

Self-Healing Infrastructure

Theory & Conceptual Documentation
AWS Internship — Center of Excellence, KIET

1. Introduction

Self-healing infrastructure is an approach in which infrastructure can automatically detect a known failure or undesired state and take a predefined corrective action without requiring manual intervention. In this project, Amazon EventBridge detects when a specific Amazon EC2 instance enters the **stopped** state and triggers AWS Systems Manager Automation to start the instance again.

The workflow demonstrates an event-driven automation pattern: **EC2** → **EventBridge** → **Systems Manager Automation** → **EC2**.

2. Project Objective

- Detect a stopped state for a selected EC2 test instance.
- Use Amazon EventBridge to match the EC2 state-change event.
- Trigger the AWS Systems Manager Automation runbook **AWS-StartEC2Instance**.
- Use an IAM role to provide the permissions required by the automation.
- Automatically return the EC2 instance to the running state.
- Demonstrate the basic concept of self-healing infrastructure.

3. Architecture

Component	Role
Amazon EC2	Test instance whose stopped state is monitored.
Amazon EventBridge	Receives the EC2 state-change event and matches the configured event pattern.
Systems Manager Automation	Runs the AWS-StartEC2Instance automation document.
AWS-StartEC2Instance	AWS-managed runbook used to start the selected EC2 instance.
IAM Role	Provides the permissions required by the automation.

4. End-to-End Workflow

1. The EC2 test instance is initially in the Running state.
2. The instance is stopped for testing.
3. EC2 generates an Instance State-change Notification.
4. Amazon EventBridge receives the event.
5. The EventBridge rule checks whether the event matches the configured pattern:
source = aws.ec2
detail-type = EC2 Instance State-change Notification
state = stopped
instance-id = the selected EC2 instance
6. When the pattern matches, EventBridge invokes Systems Manager Automation.
7. Systems Manager executes AWS-StartEC2Instance.
8. The automation uses the configured IAM role.
9. The EC2 instance changes through Pending and returns to Running.
10. The execution can be verified in Systems Manager Automation execution history.

5. Amazon EC2

Amazon Elastic Compute Cloud (EC2) provides resizable compute capacity in AWS. For this project, one EC2 instance is used as the test resource. The important state transition is from **running** to **stopped**, followed by the automated recovery back to **running**.

The test instance should be running before the automation is tested. Its Instance ID is used in the EventBridge event pattern and in the Systems Manager Automation target.

6. Amazon EventBridge

Amazon EventBridge is an event-driven service that can receive events and route matching events to configured targets. In this project, EventBridge acts as the detection and routing layer between EC2 and Systems Manager Automation.

The rule is named **AutoStartStoppedEC2** and uses the default event bus. Its event pattern matches EC2 state-change notifications for the selected instance when its state becomes **stopped**.

7. Event Pattern

```
{
  "source": ["aws.ec2"],
  "detail-type": ["EC2 Instance State-change Notification"],
  "detail": {
    "state": ["stopped"],
    "instance-id": ["i-xxxxxxxxxxxxxxxxxx"]
  }
}
```

The example Instance ID must be replaced with the actual EC2 Instance ID used in the lab. The rule therefore focuses the automation on one specific test instance.

8. AWS Systems Manager Automation

AWS Systems Manager Automation is used to perform automated operational actions on AWS resources. In this project, the AWS-managed runbook **AWS-StartEC2Instance** is selected as the target of the EventBridge rule.

The main parameter is **InstanceId**, which identifies the EC2 instance that must be started. The automation also uses an **AutomationAssumeRole** value containing the ARN of the IAM role created for this workflow.

9. IAM Configuration

IAM controls permissions for AWS API operations. The automation role created for this project is named **EC2AutoStartAutomationRole**.

IAM Element	Configuration
Role	EC2AutoStartAutomationRole
AWS-managed policy	AmazonSSMAutomationRole
Inline policy	AllowStartEC2Instance
EC2 permission	ec2:StartInstances
EC2 permission	ec2:DescribeInstances

The inline policy grants the automation the ability to start EC2 instances and describe their state. The role ARN follows the general form:

```
arn:aws:iam::ACCOUNT-ID:role/EC2AutoStartAutomationRole
```

10. Self-Healing Concept

Traditional monitoring may only detect a problem and send an alert to an administrator. A self-healing workflow adds an automated corrective action for known conditions.

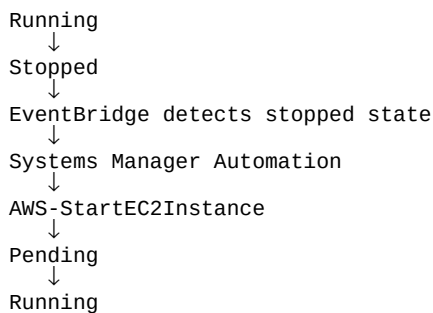
Monitoring detects something is wrong → Automation fixes it.

In this project, the undesired condition is that the selected EC2 instance has entered the stopped state. EventBridge detects the condition and Systems Manager performs the corrective action of starting the instance.

11. Verification

- Stop the test EC2 instance.
- Wait until the instance reaches the Stopped state.
- Wait for EventBridge to detect the state-change event.
- Verify that Systems Manager Automation starts the instance.
- Confirm the instance transitions through Pending and reaches Running.
- Open Systems Manager → Automation → Execution history and verify that AWS-StartEC2Instance completed successfully.

12. Expected Result



A successful test demonstrates that the infrastructure can automatically respond to the configured EC2 state-change event.

13. Security and Operational Considerations

The automation should be granted only the permissions required for its intended task. The project role is limited to EC2 start and describe operations through the shown inline policy.

The automation should also be designed carefully in production. An EC2 instance can be stopped intentionally for maintenance, troubleshooting, or cost saving. Automatically restarting every stopped instance could therefore create unwanted behavior or additional cost.

14. Production Improvement

A recommended improvement is to apply automatic recovery only to instances that are explicitly marked for it. For example, an instance can use the tag:

```
AutoRecovery = true
```

The EventBridge logic can then be improved to target only the appropriate production instances rather than automatically restarting every stopped instance.

15. Real-World Applications

- Automatic recovery from known infrastructure failures.

- Event-driven operational remediation.
- Automatic response to infrastructure state changes.
- Reducing manual intervention for repetitive operational tasks.
- Building resilient cloud infrastructure workflows.

16. Key Learning Outcomes

- Understanding event-driven AWS architecture.
- Understanding EC2 state-change notifications.
- Creating EventBridge rules using event patterns.
- Using Systems Manager Automation runbooks.
- Understanding IAM roles and permissions for automation.
- Understanding the basic principles of self-healing infrastructure.

17. Project Summary

This project demonstrates a simple self-healing architecture using Amazon EC2, Amazon EventBridge, AWS Systems Manager Automation, and IAM. When the selected EC2 instance is stopped, EventBridge detects the state-change notification and invokes AWS-StartEC2Instance through the configured automation role. The instance is consequently returned to the running state.

18. Documentation Structure

The project repository separates conceptual theory, commands, and architecture documentation:

File / Folder	Purpose
README.md	Project overview, architecture, workflow, configuration, verification, and result.
Commands.md	AWS CLI commands used for testing and verification.
Theory.pdf	Conceptual explanation of self-healing infrastructure and AWS services.
Architecture/	Architecture diagram for the project.
Images/	Supporting project screenshots or images.

Note: This theory document follows the supplied Project 03 implementation guide. The detailed command-based testing and verification steps are maintained separately in Commands.md.