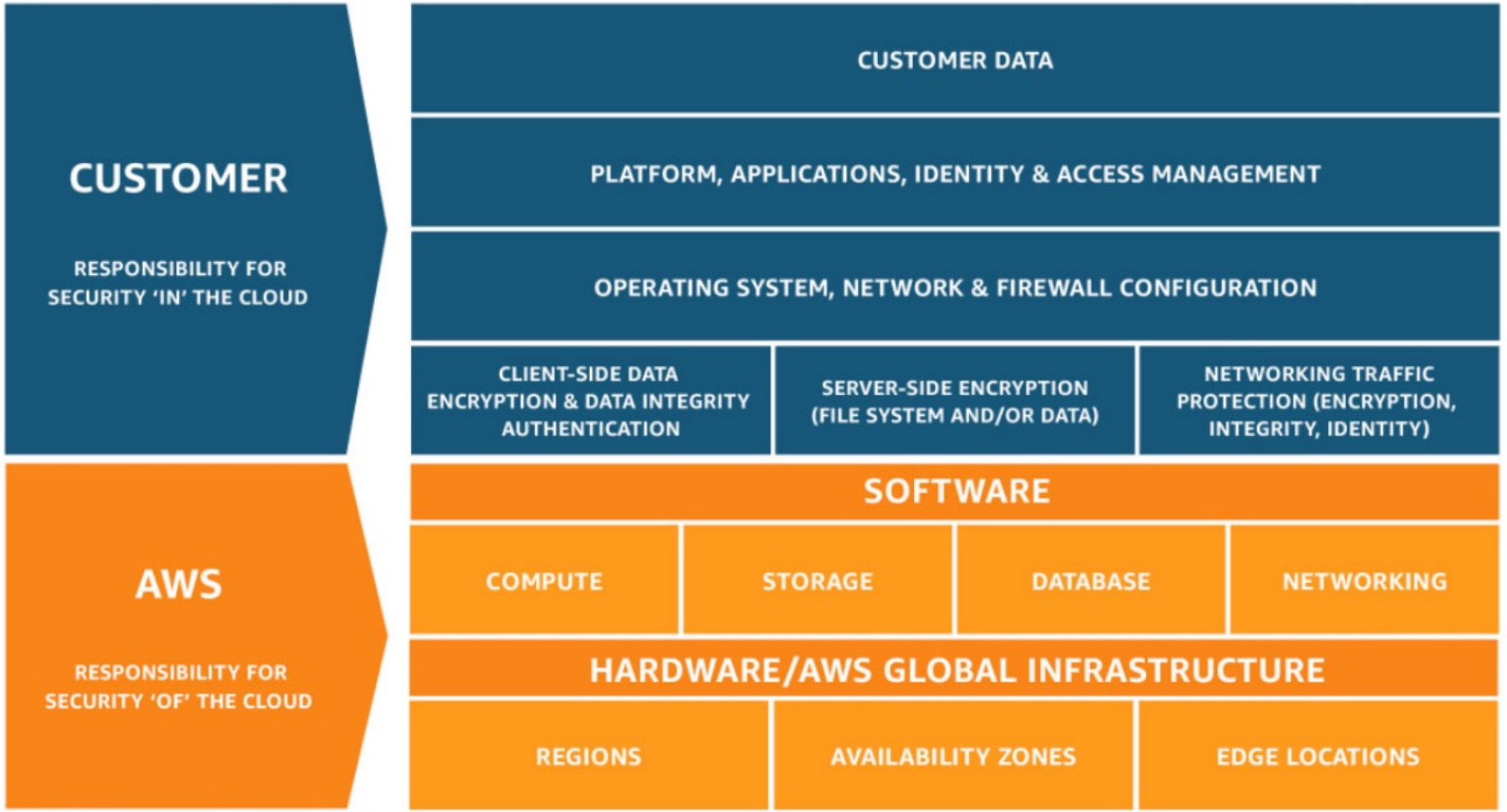
