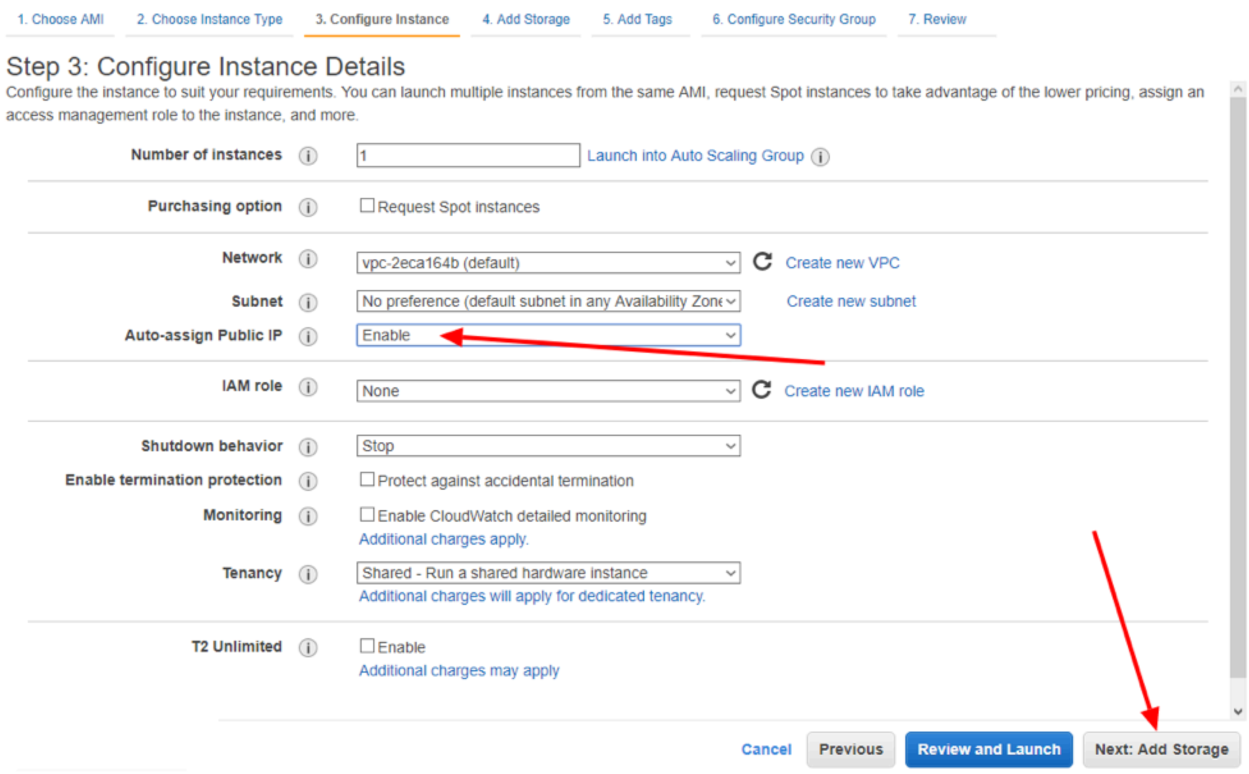
Find free software trial products in the

2. EC2 Instance Types

Instance types are categorized into families optimized for specific workloads. Naming follows the pattern: **Family + Generation + Size** (e.g., `t3.medium`, `m5.large`).

Key Families:

- **General Purpose (M5, M6i, M7i):** Balanced CPU, memory, and networking. Ideal for web servers, small databases, and microservices.
- **Compute Optimized (C5, C6i, C7i):** High CPU performance. Best for scientific computing, gaming servers, and batch processing.
- **Memory Optimized (R5, R6i, R7i):** High memory-to-CPU ratio. Suited for in-memory databases and real-time analytics.
- **Storage Optimized (I3, I4i):** High-speed NVMe storage. Designed for NoSQL databases and data warehousing.
- **Accelerated Computing (P3, P4):** GPU/FPGA acceleration. Used for ML, HPC, and graphics workloads.



Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 Launch into Auto Scaling Group

Purchasing option: □ Request Spot instances

Network: vpc-2eca164b (default) Create new VPC

Subnet: No preference (default subnet in any Availability Zone Create new subnet

Auto-assign Public IP: Enable

IAM role: None Create new IAM role

Shutdown behavior: Stop

Enable termination protection: □ Protect against accidental termination

Monitoring: □ Enable CloudWatch detailed monitoring
Additional charges apply.

Tenancy: Shared - Run a shared hardware instance
Additional charges will apply for dedicated tenancy.

T2 Unlimited: □ Enable
Additional charges may apply

Burstable (T3/T4g) vs. General Purpose (M5/M6i):

Feature	Burstable (T3)	General Purpose (M5)
Performance	Variable (credit-based bursting)	Consistent, full CPU access
Cost	Lower baseline cost	Higher but predictable
Use Cases	Intermittent workloads, cost savings	Steady production workloads

When to Choose:

- **Burstable:** For variable traffic (e.g., dev environments, small web servers).
- **General Purpose:** For production apps requiring predictable performance.

3. EC2 Key Pairs

Key pairs enable secure SSH/RDP access to instances using public-key cryptography.

Key File Types:

- **.pem:** Standard for Linux/macOS (e.g., `ssh -i key.pem ec2-user@IP`).

- **.ppk:** Required for Windows PuTTY (convert .pem via PuTTYgen).

Authentication Process:

1. Specify a key pair at launch. AWS places the public key in `~/.ssh/authorized_keys`.
2. The private key authenticates the user.

4. Security Groups

Virtual firewalls controlling instance-level traffic. Rules are stateful (allow inbound → outbound replies auto-allowed).

Common Inbound Rules:

- **Web Server:** Allow HTTP (80) and HTTPS (443) from `0.0.0.0/0`.
- **SSH:** Restrict port 22 to specific IPs (e.g., `203.0.113.0/24`).
- **Database:** Allow MySQL (3306) from a web server's security group.

Outbound Rules:

- Default: Allow all. Restrict for security (e.g., HTTPS only to `0.0.0.0/0`).

Security Groups vs. NACLs:

Feature	Security Groups	NACLs
Scope	Instance-level	Subnet-level
State	Stateful	Stateless
Rules	Allow only	Allow/deny

5. EBS (Elastic Block Storage)

Persistent network-attached storage for EC2 instances.

Volume Types:

- **General Purpose SSD (gp3):** Baseline 3,000 IOPS, scalable to 16,000. Cost-effective for most workloads.
- **Provisioned IOPS SSD (io2):** Up to 64,000 IOPS for high-performance databases.

- **Throughput HDD (st1):** Low-cost, ideal for big data/log processing.

Modifying EBS Volumes:

- Resize, change type, or adjust IOPS **without downtime** via **EC2 → Volumes → Modify Volume**.