PART I: Concepts and Theory, Algorithms

1. Consider design for your eCommerce SaaS on AWS Cloud. In this context:

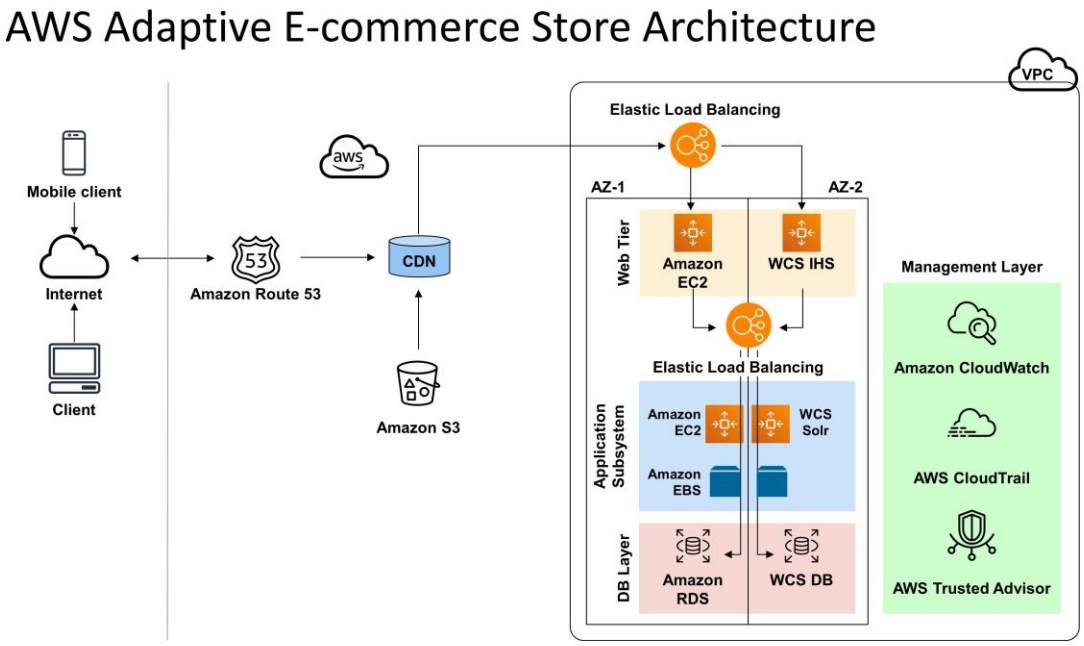
a. Explain the key functionalities of Apache Kafka. What is it and why would you need it for your eCom SaaS platform?

Key functionalities:
Publish and subscribe to streams of records
Effectively store streams of records in the order in which records were generated
Process streams of records in real time

What is Apache Kafka?
Apache Kafka is a distributed data store optimized for ingesting and processing streaming data in real-time. Streaming data is data that is continuously generated by thousands of data sources, which typically send the data records in simultaneously. A streaming platform needs to handle this constant influx of data, and process the data sequentially and incrementally.

why would I need it for my eCom SaaS platform?
For the eCom SaaS platform, Kafka can be used to build real-time streaming data pipelines and real-time streaming applications. A data pipeline reliably processes and moves data from one system to another, and a streaming application is an application that consumes streams of data. For example, if I want to create a data pipeline that takes in user activity data to track how people use the website in real-time, Kafka would be used to ingest and store streaming data while serving reads for the applications powering the data pipeline. Kafka is also often used as a message broker solution, which is a platform that processes and mediates communication between two applications.

b. Please show your design using an architectural diagram.



c. What is Amazon Kinesis? How is it similar to and different from Kafka? How would you need it for your eCom SaaS platform?

What is Amazon Kinesis?

Amazon Kinesis is a managed, scalable, cloud-based service that allows real-time processing of streaming large amount of data per second. It is designed for real-time applications and allows developers to take in any amount of data from several sources, scaling up and down that can be run on EC2 instances.

It is used to capture, store, and process data from large, distributed streams such as event logs and social media feeds. After processing the data, Kinesis distributes it to multiple consumers simultaneously.

Similarity (Kafka):

Both Apache Kafka and AWS Kinesis Data Streams are used for real-time data streaming processing and analysis offered by AWS. The two services address almost the same tasks.

Difference (Kafka):

Data retention: There's a maximum 7-day retention period on Kinesis.

Set-up: Kafka takes longer to set up than Kinesis. You'll need a team to install (and manage) data clusters.

SDK support: Kafka supports Java; Kinesis (via AWS) supports Java, Go, Android, and .NET.

Price: Kafka is open-source and free. Kinesis has no set-up cost; users pay for resources used.

Reviews: Kafka has a higher customer review score than Kinesis on the website G2 (4.4/5 vs. 4.1/5).

How would you need it for your eCom SaaS platform?

For the eCom SaaS platform, Kinesis Data Streams can be used for rapid and continuous data intake and aggregation. The type of data used can include IT infrastructure log data, application logs, social media, market data feeds, and web clickstream data. Because the response time for the data intake and processing is in real time, the processing is typically lightweight.

Some typical scenarios: Accelerated log and data feed intake and processing; Real-time metrics and reporting; Real-time data analytics; Complex stream processing

d. What are message queues (MQ) and why do we need them for SaaS? Compare and contrast any 3 MQ solutions you learned about.

What are message queues (MQ)?

Message queuing allows applications to communicate by sending messages to each other. The message queue provides temporary message storage when the destination program is busy or not connected. A message queue provides an asynchronous communications protocol, which is a system that puts a message onto a message queue and does not require an immediate response to continuing processing. For example, Email is probably the best example of asynchronous communication. When an email is sent, the sender continues to process other things without needing an immediate response from the receiver. This way of handling messages decouples the producer from the consumer so that they do not need to interact with the message queue at the same time.