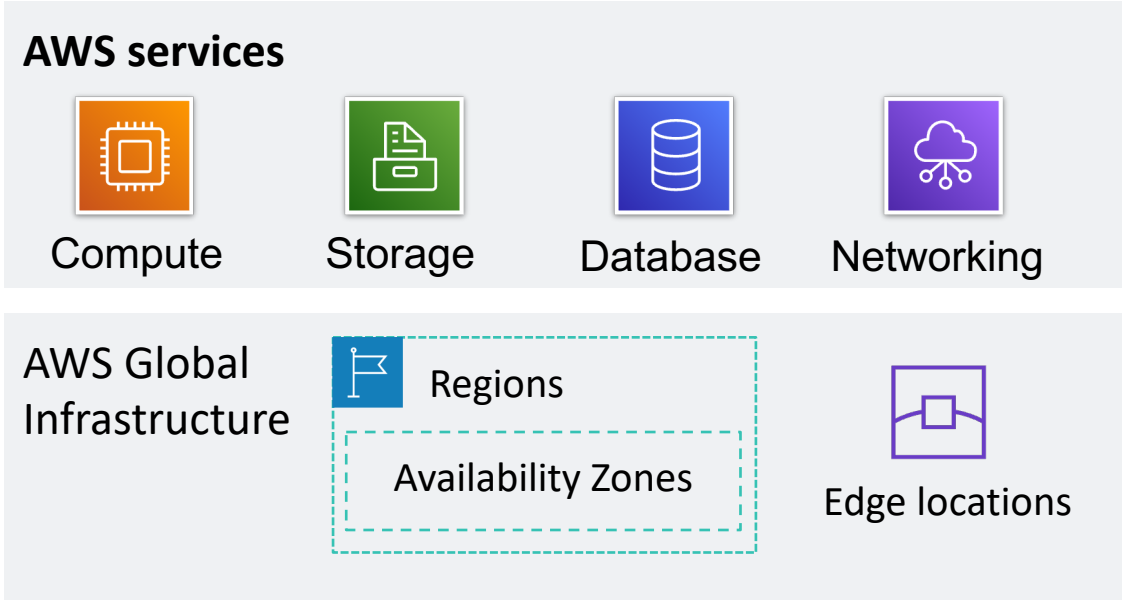
Module 4: AWS Cloud Security

