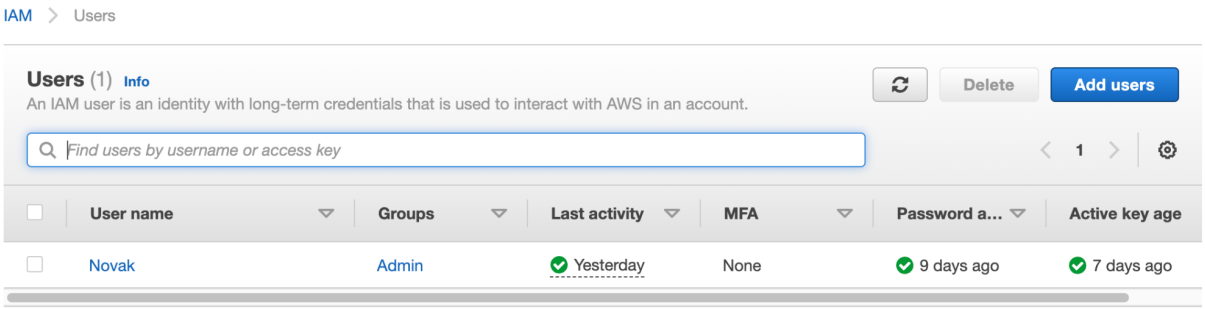
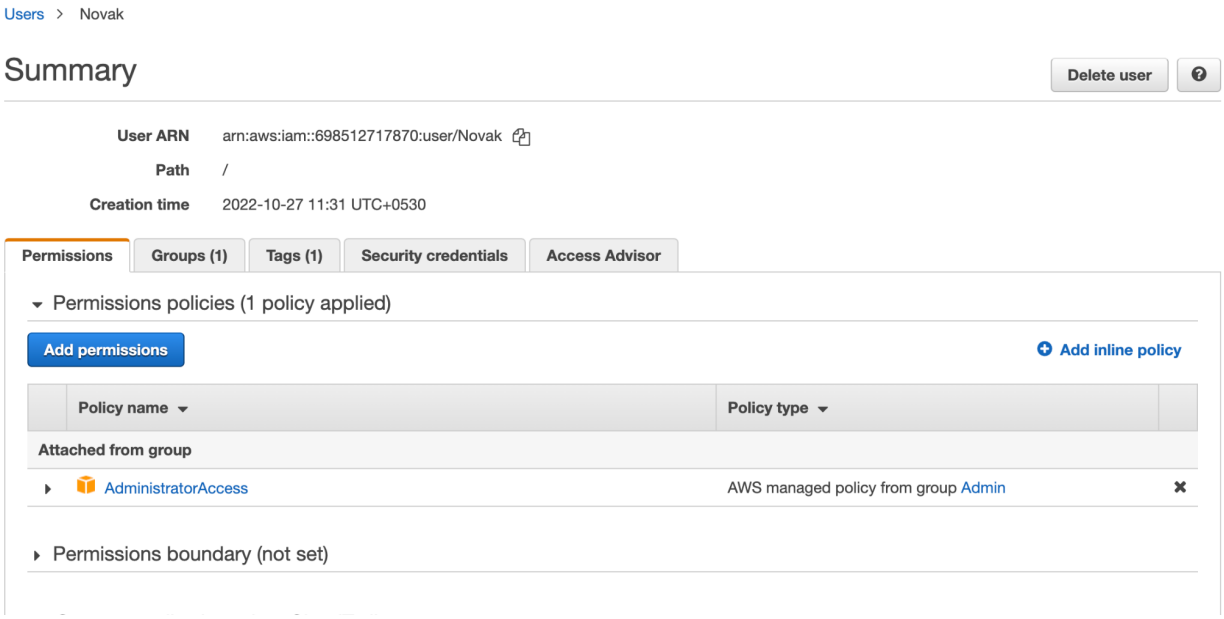
Assignment sheet for IAM

Assignment 1:- Create an IAM user with the username of your own wish and grant administrator policy.