Why do we need them for SaaS?

In modern SaaS cloud architecture, applications are decoupled into smaller, independent building blocks that are easier to develop, deploy and maintain. Message queues provide communication and coordination for these distributed applications.

Message queues can significantly simplify coding of decoupled applications, while improving performance, reliability and scalability.

Compare and contrast any 3 MQ solutions:

1. RabbitMQ

Pros:

Self hosted; Cheap and fast; Outstanding client libraries for popular language; Supports multiple protocols (amqp, mqtt, stomp)

Cons:

AMQP is complex; RabbitMQ is written in erlang. The configurations are in erlang.

2. ActiveMQ

Pros:

Extensive broker capabilities link; Free; Fast; Supports multiple protocols (amqp, openwire, mqtt, jmx, stomp); Java client support and feature set is impeccable.

Cons:

Client library support for non java languages is mediocre; High availability is achieved via a failover mechanism, which is not supported out of the box by non-jms libraries; Requires reboots for configuration changes.

3. Redis

Pros:

Free; Fast; Outstanding client libraries for popular languages

Cons:

Advanced message queue features like atleast-once delivery need to be implemented in the clients; 3rd party Admin UIs need to be used.

2. The eCommerce web store you built is for Food and Wine only. Its is acquired by a store like Aamazon.com. In this context, please explain:

a. What is SSO? Why do you need it and how would you use it with the 2 platforms' merger?

What is SSO?

Single sign-on (SSO) is an authentication scheme that allows a user to log in with a single ID and password to any of several related, yet independent, software systems

Why do you need SSO?

With SSO, users could mitigate risk for access to 3rd-party sites because user passwords not stored or managed externally. Also, SSO provide better network security by Eliminating multiple passwords and reducing a common source of security breaches.

For administrative control, SSO could reduce IT costs due to lower number of IT help desk calls about passwords.

In conclusion, it provides better administrative control and it improves user productivity.

How would you use it with the 2 platforms' merger?

For 2 platforms' merger, I would implement Single Sign On between the two platform where a user logged in once and has access to all resources across all systems (As a result, can buy or browse both wine and food). For the technique level, I would implement single sign on between two applications using JASIG's CAS server which provides clients for communication in languages such as Java, Net, PHP. If the session fails between either system, the user will need to log in again. Or tweak session duration of both systems so they are in sync.

b. Explain what these are and why you need them for your eCommerce SaaS platform: SAML, Kerberos, Oauth

Explain what these are:
All of them are most commonly used authentication protocols for user authentication in applications.
-- Kerberos: Kerberos is a network authentication protocol. It is designed to provide strong authentication for client/server applications by using secret-key cryptography.
-- Oauth 2: OAuth 2 is an authorization framework that enables applications to obtain limited access to user accounts on an HTTP service, such as Facebook, GitHub, and DigitalOcean.
-- SAML: Security Assertion Markup Language (SAML) is an XML-based, open-standard data format for exchanging authentication and authorization data between parties, in particular, between an identity provider and a service provider. SAML is a product of the OASIS Security Services Technical Committee.

Why I need them for the eCommerce SaaS platform?
They are employed heavily on secure systems that depend on reliable auditing and authentication features. For an eCommerce SaaS platform, a password is used to prove a user's identity for the software. Obviously, it is necessary to prevent anyone from intercepting or eavesdropping on the transmitted password.

c. How is Amazon IAM different from the above?

IAM and SSO are related, they denote different things, and should not be conflated. As the name suggests, IAM is a general security tech and business discipline that encourages granting the right users the right access to the right resources or apps.
For the difference on how is IAM different from the above, SSO is a subset of a larger IAM system. A fully-featured IAM solution has features such as automated provisioning and de-provisioning features, secured authentication and identity governance – features that are lacking in SSO.

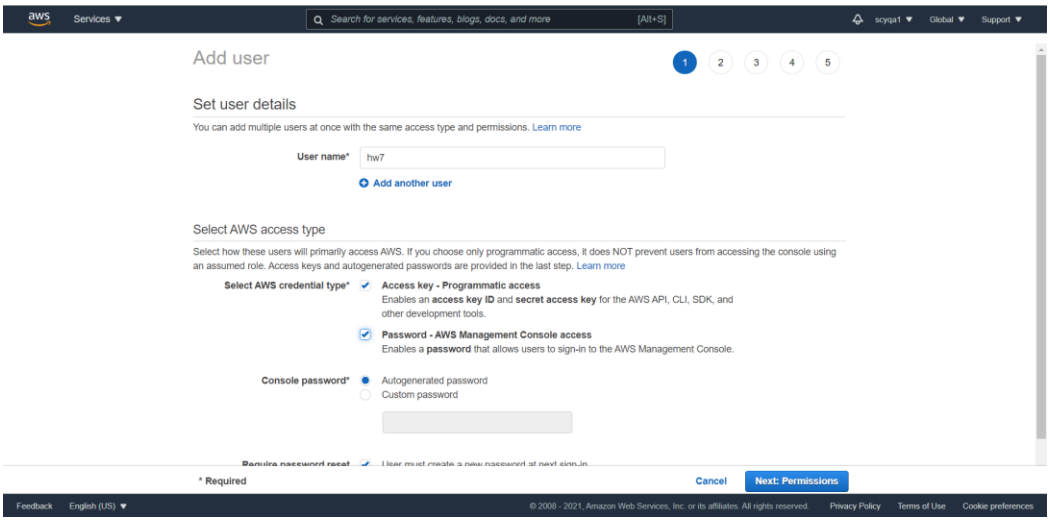
d. Which one would you use for your eCom store and why? Please justify your answer.