AWS Academy Cloud Foundations

AWS shared responsibility model



AWS responsibility: Security *of* the cloud

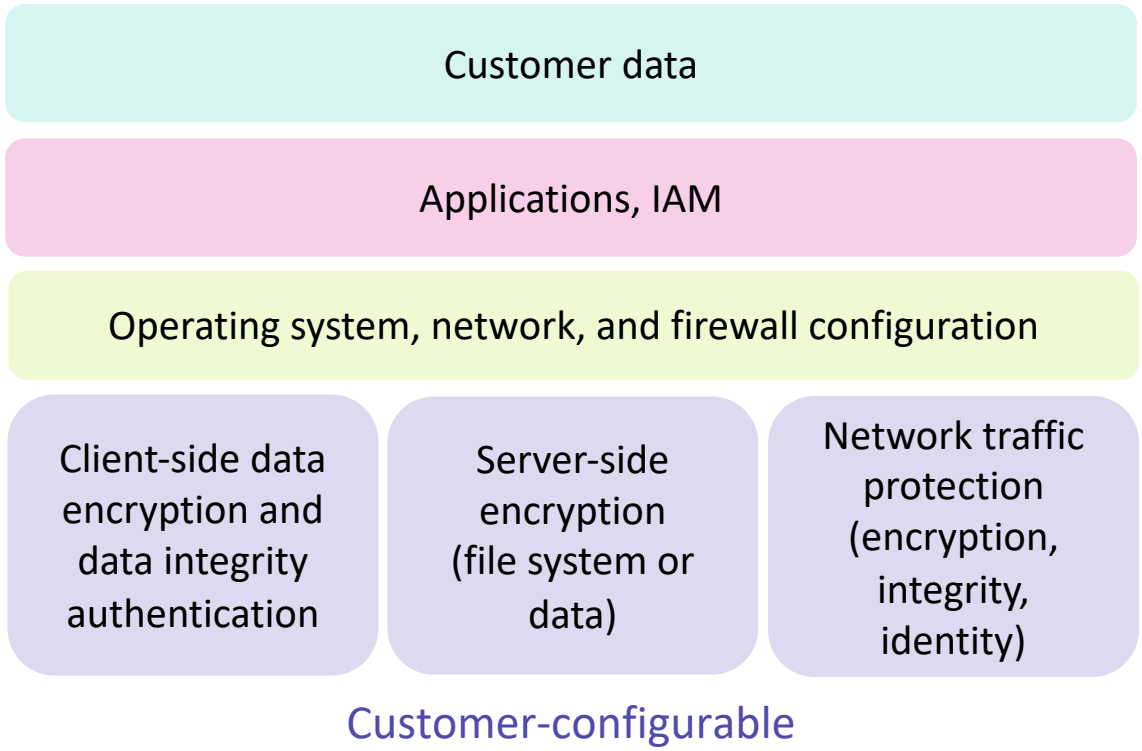


AWS responsibilities:

- Physical security of data centers
  - Controlled, need-based access
- Hardware and software infrastructure
  - Storage decommissioning, host operating system (OS) access logging, and auditing

- Network infrastructure
  - Intrusion detection
- Virtualization infrastructure
  - Instance isolation

Customer responsibility: Security *in* the cloud

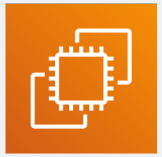


Customer responsibilities:

- Amazon Elastic Compute Cloud (Amazon EC2) instance operating system
  - Including patching, maintenance
- Applications
  - Passwords, role-based access, etc.
- Security group configuration
- OS or host-based firewalls
  - Including intrusion detection or prevention systems
- Network configurations
- Account management
  - Login and permission settings for each user

Service characteristics and security responsibility (1 of 2)

Example services managed by the customer



Amazon EC2

Amazon Elastic Block Store (Amazon EBS)

Amazon Virtual Private Cloud (Amazon VPC)

Example services managed by AWS

AWS Lambda

Amazon Relational Database Service (Amazon RDS)

AWS Elastic Beanstalk

Infrastructure as a service (IaaS)

- Customer has more flexibility over configuring networking and storage settings
- Customer is responsible for managing more aspects of the security
- Customer configures the access controls

Platform as a service (PaaS)

- Customer does not need to manage the underlying infrastructure
- AWS handles the operating system, database patching, firewall configuration, and disaster recovery
- Customer can focus on managing code or data

Service characteristics and security responsibility (2 of 2)

SaaS examples

AWS Trusted Advisor

AWS Shield

Amazon Chime

Software as a service (SaaS)

- Software is centrally hosted
- Licensed on a subscription model or pay-as-you-go basis.
- Services are typically accessed via web browser, mobile app, or application programming interface (API)
- Customers do not need to manage the infrastructure that supports the service

Section 2: AWS Identity and Access Management (IAM)

Module 4: AWS Cloud Security

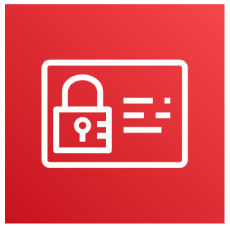
AWS Identity and Access Management (IAM)

- Use **IAM** to manage access to **AWS resources** –
  - A resource is an entity in an AWS account that you can work with
  - Example resources; An Amazon EC2 instance or an Amazon S3 bucket

- *Example* – Control who can terminate Amazon EC2 instances

- Define fine-grained access rights –
  - **Who** can access the resource
  - **Which** resources can be accessed and what can the user do to the resource
  - **How** resources can be accessed

- IAM is a no-cost AWS account feature



AWS Identity and Access Management (IAM)

IAM: Essential components

IAM user

A **person** *or* **application** that can authenticate with an AWS account.

IAM group

A **collection of IAM users** that are granted identical authorization.

