**Compliance Automation Framework**

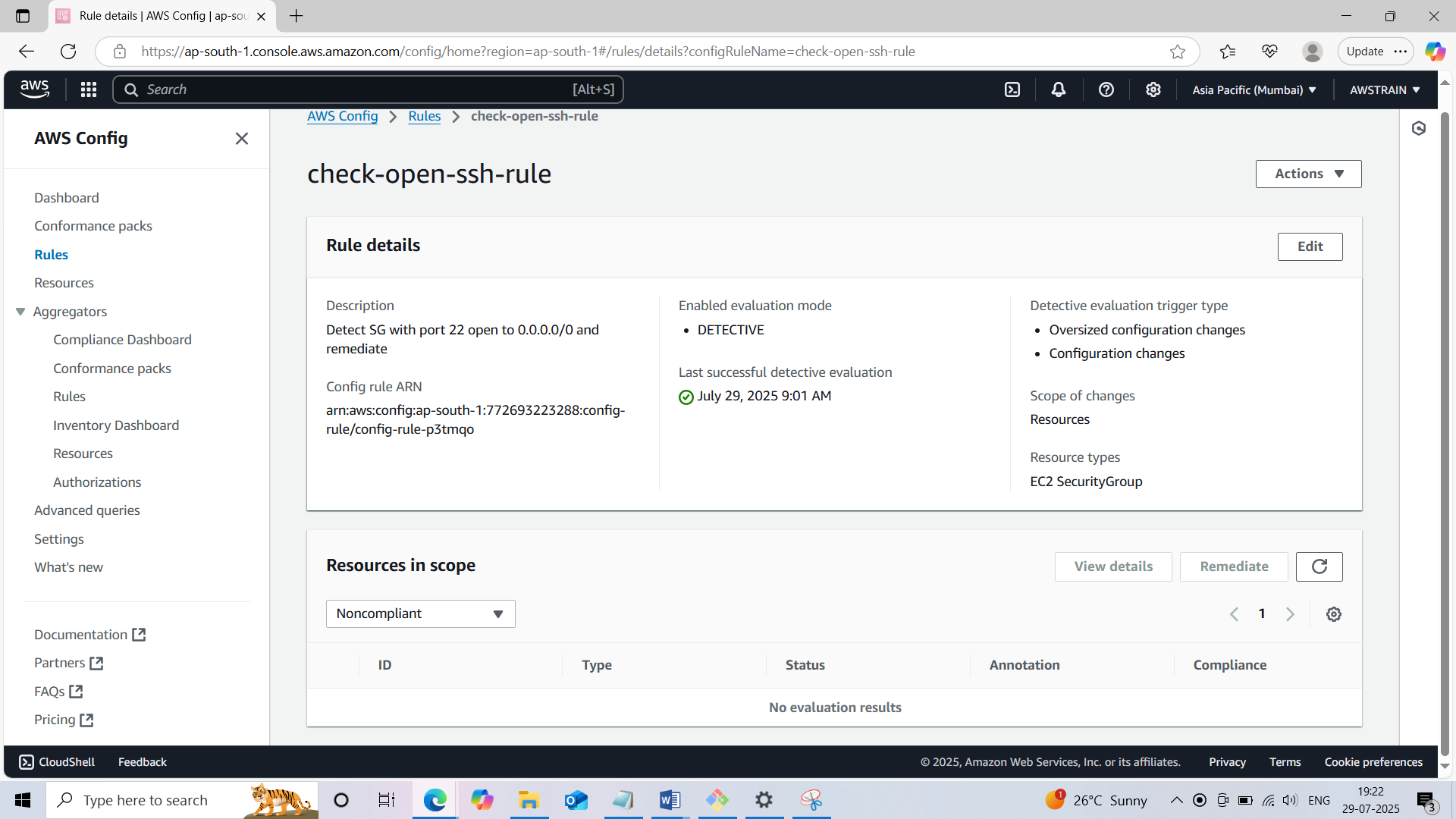
**Objective:** Build a Compliance Automation Framework using AWS Config and Systems Manager (SSM) that not only detects policy violations but also remediates them automatically using SSM automation runbooks—securely, reliably, and with traceability.

**Services used:**

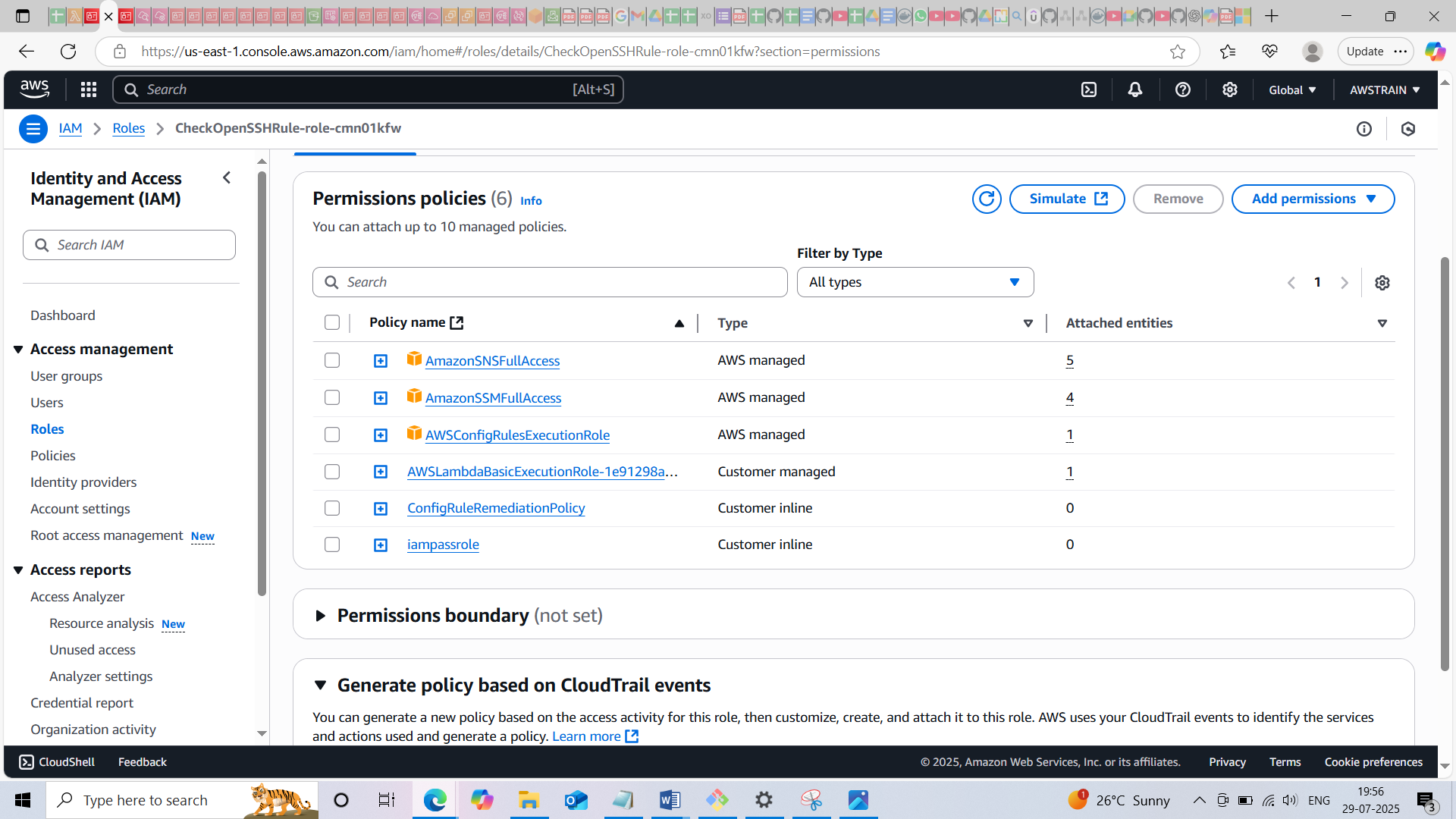
* **AWS Congif :** AWS Config is a **continuous monitoring and compliance service** that records the configuration of AWS resources, tracks configuration changes over time, and evaluates them against **custom or managed compliance rules.**
* **AWS Systems Manager(SSM) :** AWS Systems Manager (SSM) is a **management and automation service** that provides operational visibility and control of AWS and on-premises infrastructure. It helps **automate tasks** such as patching, configuration enforcement, software deployment, and incident remediation using **Run Command, Automation documents (SSM Documents), and State Manager**.
* **AWS Lambda :** AWS Lambda is **a serverless compute service** that automatically runs code in response to events **without provisioning or managing servers**. It scales automatically and you **pay only for execution time.**
* **AWS S3 :** Amazon S3 is an **object storage service** that **offers scalable, highly durable, and secure** storage for any amount of data. It is designed for **durability** and supports features like versioning, lifecycle management, and cross-region replication.
* **AWS Cloudtrail :** AWS CloudTrail is a **governance, compliance, and auditing service** that **records account activity** by capturing API calls and user actions across AWS services. It provides **event history for security analysis and compliance auditing.**
* **Evenbridge :** Amazon EventBridge is a **serverless event bus service** that allows applications and AWS services to **communicate via events**. It can **ingest events from AWS services, SaaS applications, or custom applications** and route them to targets like Lambda, SQS, or Step Functions.