I would use SSO for the eCom store.
For the reason,
Firstly, I have more than one website (wine, food, Amazon) in the business architecture. In this case, I have to login to each website individually which is a tedious job. However, having one common password for each and every website is not a secure solution. Also, it is difficult to maintain and remember different passwords for all the websites.
The customers who access the same web store from different platforms will have to login every time. This is not a feasible solution for customers as it will irritate them. The same happens with merchants and their staff members.
To overcome these challenges, Single Sign-On (commonly referred to as SSO) helps in the automation of user logins, sessions and maintaining their assigned roles.

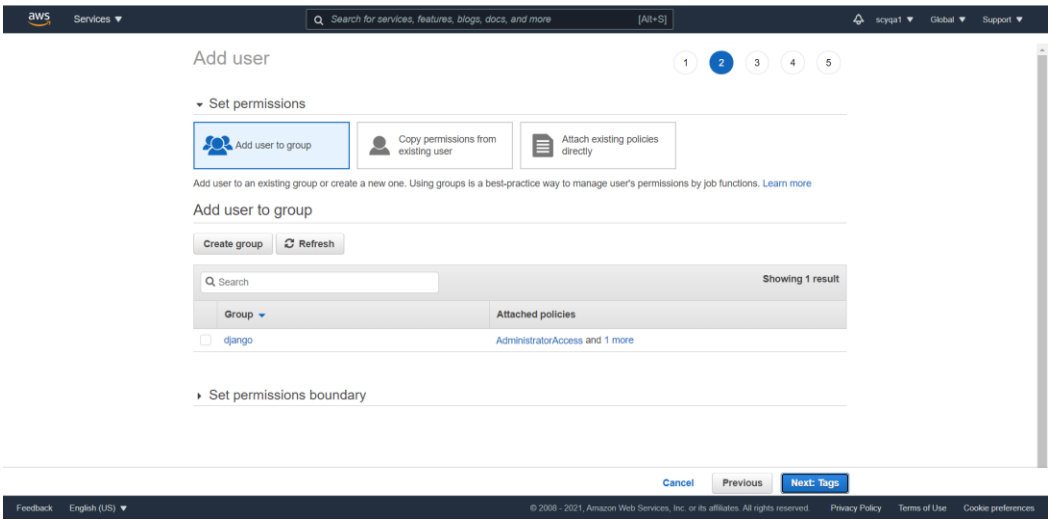
PART II: LAB [30 points]

1. Following the examples in Chapter 13, section on AWS IAM from your textbook, create a basic IAM user, Group and Roles and show screenshots and explain your results – why you need these for your eCommerce SaaS workloads.

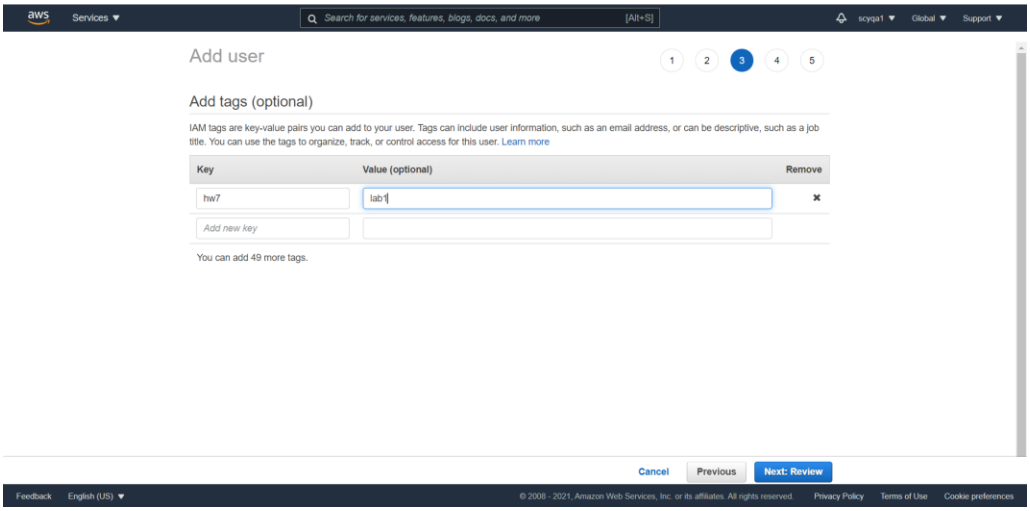
Create an IAM user:



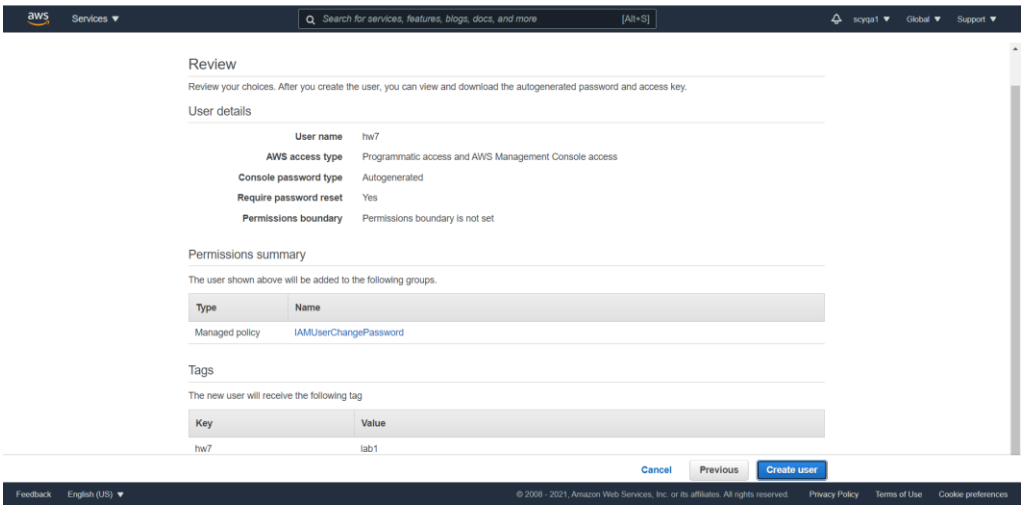
Add user to group



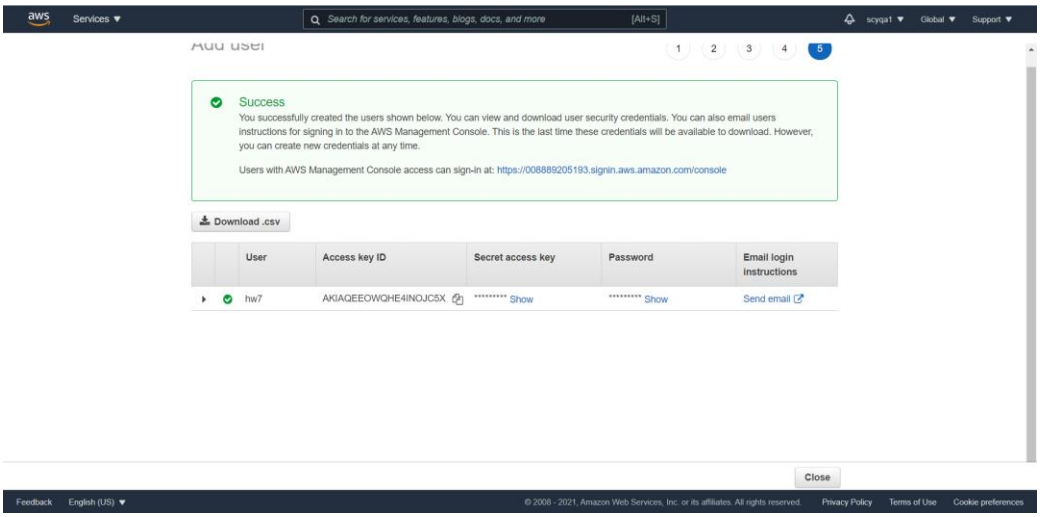
Add tags



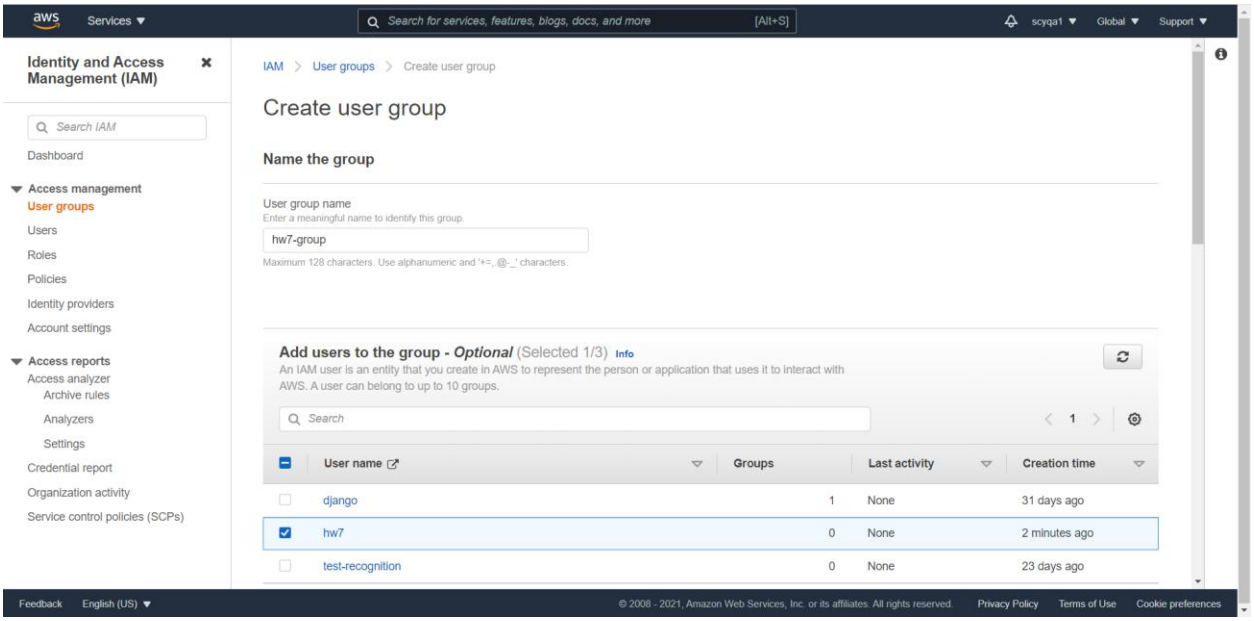