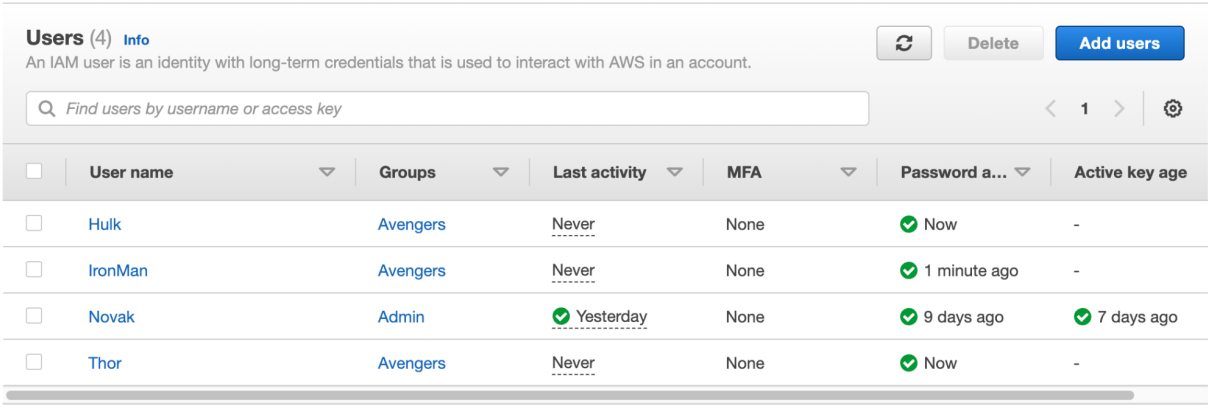
Proof:

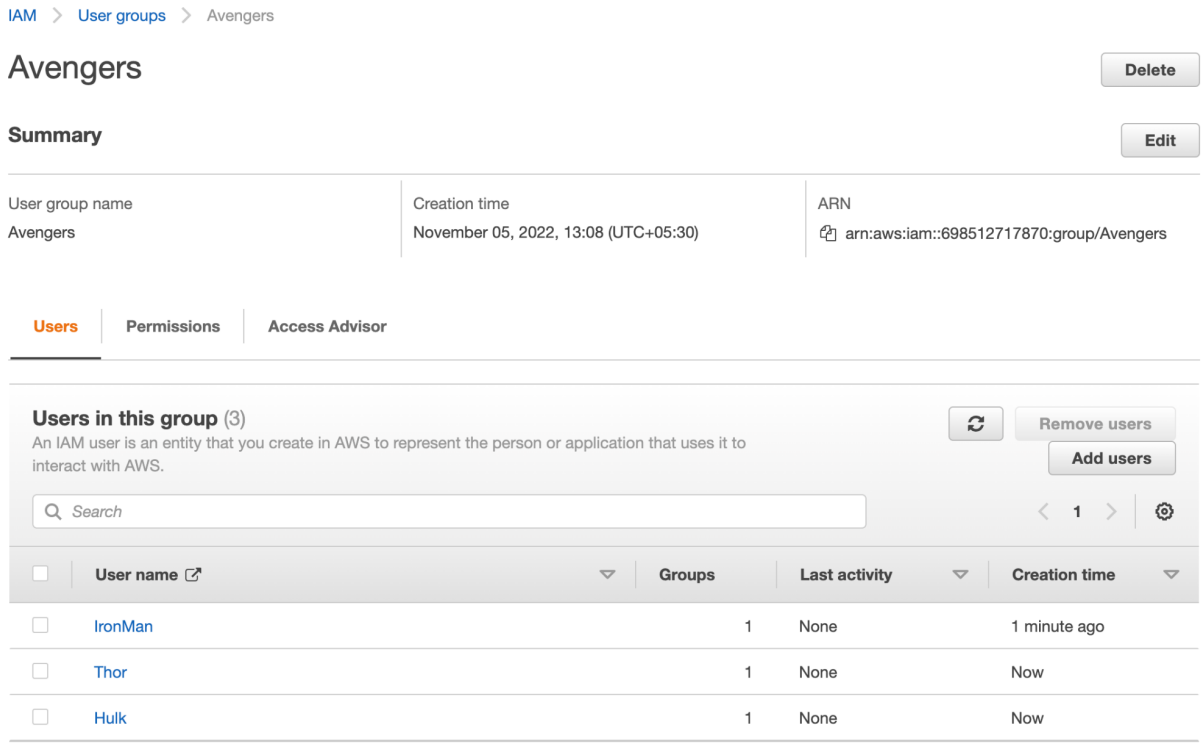




Assignment 2:- Hello students, in this assignment you need to prepare a developers team of avengers.
- Create 3 IAM users of avengers and assign them to developer's groups with IAM policy.