IAM policy

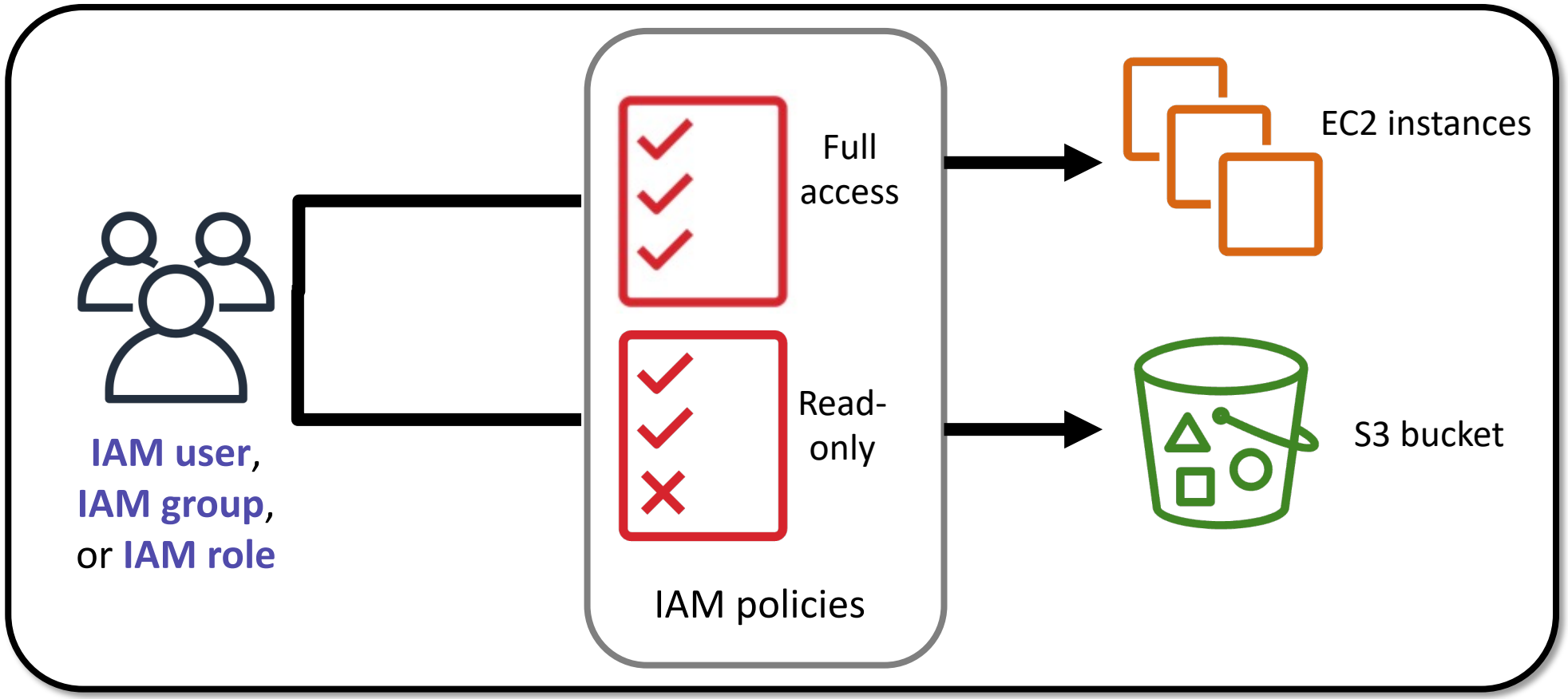
The document that defines **which resources can be accessed** and the **level of access** to each resource.

IAM role

Useful mechanism to grant a set of permissions for making AWS service requests.

Authorization: What actions are permitted

After the user or application is connected to the AWS account, what are they allowed to do?



IAM: Authorization

- Assign permissions by creating an IAM policy.
- Permissions determine **which resources and operations** are allowed:
  - All permissions are implicitly denied by default.
  - If something is explicitly denied, it is never allowed.

Best practice: Follow the **principle of least privilege**.

IAM permissions

Note: The scope of IAM service configurations is **global**. Settings apply across all AWS Regions.