Review



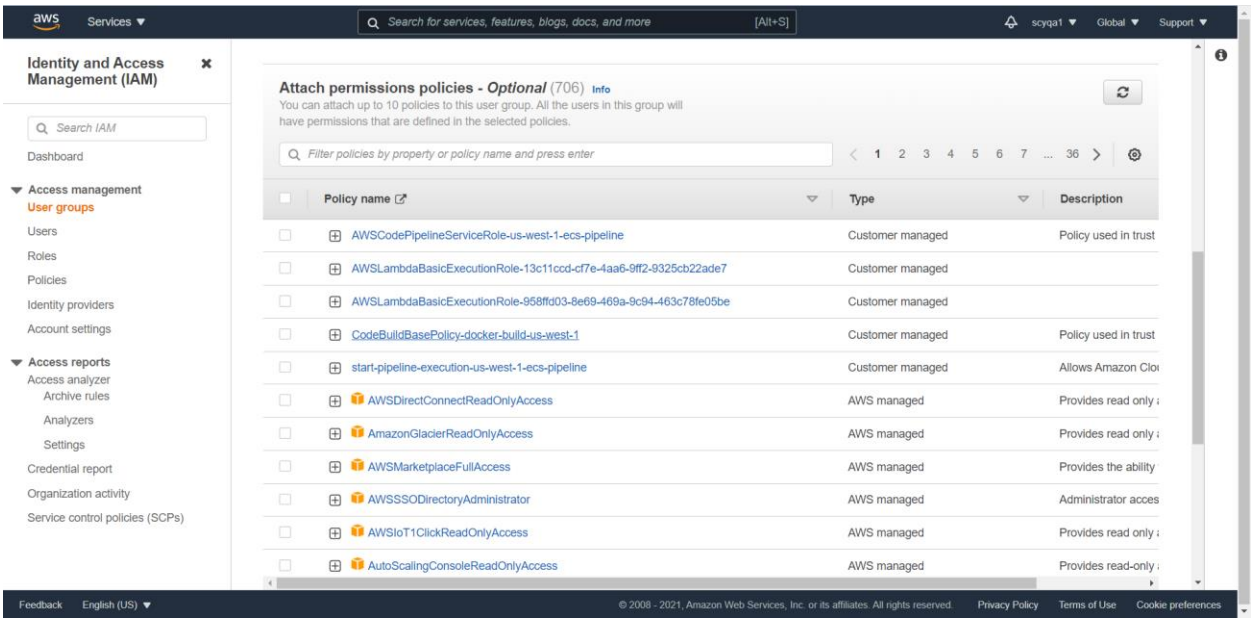
Result:



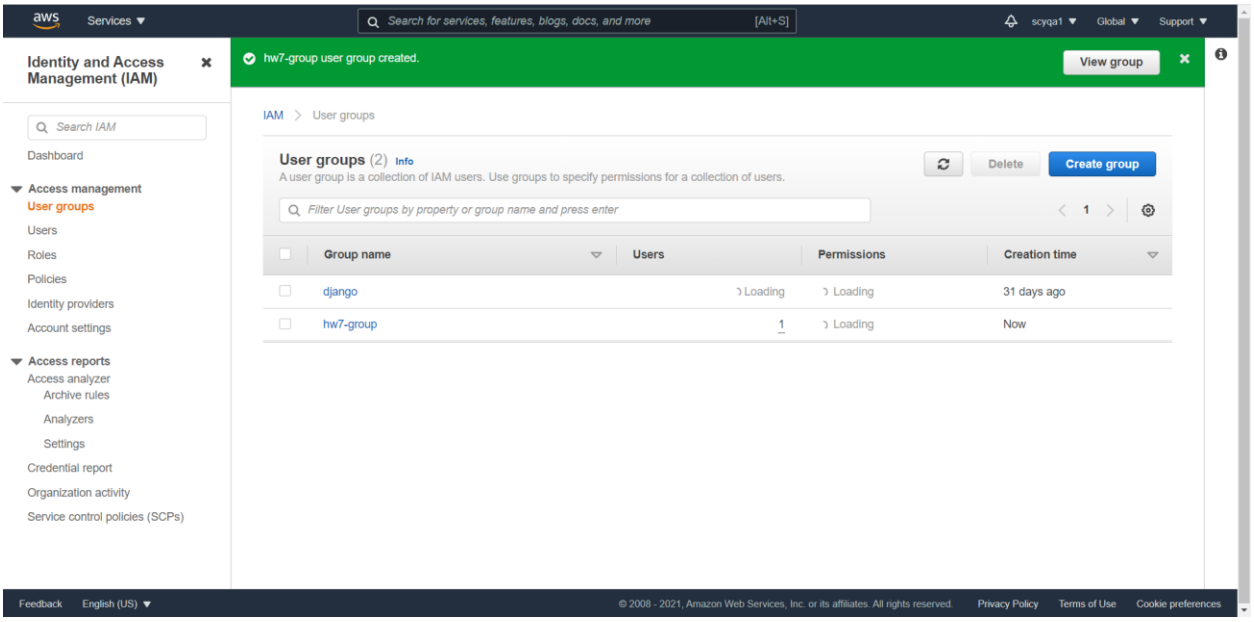
Create an IAM group:



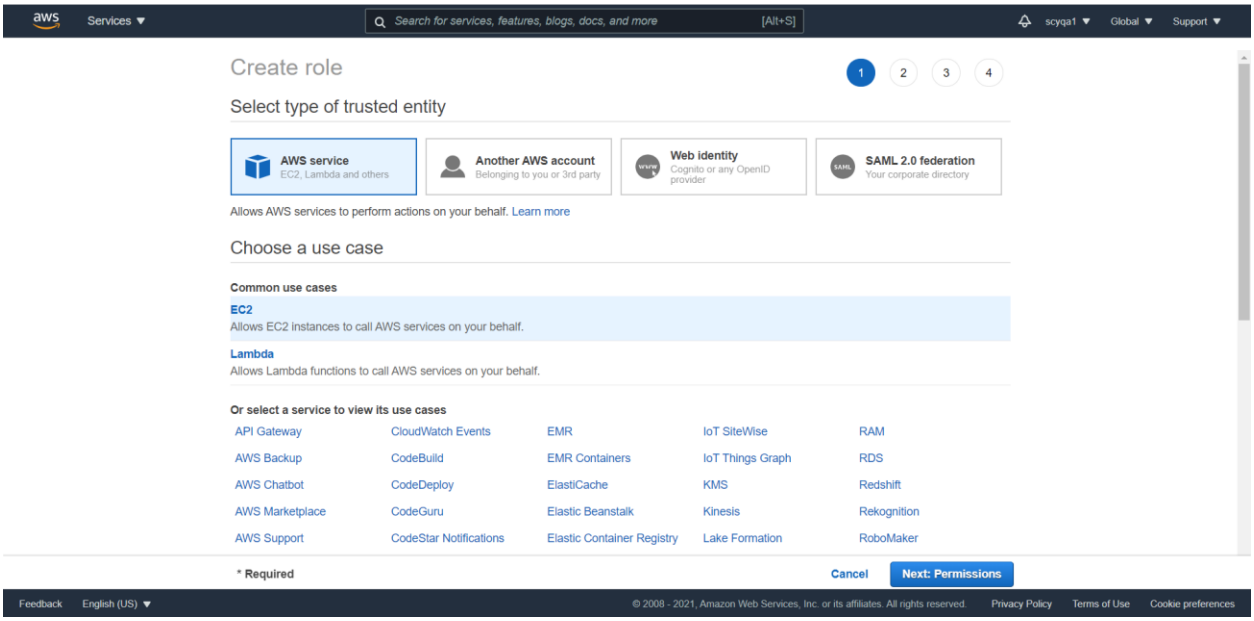
Attach policies



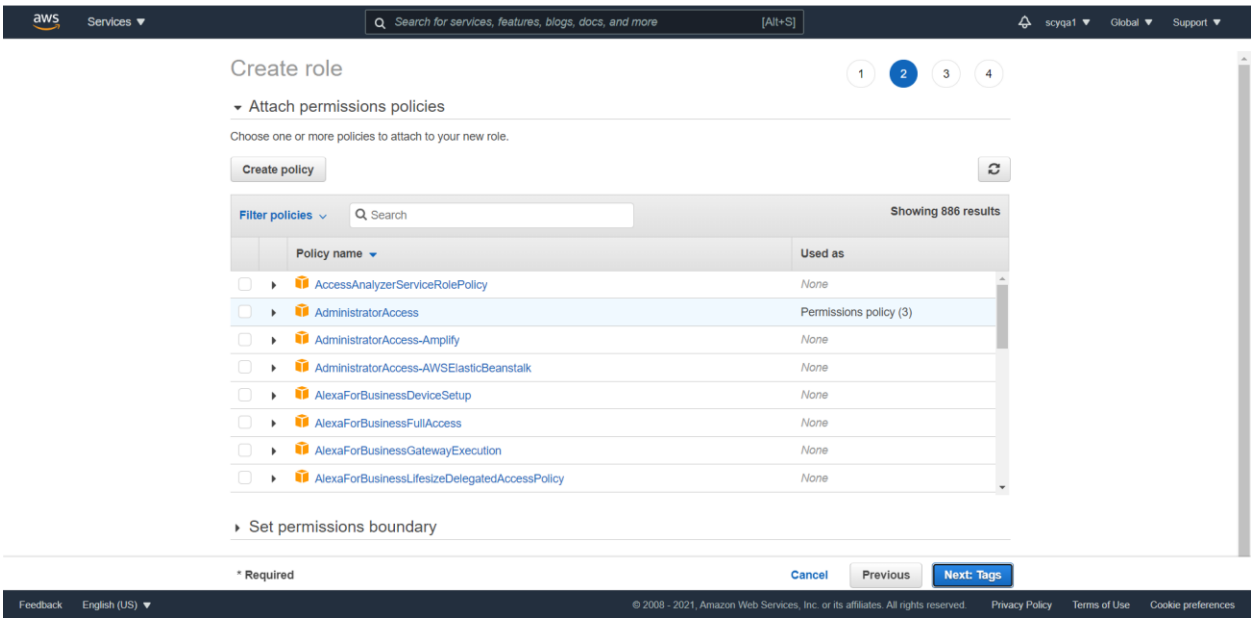
Result


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
scyqa1
Global
Support
Identity and Access Management (IAM)
hw7-group user group created.
View group
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
Service control policies (SCPs)
IAM > User groups
User groups (2) Info
A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.
Delete
Create group
Filter User groups by property or group name and press enter
1
Group name
Users
Permissions
Creation time
django
Loading
Loading
31 days ago
hw7-group
1
Loading
Now
Feedback
English (US)
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences

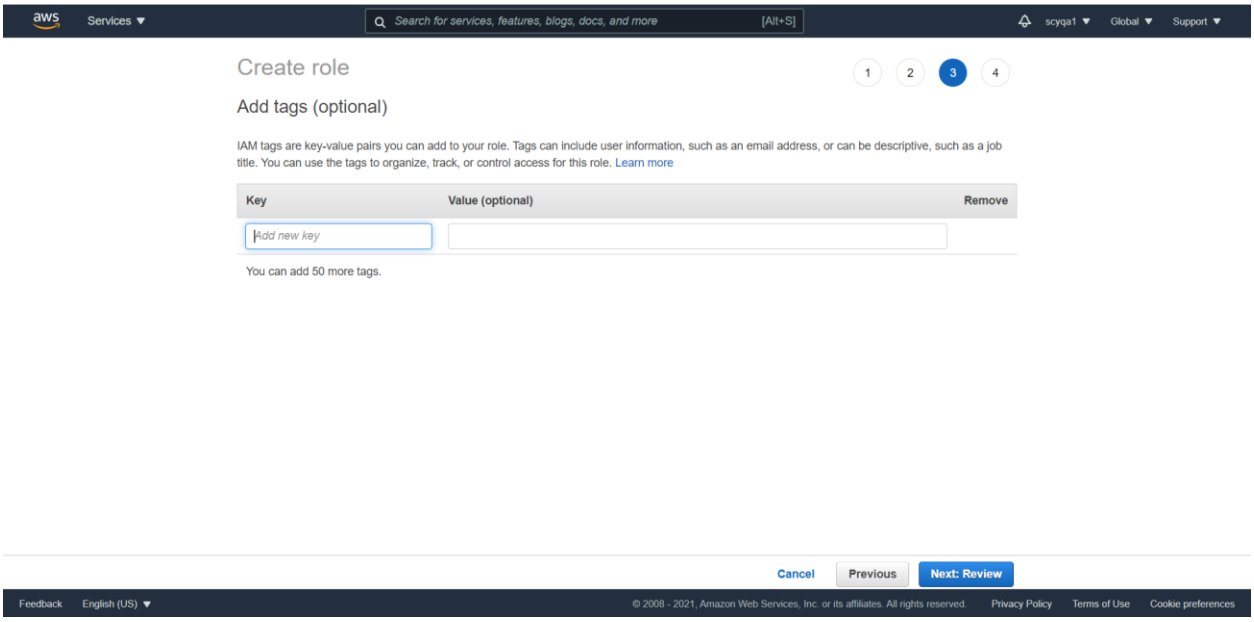
Create an IAM role:



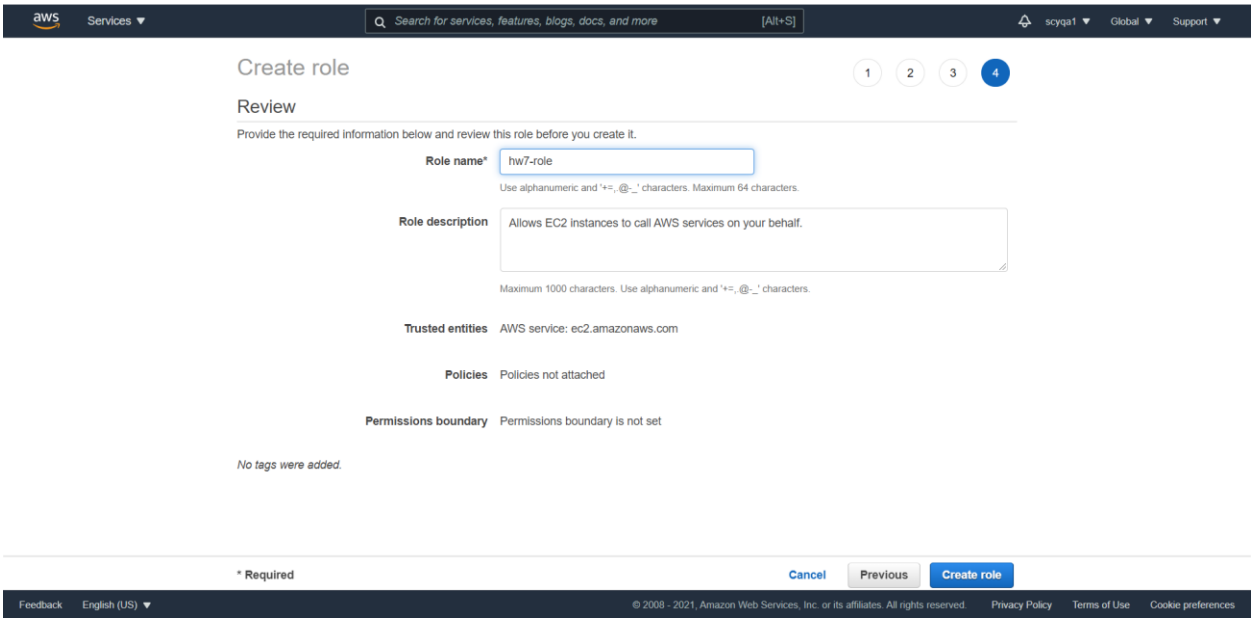
Attach policies



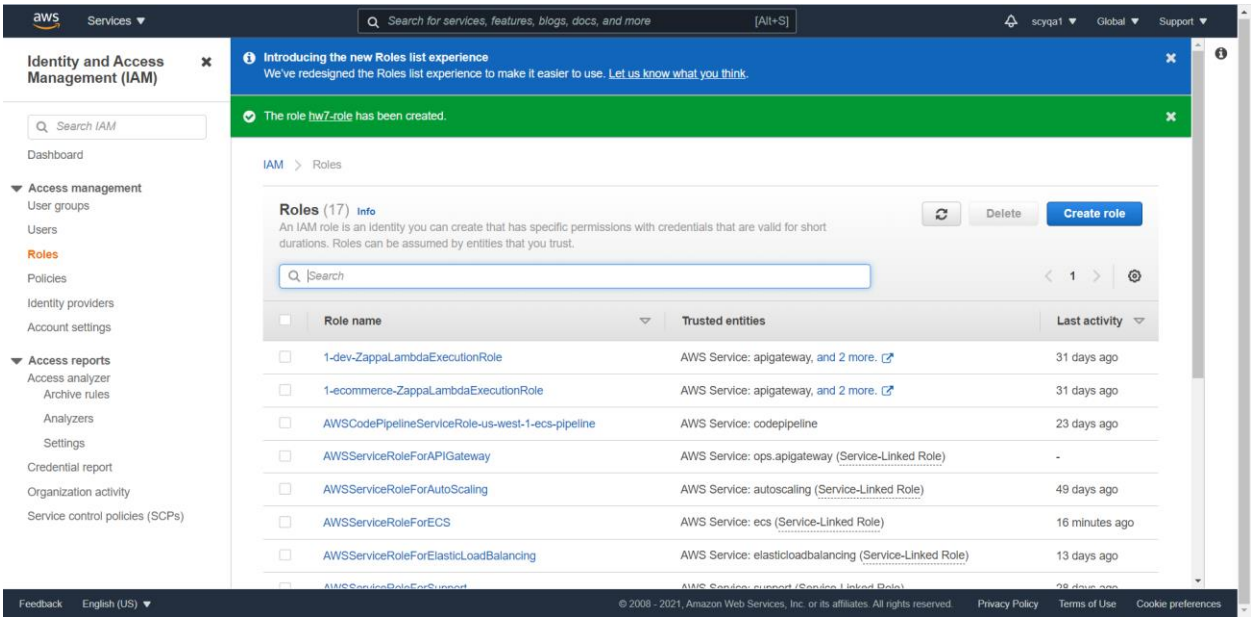
Add tags



Review



Result


aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
scyqa1
Global
Support
Identity and Access Management (IAM)
Introducing the new Roles list experience
We've redesigned the Roles list experience to make it easier to use. Let us know what you think.
The role hw7-role has been created.
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
Service control policies (SCPs)
IAM > Roles
Roles (17) Info
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.
Delete
Create role
Search
1
Role name
Trusted entities
Last activity
1-dev-ZappaLambdaExecutionRole
AWS Service: apigateway, and 2 more.
31 days ago
1-ecommerce-ZappaLambdaExecutionRole
AWS Service: apigateway, and 2 more.
31 days ago
AWSCodePipelineServiceRole-us-west-1-ecs-pipeline
AWS Service: codepipeline
23 days ago
AWSServiceRoleForAPIGateway
AWS Service: ops.apigateway (Service-Linked Role)
-
AWSServiceRoleForAutoScaling
AWS Service: autoscaling (Service-Linked Role)
49 days ago
AWSServiceRoleForECS
AWS Service: ecs (Service-Linked Role)
16 minutes ago
AWSServiceRoleForElasticLoadBalancing
AWS Service: elasticloadbalancing (Service-Linked Role)
13 days ago
Feedback
English (US)
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences

Explain why you need IAM for your eCommerce SaaS workloads:

For the eCommerce SaaS workloads, an IAM system can provide assurances and help keep track of employee activity. Having the ability to know that only certain employees can view programs and applications will strengthen both security and operational programs for an organization. Parameters can also be set in the system to detect any suspicious user activity, communication, or issues that might otherwise go undetected. User information, whether it is passwords or email addresses, can quickly become a complex issue to track without a proper control system in place. IAM helps protect against security incidents by allowing administrators to automate numerous user account related tasks. This includes the ability to have automated workflow for on-boarding of employees, granting access to systems and applications they are authorized access to, based on their role. It also includes "one button" control to remove employee access from all systems they were granted access to through the IAM platform.