**Scenerio 1: Open Security Group (Port 22 exposed to 0.0.0.0/0)**

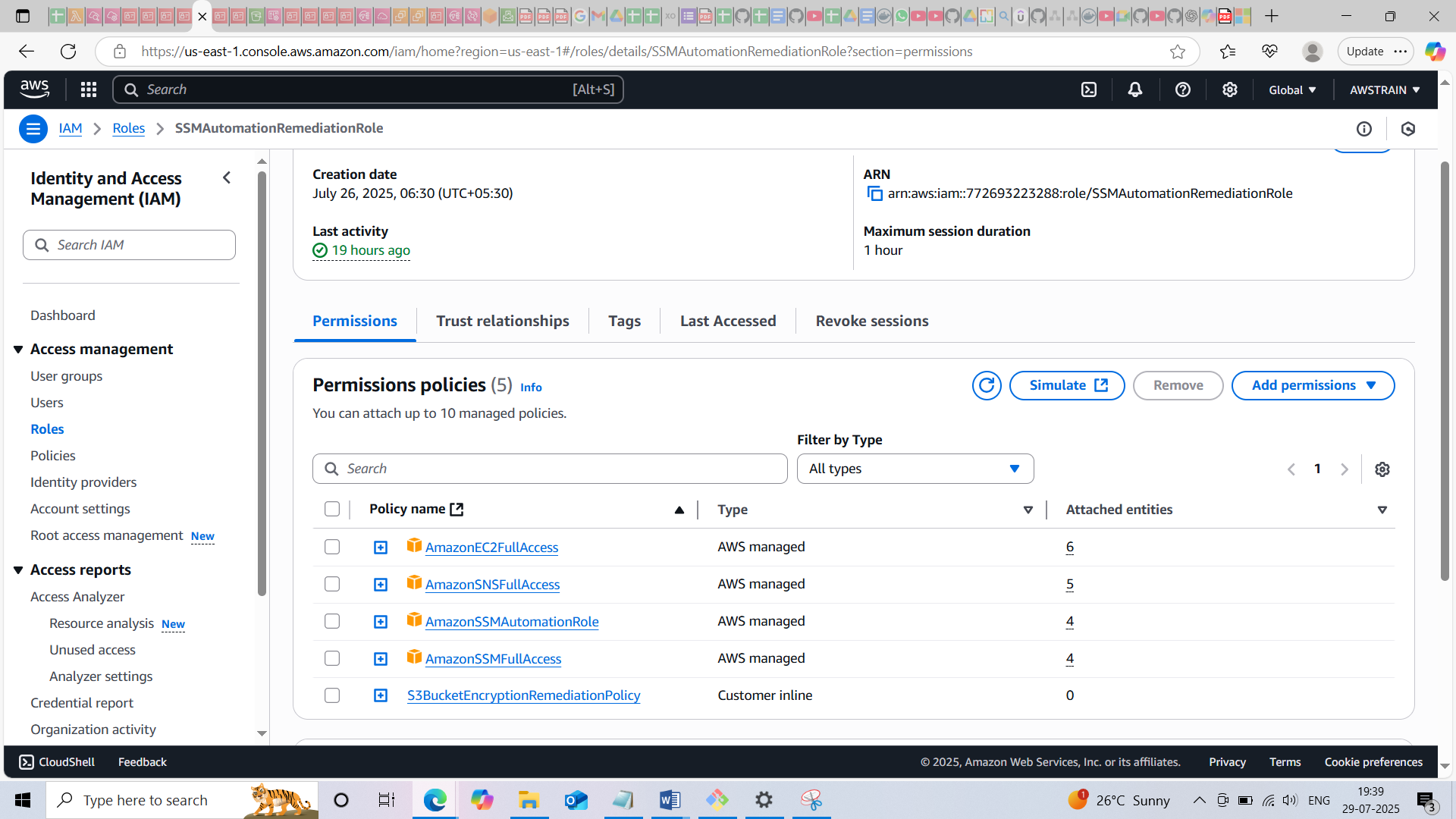
* Created rule named ‘check-open-ssh’ in AWS config
* Type: Custom lambda
* Created lambda function which terminates the ports with port 22 exposed to 0.0.0.0/0
* Trigger type: Configuration changes
* Provided the lambda arn in AWS config
* Provided EC2 Secuirty group as Resources
* Provide SSM Document name in the ‘Manage remediation’ and and security group id as Resource Id also the SSMAutomationRemediationRole ARN

****

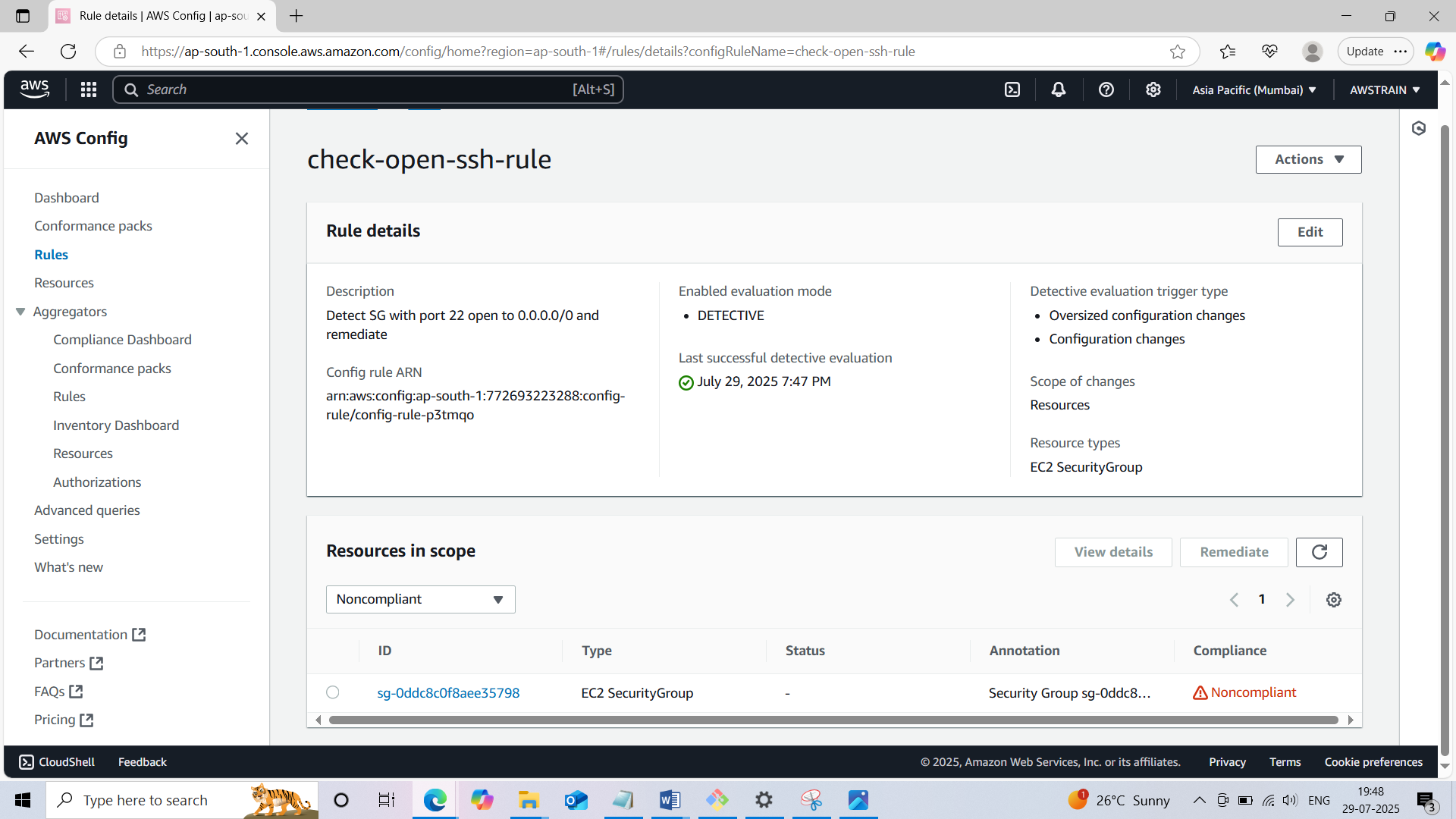
* **In this scenario we are creating a lambda function which triggers the SSM where it terminates the port 22 exposed to the 0.0.0.0/0 in the security group and makes it complaint from non complaint. Provided required permissions for lambda IAM role**