Proof:





Assignment 3:- Define a condition in policy for expiration like
"DateGreaterThan": {"aws:CurrentTime": "2020-04-01T00:00:00Z"},

"DateLessThan": {"aws:CurrentTime":

"2020-06-30T23:59:59Z"}
Define the span of 4 months as per your wish

Proof:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "iam:GenerateCredentialReport",
                "iam:GetPolicyVersion",
                "iam:GetAccountPasswordPolicy",
                "iam:GetServiceLastAccessedDetailsWithEntities",
                "iam:GenerateServiceLastAccessedDetails",
                "iam:GetServiceLastAccessedDetails",
                "iam:GetGroup",
                "iam:GetContextKeysForPrincipalPolicy",
                "iam:GetOrganizationsAccessReport",
                "iam:GetServiceLinkedRoleDeletionStatus",
                "iam:SimulateCustomPolicy",
                "iam:SimulatePrincipalPolicy",
                "iam:GenerateOrganizationsAccessReport",
                "iam:GetAccountAuthorizationDetails",
                "iam:GetCredentialReport",
                "iam:GetSAMLProvider",
                "iam:GetServerCertificate",
                "iam:GetRole",
                "iam:GetInstanceProfile",
                "iam:GetPolicy",
                "iam:GetAccessKeyLastUsed",
                "iam:GetSSHPublicKey",
                "iam:GetContextKeysForCustomPolicy",
                "iam:GetUserPolicy",
                "iam:GetGroupPolicy",
                "iam:GetUser",
                "iam:GetOpenIDConnectProvider",
                "iam:GetRolePolicy"
            ],
            "Resource": "*",
            "Condition": {
                "DateGreaterThan": {
                    "aws:CurrentTime": "2022-11-04T00:00:00Z"
                },
                "DateLessThan": {
                    "aws:CurrentTime": "2023-02-04T23:59:59Z"
                }
```