IAM solutions help the eCommerce SaaS workloads meet industry compliance requirements and help them save costs by minimizing the time needed to deal with user account related issues. Identity and access management standardizes and even automates critical aspects of managing identities, authentication, and authorization, saving time and money while reducing risk to the business.

The varying aspects of protection offered by IAM solutions are key to building a strong information security program. These are just some of the areas security professionals must consider while developing strong identity and access control systems to protect their organizations. The ability to be able to control and audit who comes in and out of your organization's network is vital to operationally supporting and securing an environment.

2. Try the Encryption code (a) AES and (b) asymmetric (Box 13.3 and 13.4) on any example data and show/explain results.

(a). AES

```
(base) aqc@ubuntu:~/hw7$ vim sysEncry.py
```

aqc@ubuntu: ~/hw7

```
from Cropto.Cipher import AES

aesObj = AES.new('This is a key123', AES.MODE_CBC, 'This is an IV46')
data = "Hello World!"

encrypted_data = obj.encrypt(data)
print encrypted_data

asdObj1 = AES.new('This is a key123', AES.MODE_CBC, 'This is an IV46')
decrypted_data = aesObj1.decrypt(encrypted_data)
print decrypted_data
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                              11,20         All
```

Packages:

```
(base) aqc@ubuntu:~/hw7$ pip install Crypto
Collecting Crypto
  Downloading crypto-1.4.1-py2.py3-none-any.whl (18 kB)
Collecting Naked
  Downloading Naked-0.1.31-py2.py3-none-any.whl (590 kB)
     |████████████████████████████████| 590 kB 327 kB/s
Collecting shellescape
```

```
(base) aqc@ubuntu:~/hw7$  pip install pycryptodome
Collecting pycryptodome
  Downloading pycryptodome-3.11.0-cp35-abi3-manylinux2010_x86_64.whl (1.9 MB)
     |████████████████████████████████| 1.9 MB 361 kB/s
Installing collected packages: pycryptodome
Successfully installed pycryptodome-3.11.0
```

Results:

```
(base) aqc@ubuntu:~/hw7$ python sysEncry.py
b'`7z\xba,\x94\x97\x84\x9c\xa0.\xac\x19\xcaEh'
b'Hello World!\x00\x00\x00\x00'
```

(b) asymmetric

```
(base) aqc@ubuntu:~/hw7$ vim asyEncry.py
```

aqc@ubuntu: ~/hw7

```
from Crypto.PublicKey import RSA
from Crypto import Random
random_generator = Random.new().read

key_size = 1024

key = RSA.generate(key_size, random_generator)

publica_key = key.publickey()
print public_key

data = "Hello World!"

encrypted_data = public_key.encrypt(data,32)
decrypted_data = key.decrypt(encrypted_data)
~
~
~
~
~
~
~
~
-- INSERT --                                    15,45         All
```

Results:

```
(base) aqc@ubuntu:~/hw7$ python asyEncry.py
Public RSA key at 0x7F858312E430
```