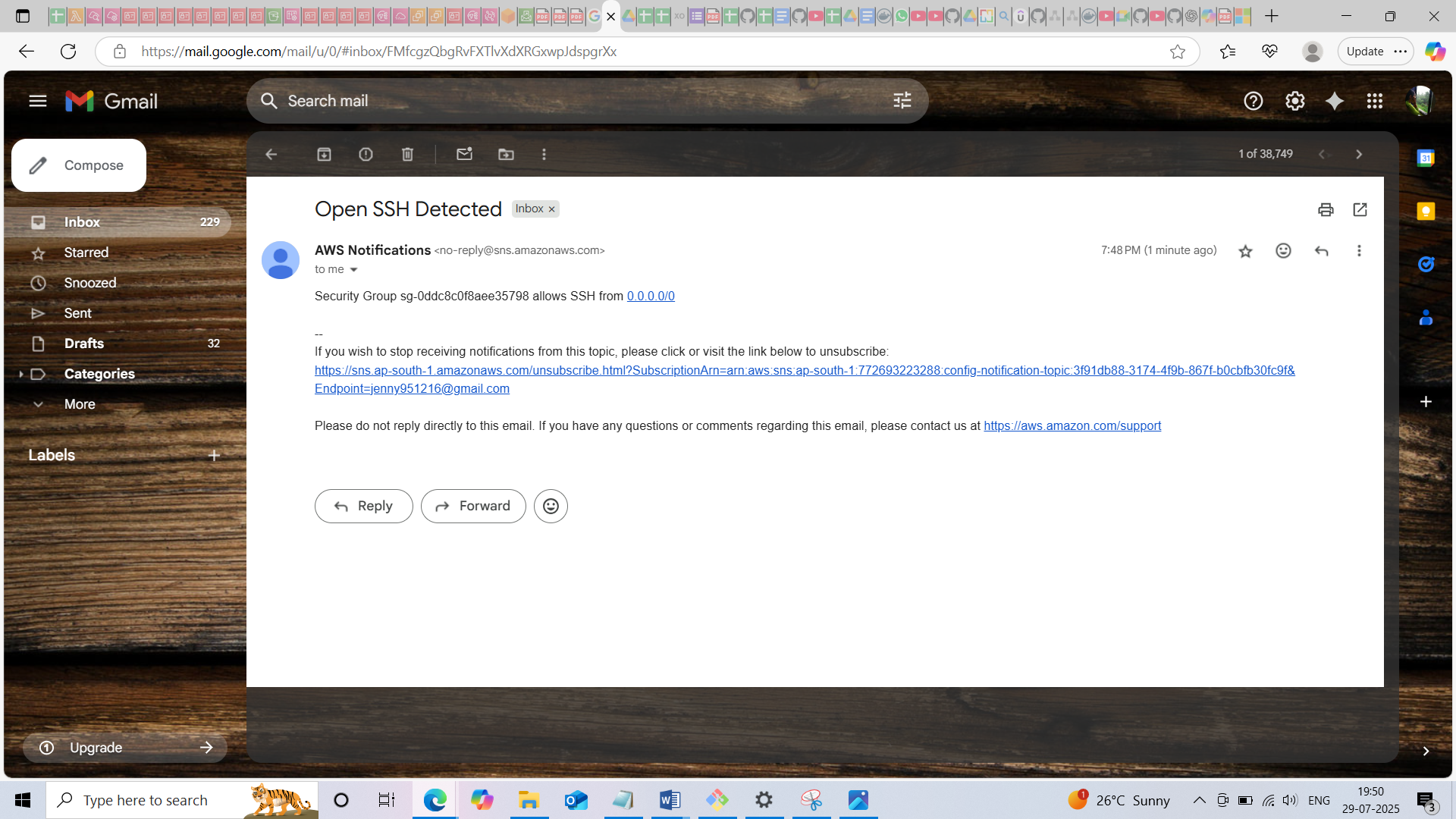
****

* **In the lambda function we will provide the SNS topic ARN where we received Email whenever a port with 22 opened to internet. Also provided the ARN of SSMAutomationRemediationRole ,a IAM role with required permissions**

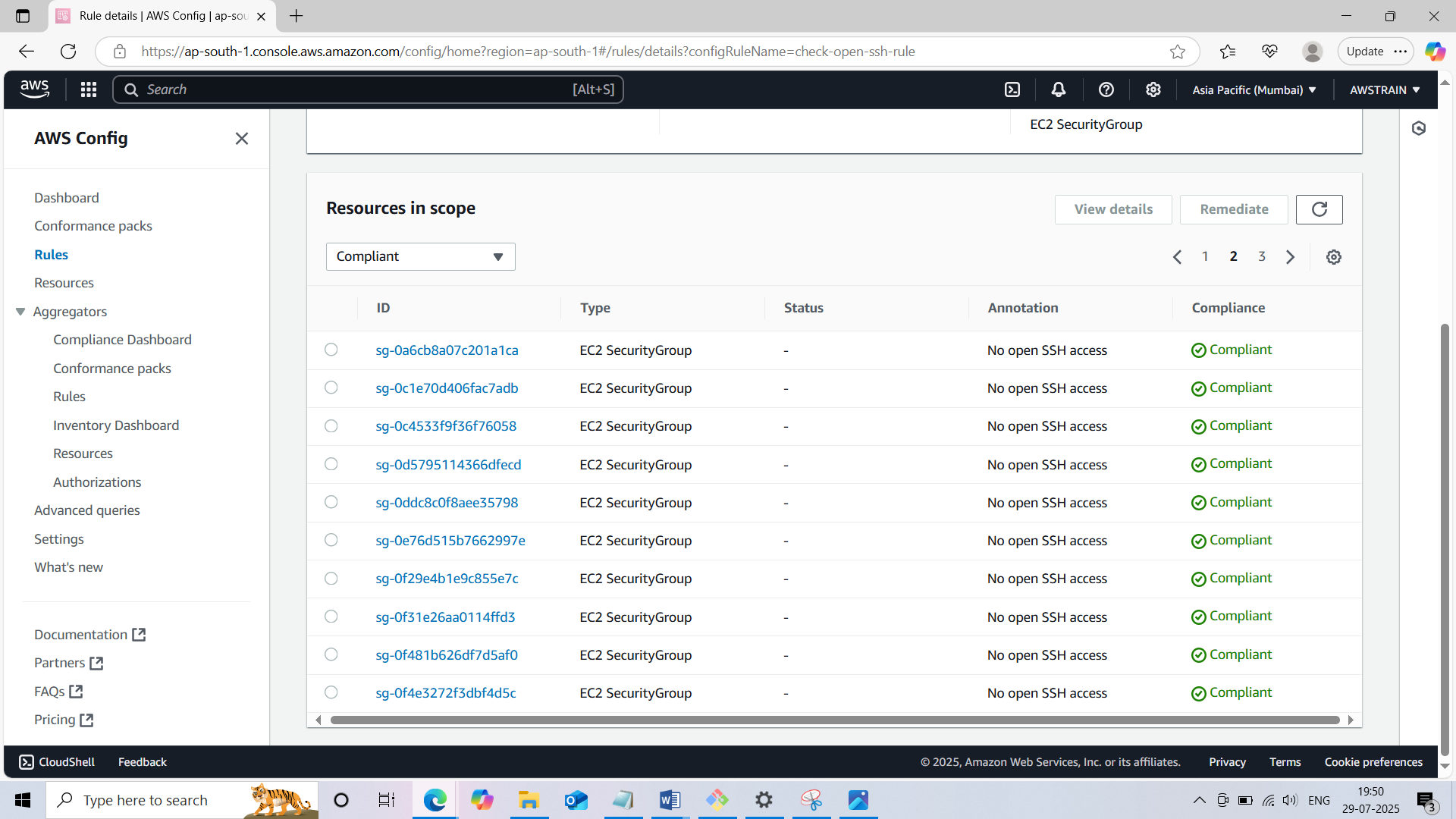
****

* **Once we created security group with port 22 exposed to internet will get an email via SNS also the respective security group id with status non compliant in the AWS config**

****

****

* **After few seconds according to the SSM document the port with range 22 will get deleted and security goup id changes to complaint status in AWS Config**

