Q: Can two EC2 instances in the same subnet be isolated from each other?

Yes! Even if two EC2 instances are in the same subnet, we can block them from talking to each other using different methods.

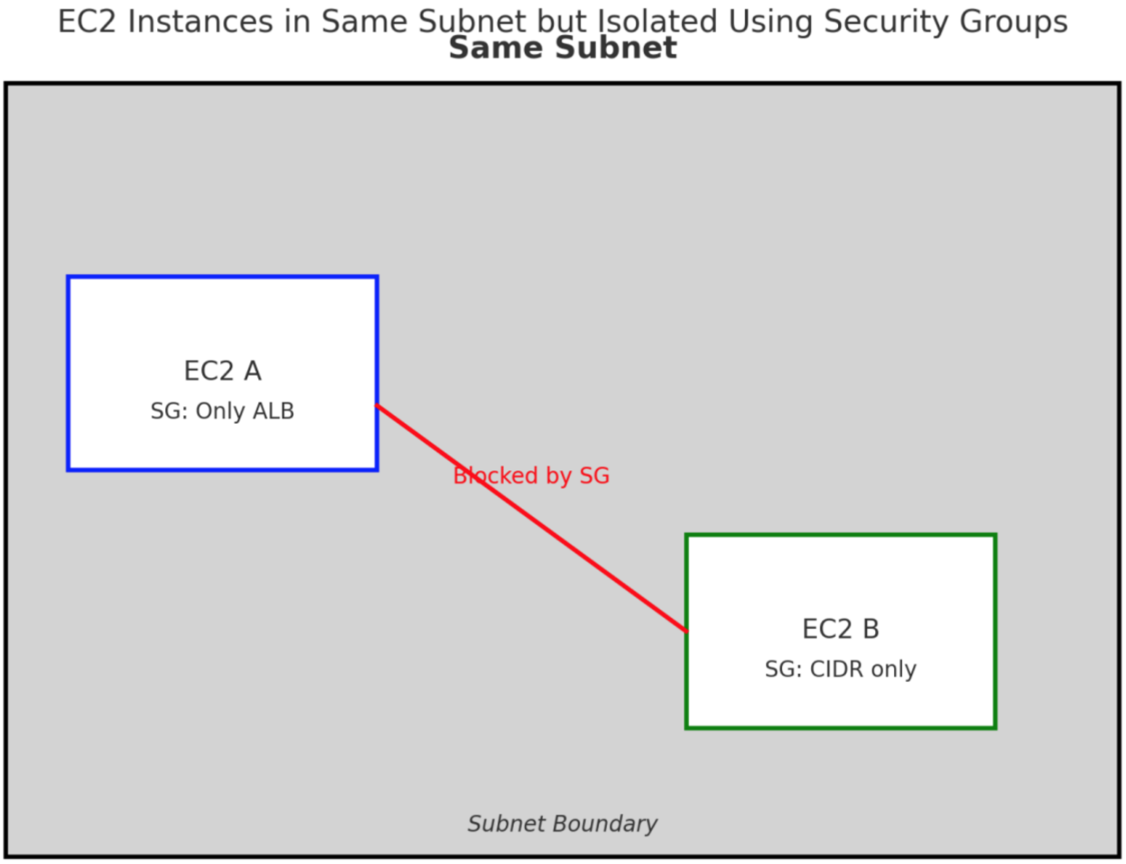
Method 1: Security Groups (Most Common & Recommended)

- **Think of Security Groups like walls with doors you can lock or open.**
- **Assign different security groups to each instance.**
- **Set rules to allow only specific traffic (like from an ALB or your laptop).**
- **Do not allow traffic from other instances.**

✅ **Easy to set up, works well, and automatically allows reply traffic.**

Example:

- **Instance A allows traffic only from the Load Balancer.**
- **Instance B allows traffic only from a known IP (e.g., office).**
- **They can't see each other = isolated.**



🖥️ Method 2: OS-level Firewall

- Each EC2 instance can use its own firewall like iptables (Linux) or Windows Firewall.
- You can block or allow traffic at the operating system level.

Works even without AWS configuration, but setup is manual.

Method 3: Application-level Control

- Your app itself can be designed to reject unwanted traffic.
- Example: Bind app only to localhost, or require login/authentication.

🛠️ Not very common for isolation, but adds an extra layer.

Best Practices

- ✅ Use Security Groups as your first line of defense.
- ✅ Follow least privilege – allow only what's needed.
- ✅ Regularly review and test your rules.
- ✅ Use multiple methods for stronger security (defense in depth).

Note

- Even when isolated, EC2s may:
  - Access AWS services or APIs
  - Use metadata service
  - Do DNS lookups
- For stronger isolation, put them in separate subnets.

Question 1:

Which of the following is the most common and recommended method to isolate two EC2 instances in the same subnet?

A. Subnet Masking

B. Network ACLs

C. Security Groups

D. Availability Zones

✅ **Correct Answer: C. Security Groups**

📝 **Explanation: Security Groups are the primary method for controlling inbound and outbound traffic to instances, and they are easy to configure for instance-level isolation.**

Question 2:

Why are Network ACLs considered stateless in AWS?

A. They automatically allow return traffic.

B. They only support allow rules.

C. You must define rules for both inbound and outbound traffic.

D. They operate at the instance level.

✅ **Correct Answer: C. You must define rules for both inbound and outbound traffic.**

📝 **Explanation: Network ACLs do not remember connection state, so you must explicitly allow traffic in both directions.**