IAM policy example

```
{
  "Version": "2012-10-17",
  "Statement":[{
    "Effect":"Allow",
    "Action":["DynamoDB:*","s3:*"],
    "Resource":[
      "arn:aws:dynamodb:region:account-number-without-hyphens:table/table-name",
      "arn:aws:s3:::bucket-name",
      "arn:aws:s3:::bucket-name/*"]
    },
    {
    "Effect":"Deny",
    "Action":["dynamodb:*","s3:*"],
    "NotResource":["arn:aws:dynamodb:region:account-number-without-hyphens:table/table-name”,
      "arn:aws:s3:::bucket-name",
      "arn:aws:s3:::bucket-name/*"]
    }
  ]
}
```

Explicit allow gives users access to a specific DynamoDB table and…

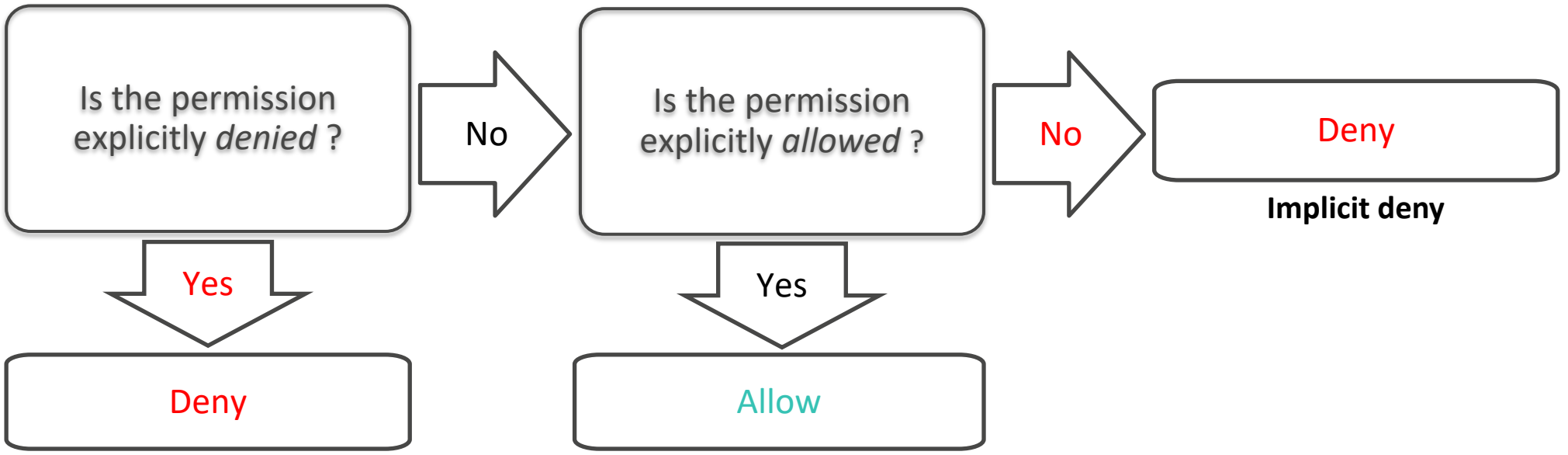
…Amazon S3 buckets.

Explicit deny ensures that the users cannot use any other AWS actions or resources other than that table and those buckets.

An explicit deny statement **takes precedence** over an allow statement.