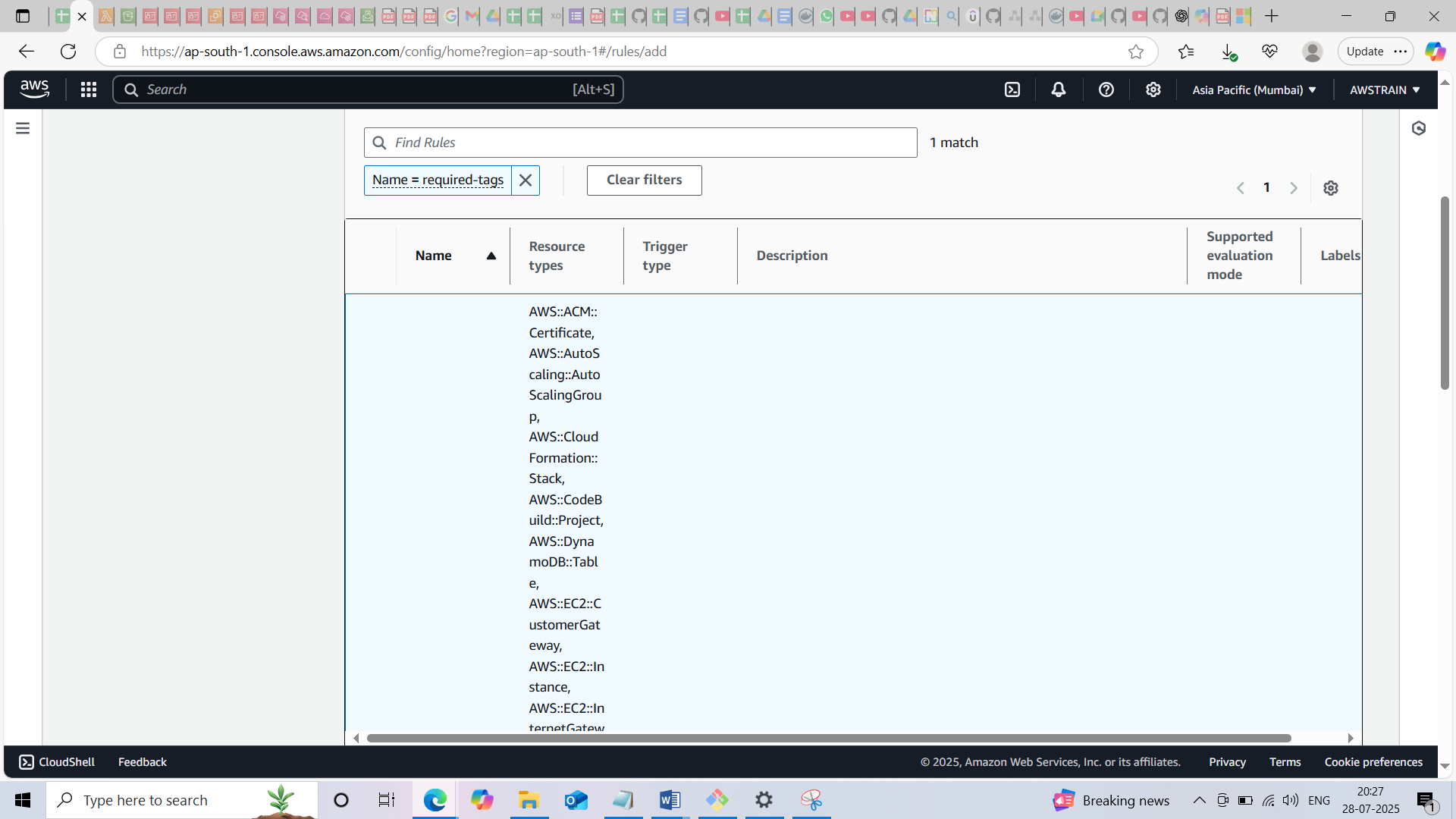
****

**AWS Config detect the wide open rule and email received via SNS also SSM revokes the**

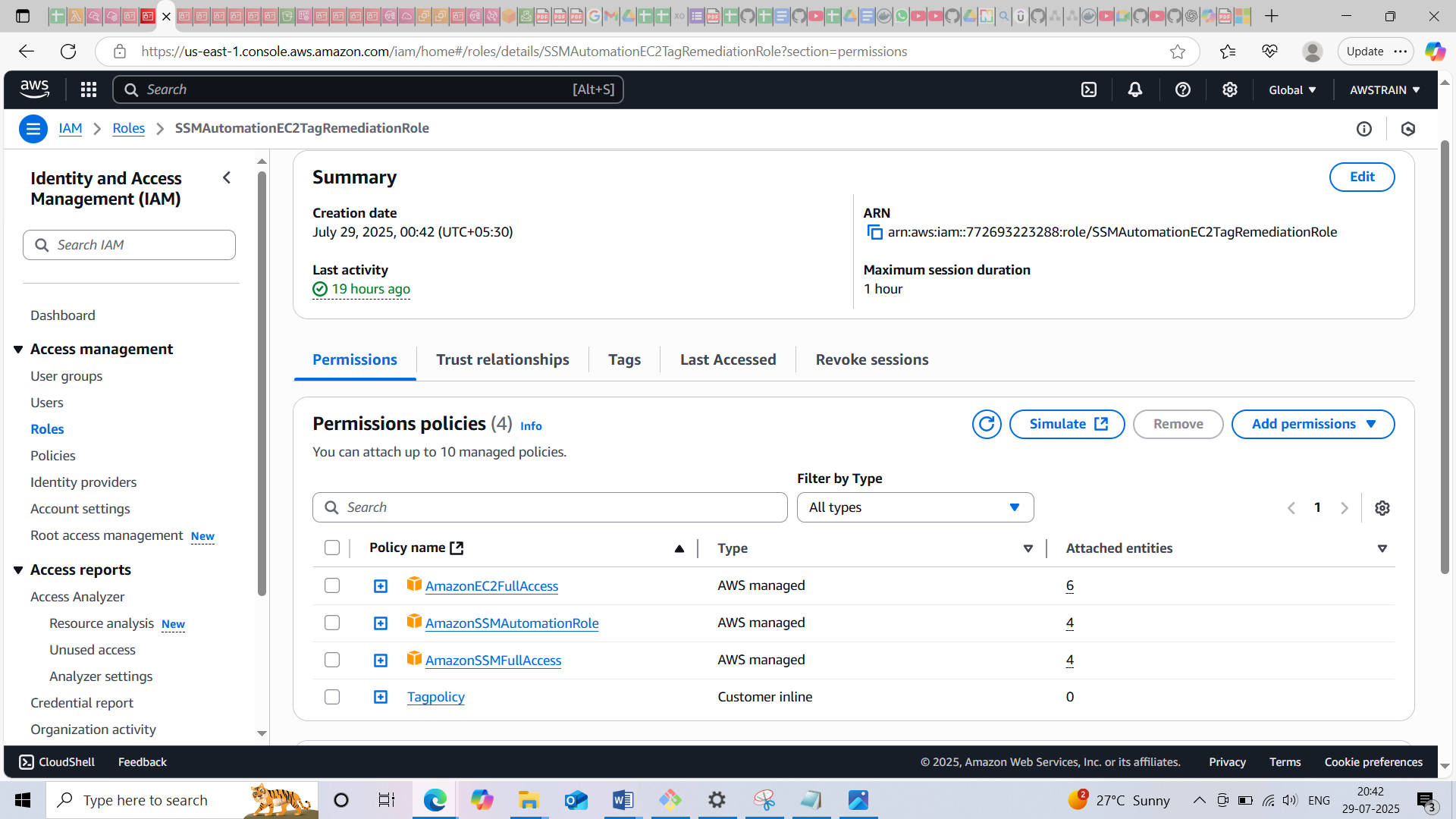
**Ingress rule successfully!**

**Scenario 2: Missing Required Tags on EC2**

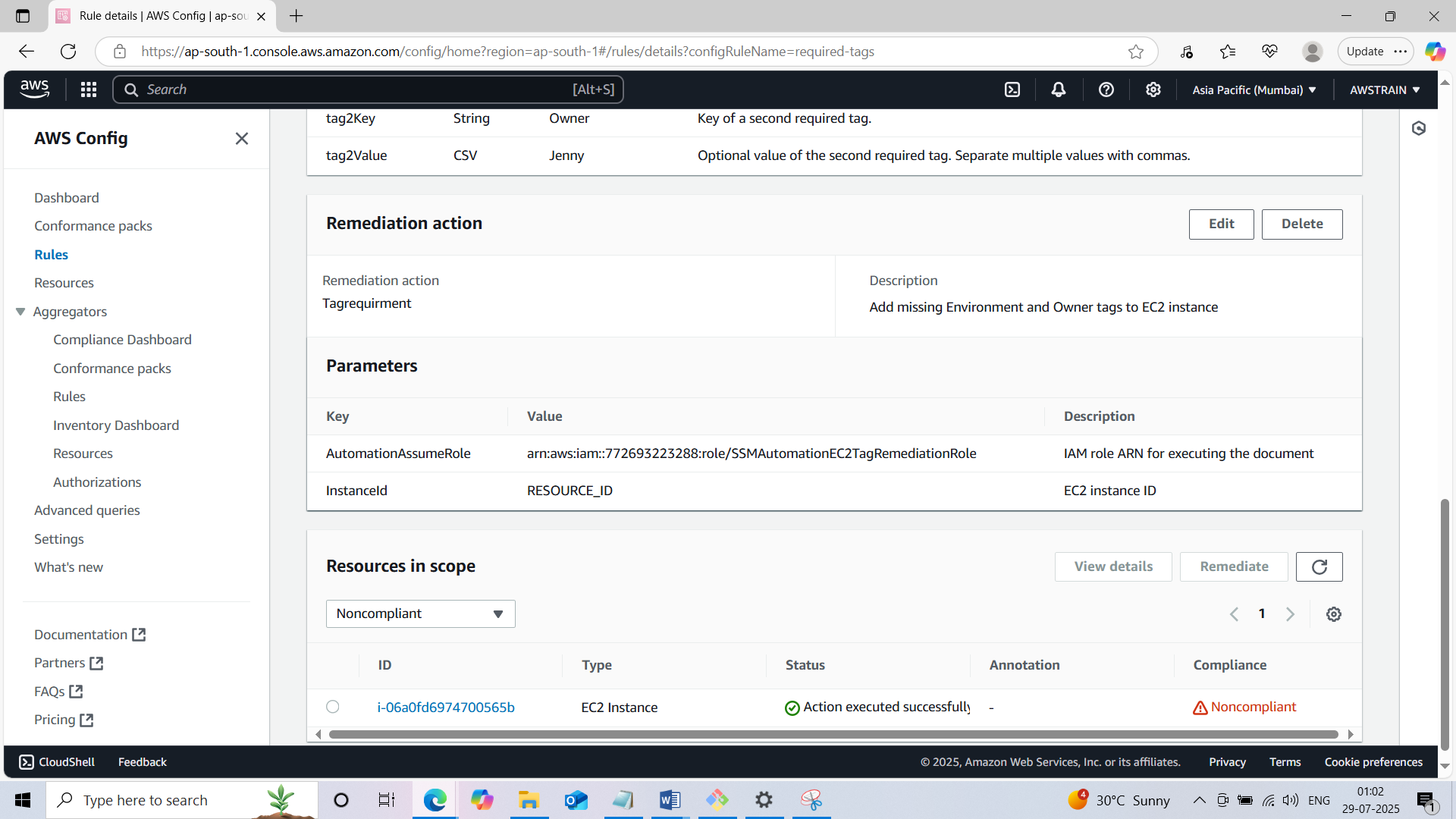
* Created AWS Config rule with AWS Managed type (search for required-tag)
* Trigger type: Configuration changes