IAM permissions

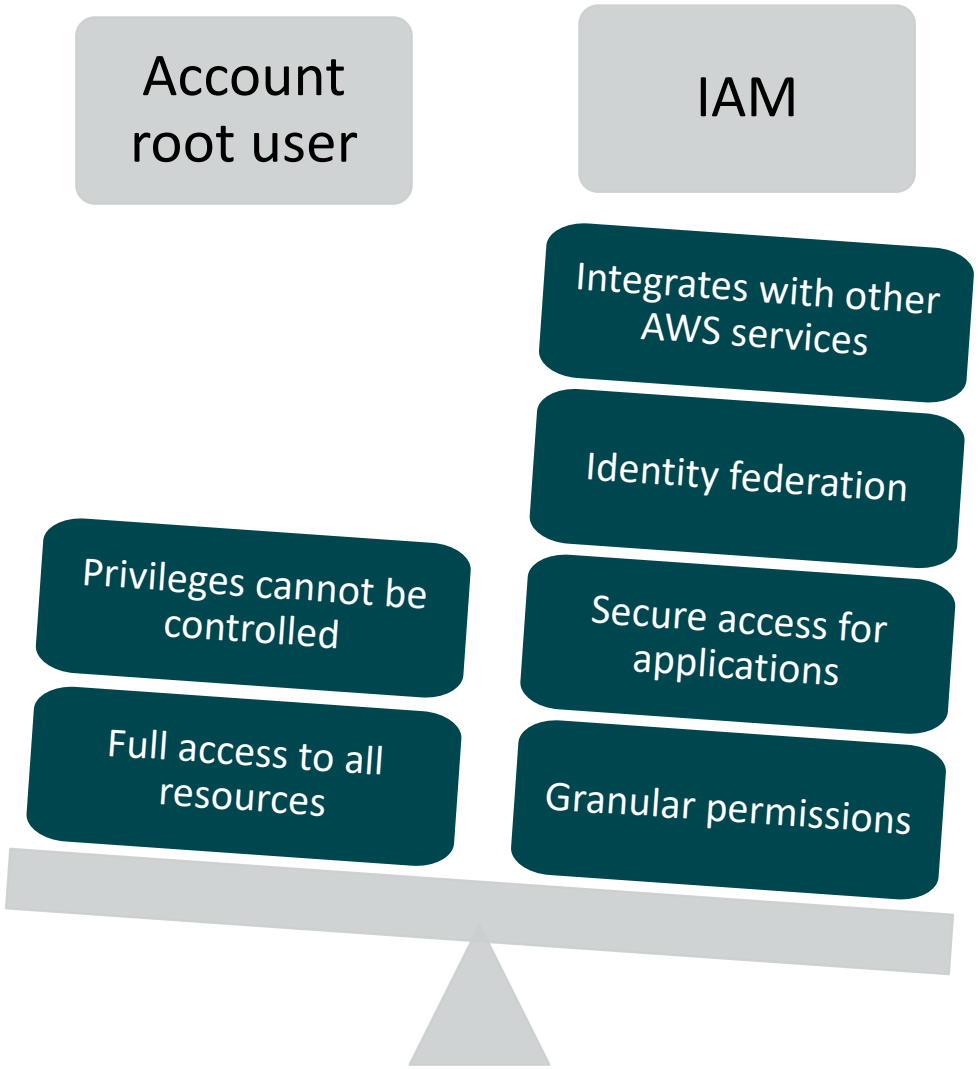
How IAM determines permissions:



Section 3: Securing a new AWS account

Module 4: AWS Cloud Security

AWS account root user access versus IAM access



- **Best practice**: Do not use the AWS account root user except when necessary.
  - Access to the **account root user** requires logging in with the *email address* (and password) that you used to create the account.
- Example actions that can only be done with the account root user:
  - Update the account root user password
  - Change the AWS Support plan
  - Restore an IAM user's permissions
  - Change account settings (for example, contact information, allowed Regions)

Securing a new AWS account: Account root user

Step 1: Stop using the account root user as soon as possible.

- The account root user has unrestricted access to all your resources.

- To stop using the account root user:
  1. While you are logged in as the account root user, create an IAM user for yourself. Save the access keys if needed.
  2. Create an IAM group, give it full administrator permissions, and add the IAM user to the group.
  3. Disable and remove your account root user access keys, if they exist.
  4. Enable a password policy for users.
  5. Sign in with your new IAM user credentials.
  6. Store your account root user credentials in a secure place.

Securing a new AWS account: MFA

Step 2: Enable multi-factor authentication (MFA).

- Require MFA for your account root user and for all IAM users.
- You can also use MFA to control access to AWS service APIs.