* Provided EC2 instance as Resources

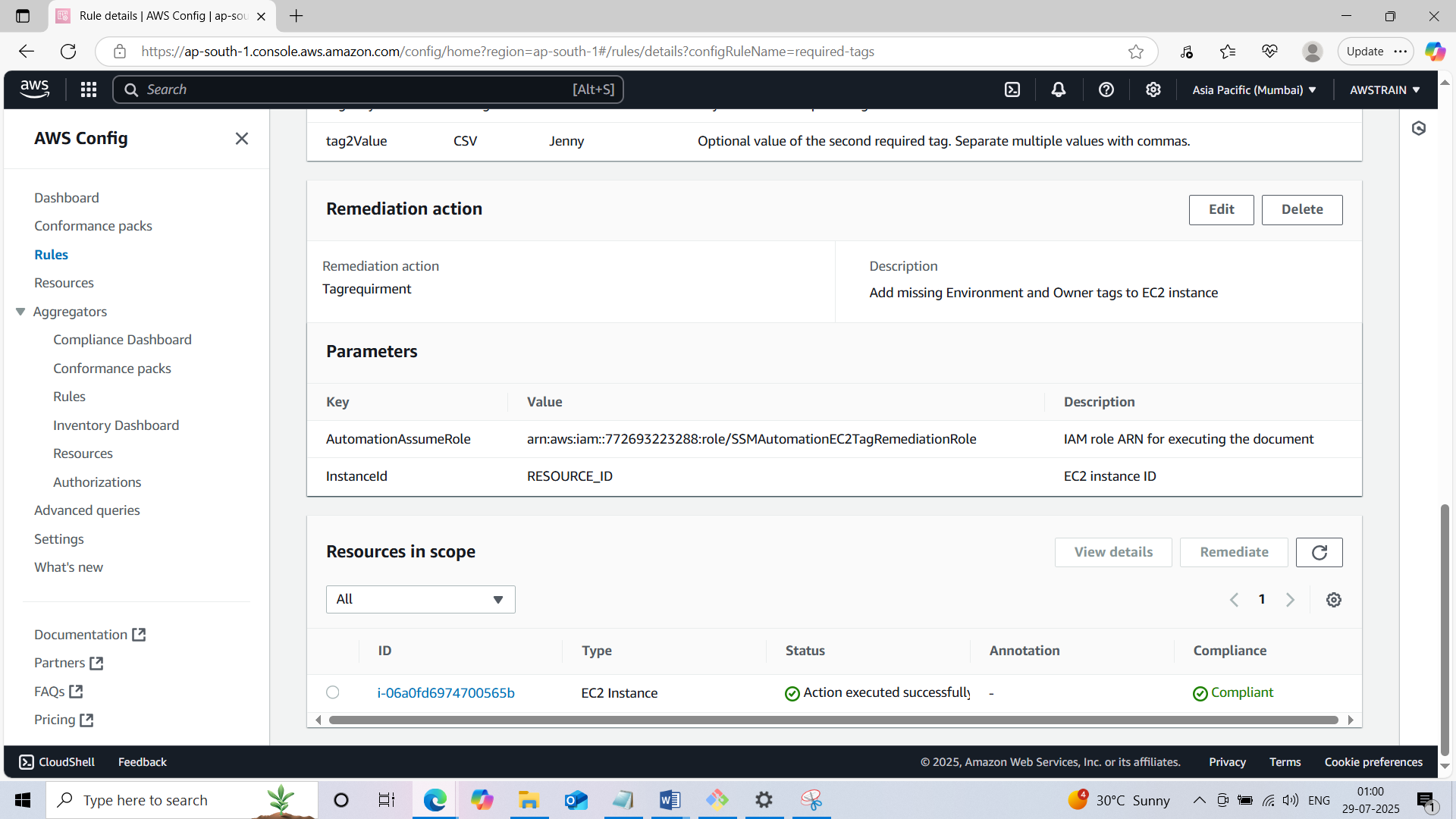
****

* Provide SSM Document name in the ‘Manage remediation’ and and EC2 instance id as Resource Id also the SSMAutomationEC2TagRemediationRole ARN



* In this scenario AWS Config detects the ec2 instances without the tags and SSM create the tags Keys Environment or Owner which is hardcoded in the document.

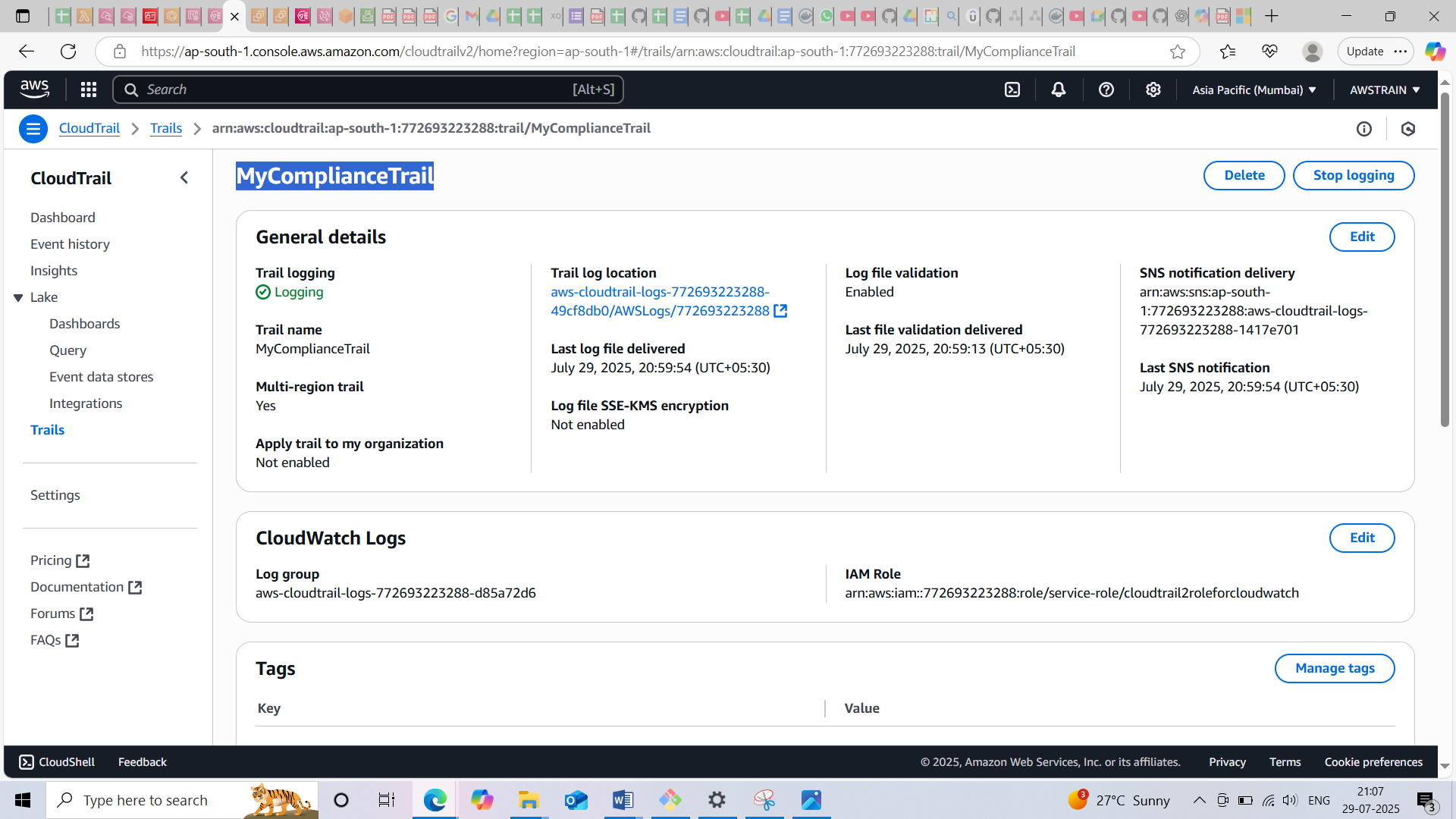




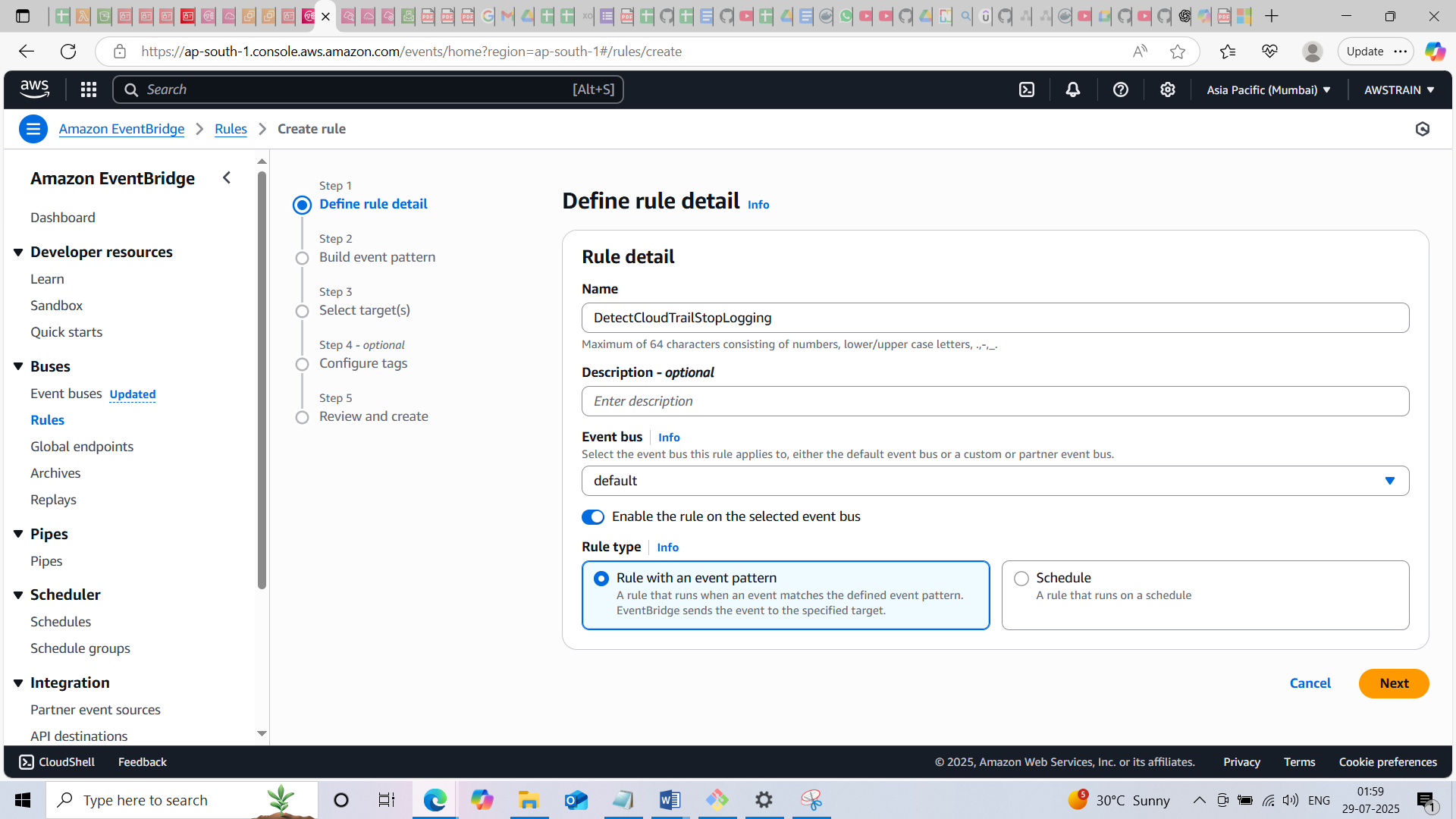
**Successfully detected the ec2 instance with the missing tag and created tag via SSM Document!**