- Options for retrieving the MFA token –
  - Virtual MFA-compliant applications:
    - Google Authenticator.
    - Authy Authenticator (Windows phone app).
  - U2F security key devices:
    - For example, YubiKey.
  - Hardware MFA options:
    - Key fob or display card offered by Gemalto.

MFA token

Securing a new AWS account: AWS CloudTrail

Step 3: Use AWS CloudTrail.

- CloudTrail tracks user activity on your account.
  - Logs all API requests to resources in all supported services your account.
- Basic AWS CloudTrail event history is enabled by default and is free.
  - It contains all management event data on latest 90 days of account activity.
- To access CloudTrail –
  1. Log in to the **AWS Management Console** and choose the **CloudTrail** service.
  2. Click **Event history** to view, filter, and search the last 90 days of events.
- **To enable logs beyond 90 days and enable specified event alerting, create a trail.**
  1. From the CloudTrail Console trails page, click **Create trail**.
  2. Give it a name, apply it to all Regions, and create a new Amazon S3 bucket for log storage.
  3. Configure access restrictions on the S3 bucket (for example, only admin users should have access).

Securing a new AWS account: Billing reports

Step 4: Enable a billing report, **such as the AWS Cost and Usage Report.**

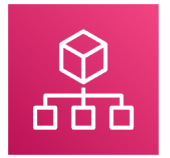
- Billing reports provide information about your use of AWS resources and estimated costs for that use.
- AWS delivers the reports to an Amazon S3 bucket that you specify.
    - Report is updated at least once per day.
- The **AWS Cost and Usage Report** tracks your AWS usage and provides estimated charges associated with your AWS account, either by the hour or by the day.

Section 4: Securing accounts

Module 4: AWS Cloud Security

AWS Organizations

- **AWS Organizations** enables you to consolidate multiple AWS accounts so that you centrally manage them.



AWS Organizations

- Security features of AWS Organizations:
  - **Group AWS accounts into organizational units** (OUs) and attach different access policies to each OU.
  - **Integration and support for IAM**
    - Permissions to a user are the intersection of what is allowed by AWS Organizations and what is granted by IAM in that account.
  - **Use service control policies** to establish control over the AWS services and API actions that each AWS account can access

AWS Organizations: Service control policies

- **Service control policies (SCPs)** offer centralized control over accounts.
  - Limit permissions that are available in an account that is part of an organization.
- Ensures that accounts comply with access control guidelines.
- SCPs are *similar* to IAM permissions policies –
  - They use similar syntax.
  - However, an SCP never grants permissions.
  - Instead, SCPs specify the maximum permissions for an organization.

AWS Key Management Service (AWS KMS)

AWS Key Management Service (AWS KMS) features:

- Enables you to **create and manage encryption keys**
- Enables you to control the use of encryption across AWS services and in your applications.
- Integrates with AWS CloudTrail to log all key usage.
- Uses hardware security modules (HSMs) that are validated by Federal Information Processing Standards (FIPS) 140-2 to protect keys

AWS Key Management Service (AWS KMS)

Amazon Cognito

Amazon Cognito features:

- **Adds user sign-up, sign-in, and access control to your web and mobile applications**.
- Scales to millions of users.
- Supports sign-in with social identity providers, such as Facebook, Google, and Amazon; and enterprise identity providers, such as Microsoft Active Directory via Security Assertion Markup Language (SAML) 2.0.

Amazon Cognito

AWS Shield

- **AWS Shield** features:
  - Is a managed distributed denial of service (DDoS) protection service
  - Safeguards applications running on AWS
  - Provides always-on detection and automatic inline mitigations
  - *AWS Shield Standard* enabled for at no additional cost. *AWS Shield Advanced* is an optional paid service.
- Use it to **minimize application downtime and latency.**