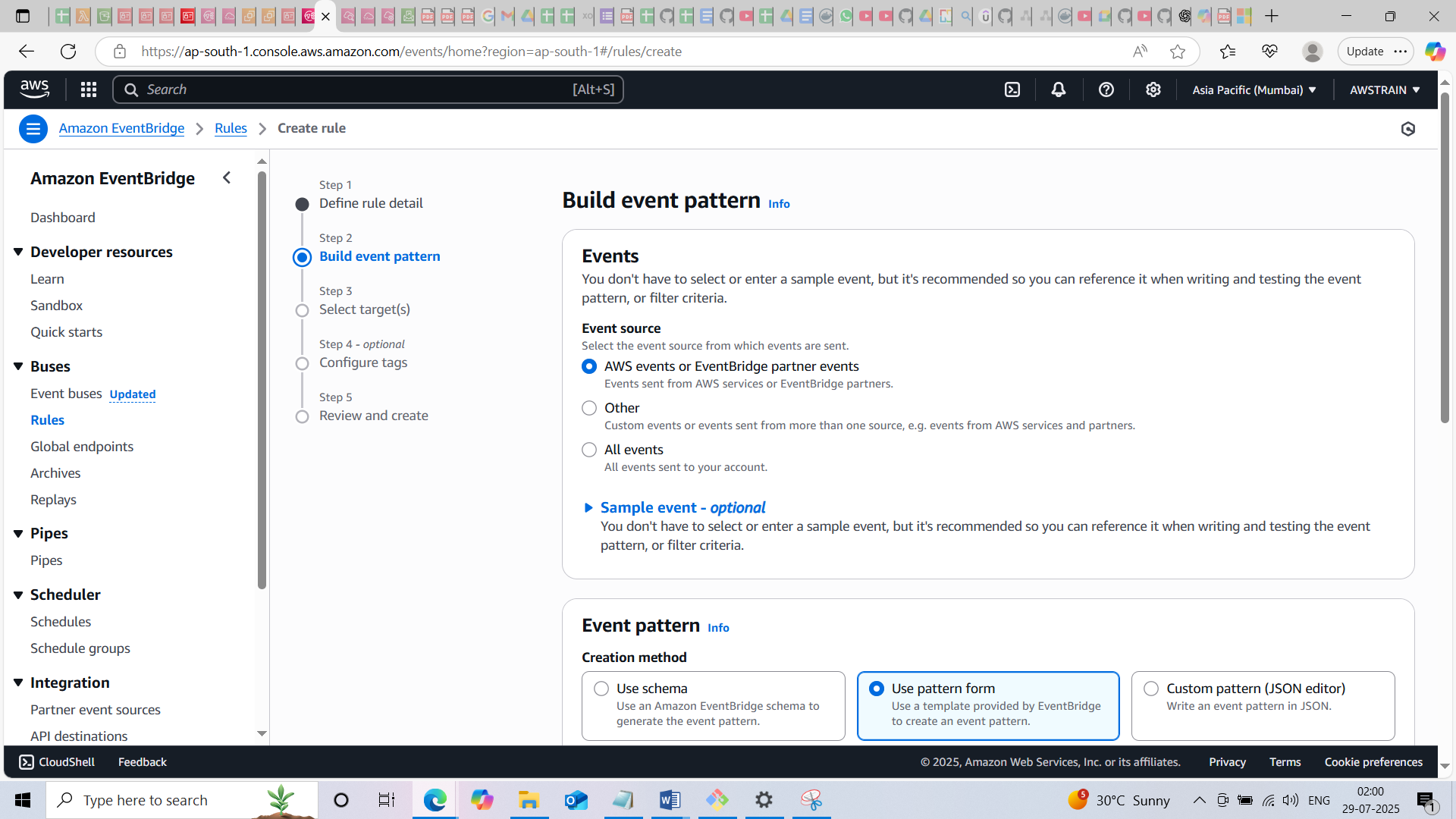
**Scenerio 4: Disabled Cloudtrail**

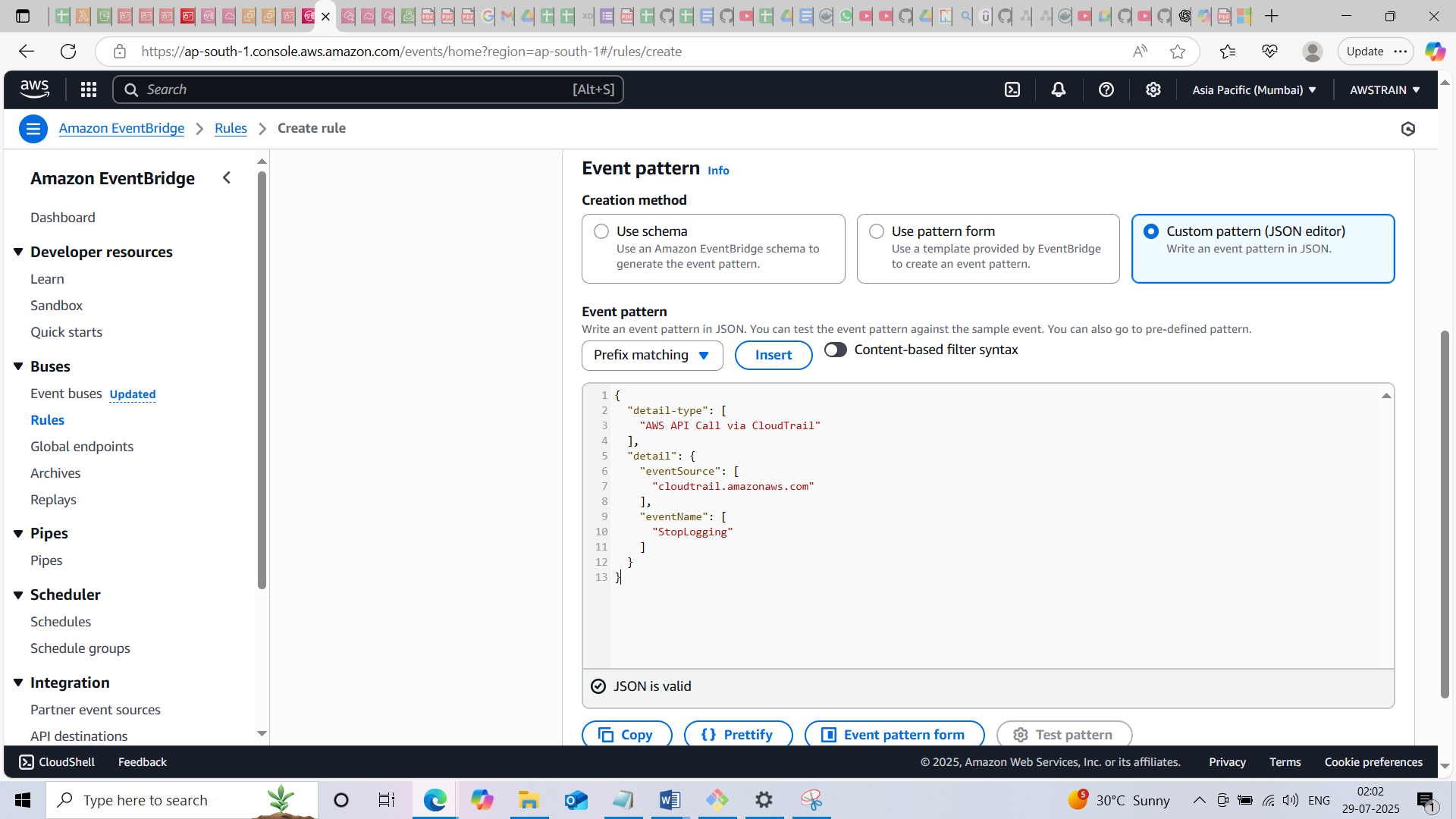
* **Created a cloudtrail named ‘MyComplianceTrail’**

****

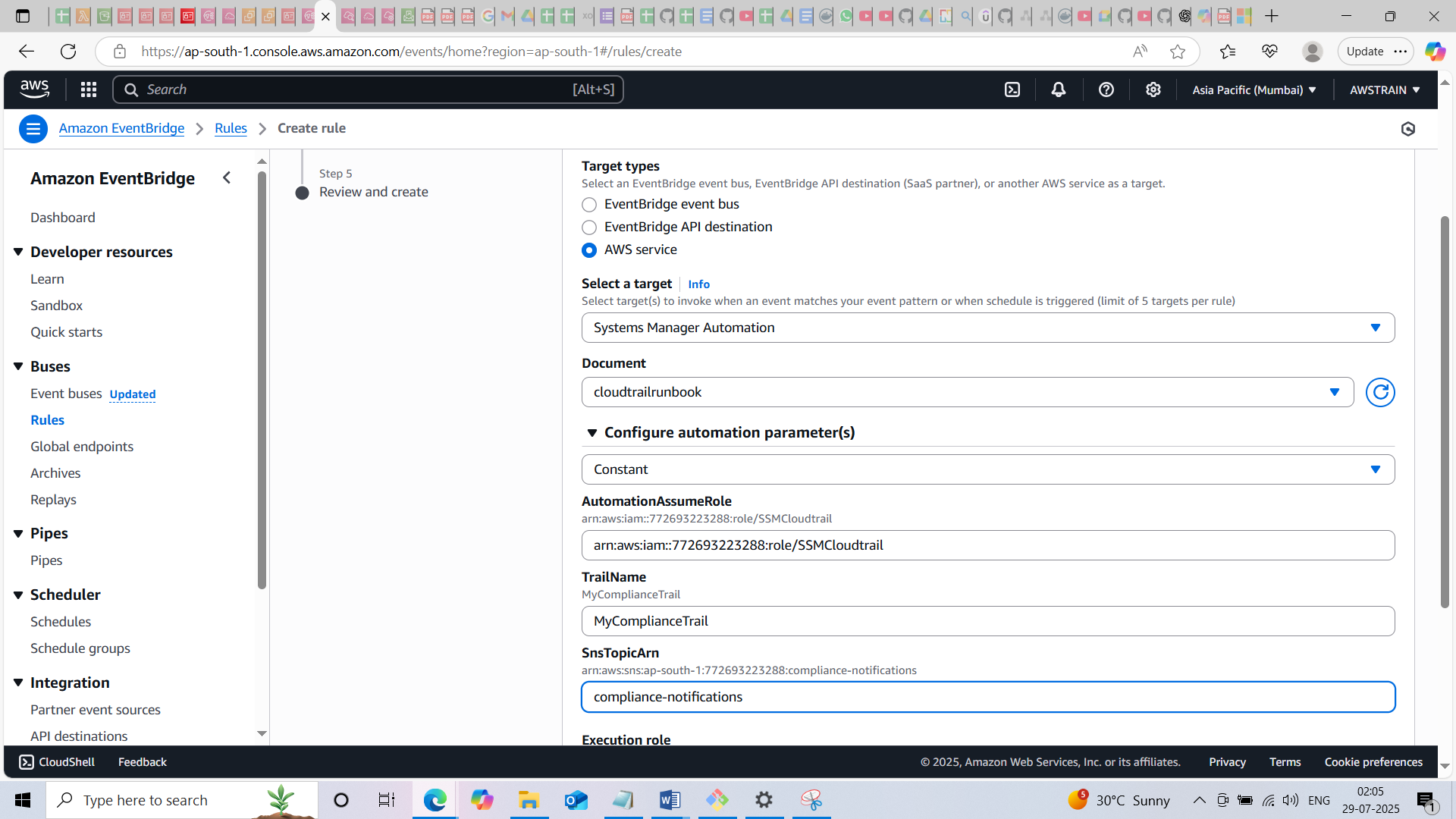
* **Created Event bridge rule named ‘cloudtrail-logging-disabled-rule’**
* **Provide a custom event pattern for call API when cloudtrail stops**