AWS Shield

Section 5: Securing data on AWS

Module 4: AWS Cloud Security

Encryption of data *at rest*

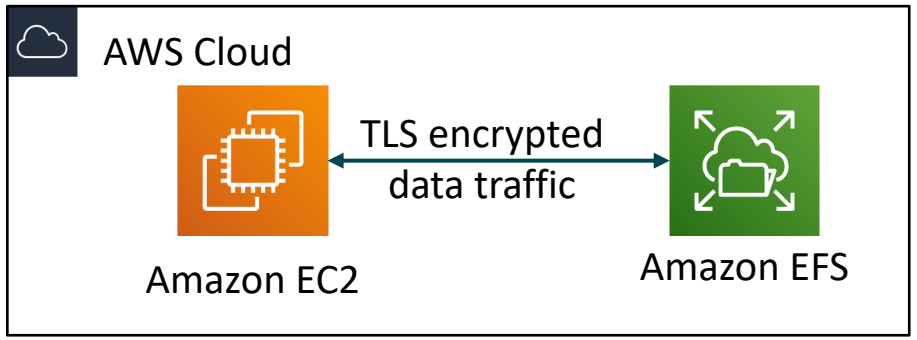
- **Encryption** encodes data with a secret key, which makes it unreadable
  - Only those who have the secret key can decode the data
  - AWS KMS can manage your secret keys

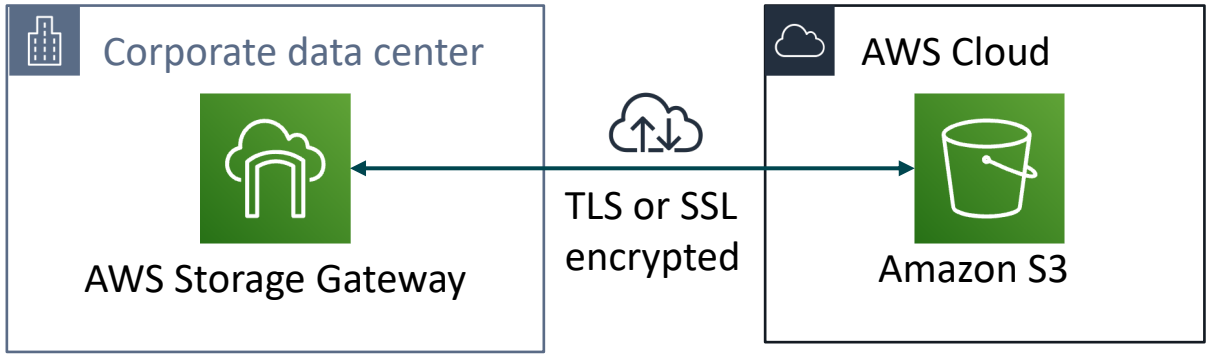
- AWS supports encryption of **data at rest**
  - Data at rest = Data stored physically (on disk or on tape)

  - You can encrypt data stored in any service that is supported by AWS KMS, including:
    - Amazon S3
    - Amazon EBS
    - Amazon Elastic File System (Amazon EFS)
    - Amazon RDS managed databases

Encryption of data *in transit*

- Encryption of **data in transit** (data moving across a network)
  - **Transport Layer Security (TLS)**—formerly SSL—is an open standard protocol
  - **AWS Certificate Manager** provides a way to manage, deploy, and renew TLS or SSL certificates
- Secure HTTP (HTTPS) creates a secure tunnel
  - Uses TLS or SSL for the bidirectional exchange of data
- **AWS services support data in transit encryption**.
  - Two examples:





Section 6: Working to ensure compliance

Module 4: AWS Cloud Security

AWS compliance programs

- Customers are subject to many different security and compliance regulations and requirements.
- **AWS engages with certifying bodies and independent auditors to provide customers with detailed information about the policies, processes, and controls that are established and operated by AWS.**

- Compliance programs can be broadly categorized –
  - **Certifications and attestations**
    - Assessed by a third-party, independent auditor
    - Examples: ISO 27001, 27017, 27018, and ISO/IEC 9001
  - **Laws, regulations, and privacy**
    - AWS provides security features and legal agreements to support compliance
    - Examples: EU General Data Protection Regulation (GDPR), HIPAA
  - **Alignments and frameworks**
    - Industry- or function-specific security or compliance requirements
    - Examples: Center for Internet Security (CIS), EU-US Privacy Shield certified