****

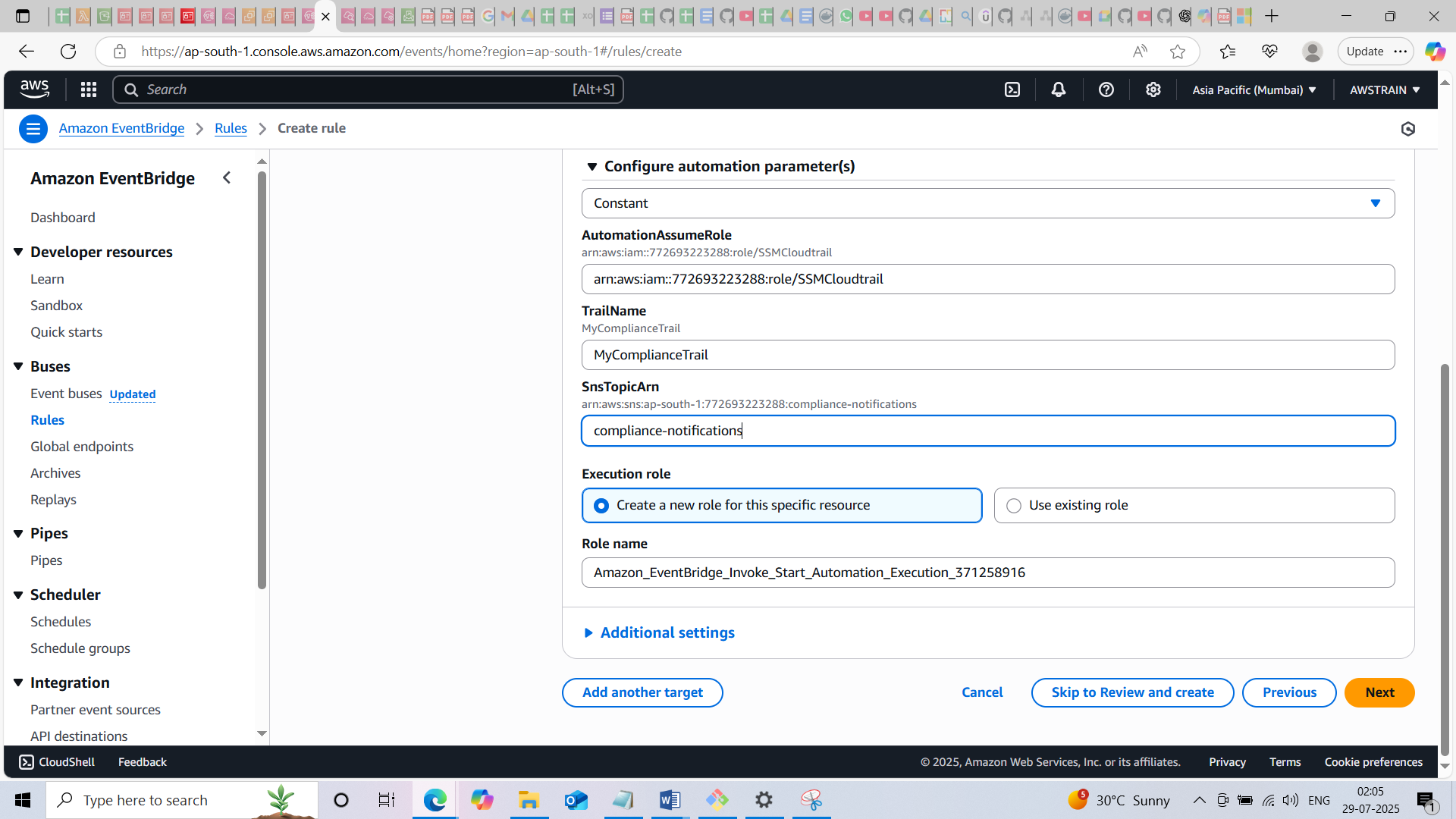
****

****

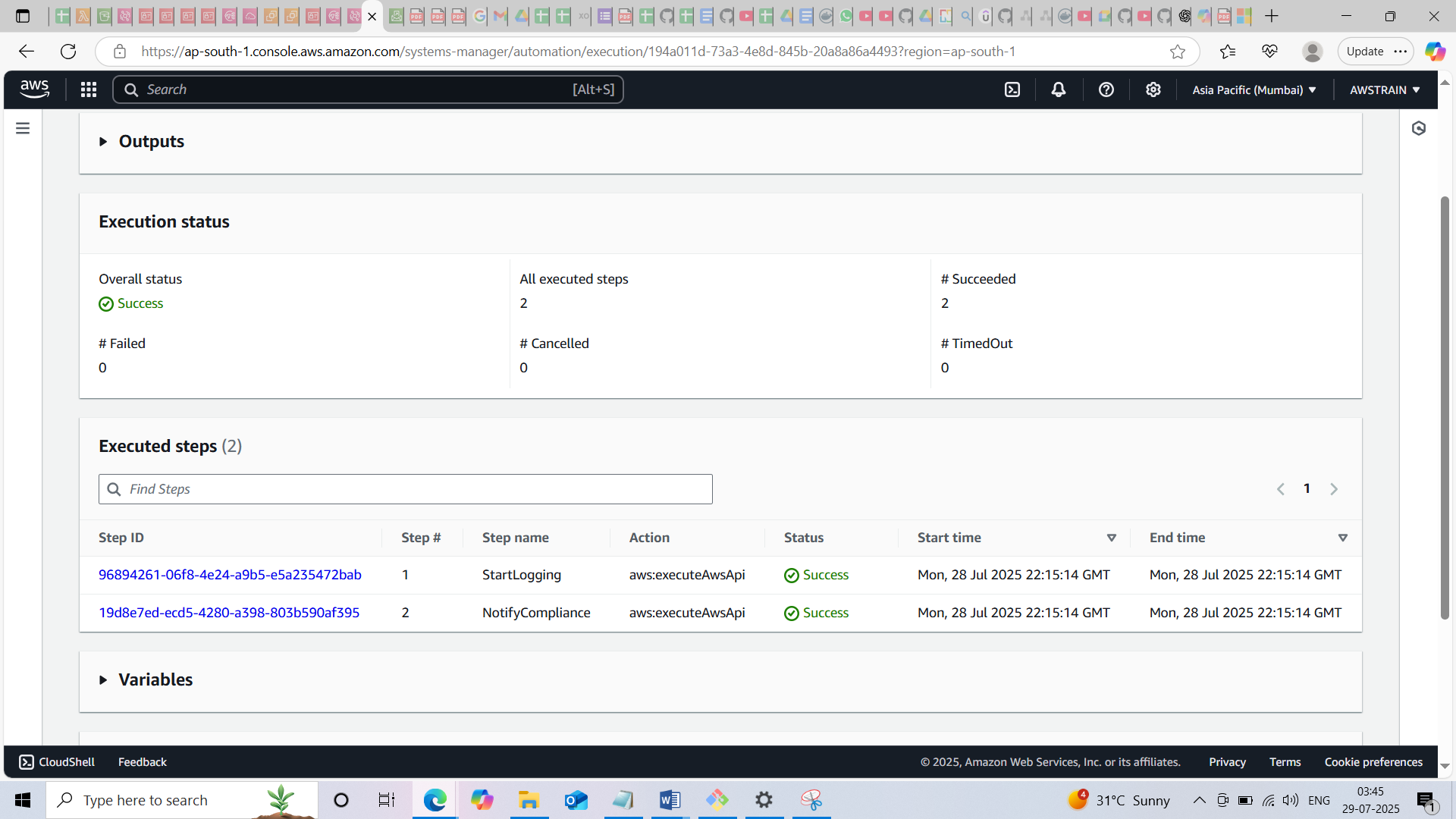
* **Provide the SSM Document created in the targets**

****

* **Provide the SNS ARN to receive email**



* **In this scenario while cloudtrail logged off event bridge detects it and SSM re-enables it**

****

* When I manually logged logged off the cloudtrail , a few seconds later the cloudtrail will automatically logged on and received the mail too!