```
        }
      }
    ]
}
```

Policies › iNeuron-ExpirationPolicy-Assignment

Summary

Delete policy

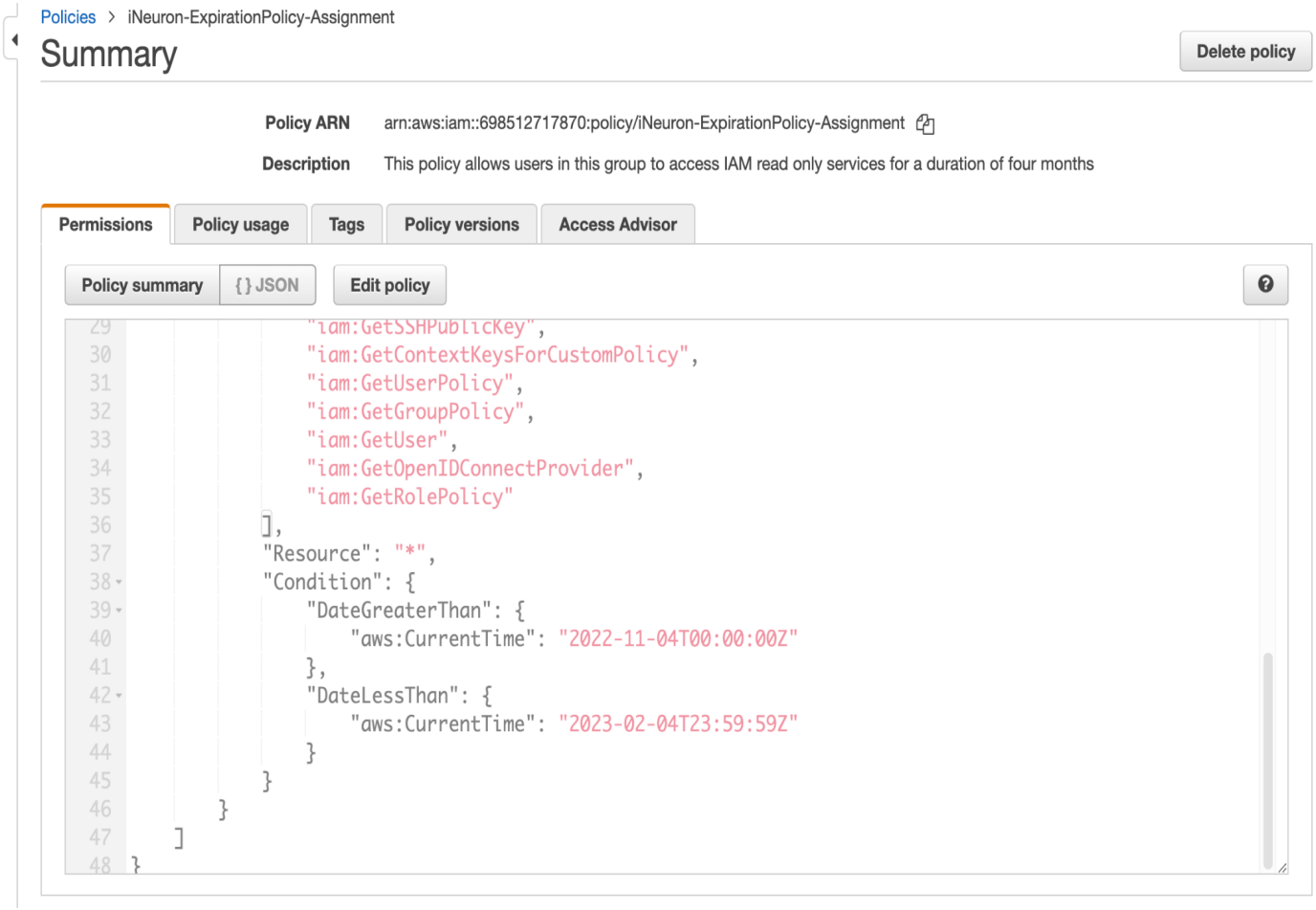
Policy ARN arn:aws:iam::698512717870:policy/iNeuron-ExpirationPolicy-Assignment

Description This policy allows users in this group to access IAM read only services for a duration of four months

Permissions | Policy usage | Tags | Policy versions | Access Advisor

Policy summary | {} JSON | Edit policy


```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "iam:GenerateCredentialReport",
                "iam:GetPolicyVersion",
                "iam:GetAccountPasswordPolicy",
                "iam:GetServiceLastAccessedDetailsWithEntities",
                "iam:GenerateServiceLastAccessedDetails",
                "iam:GetServiceLastAccessedDetails",
                "iam:GetGroup",
                "iam:GetContextKeysForPrincipalPolicy",
                "iam:GetOrganizationsAccessReport",
                "iam:GetServiceLinkedRoleDeletionStatus",
                "iam:SimulateCustomPolicy",
                "iam:SimulatePrincipalPolicy",
```



Assignment 4:- Prepare 15 authentic MCQ questions related to IAM.

1. What is AWS IAM?

The Amazon Web Services Identity and Access Management service are like a security guard at the door to Amazon Web Services. This is where Azure Services and its environment are authenticated and authorized.

The basic building blocks of AWS IAM are IAM roles, IAM users, groups and policies.

2. What are the best practices you will follow while creating IAM users?

We should always create individual IAM users for each person needing access to AWS services. Even if there are many employees who require the same access, we should create individual IAM users for all of them. This increases the security posture by providing every user of IAM with a unique set of credentials.

3. Explain AWS IAM Policies.

IAM Policies are how you determine who has access to what resources in your account. For example, you could allow users access to all Amazon EC2 instances within your AWS account, or just a specified instance.

4. AWS policies are of two types:

- Identity-based policies: This is the policy that binds with AWS identities, such as a user, group, or role. IAM policies are an example of that. These policies can be either Amazon Web Services managed or customer-managed.
- Resource-based policies: AWS resource-based policies are the ones that can be tied directly to Amazon Resources like a bucket policy (S3). Resource-based policies are only available for certain services.

5. What are the best practices you would follow while creating any IAM Policy?

When granting permissions, we should follow the least privileged principle. We should avoid giving users or roles more permissions than they need to accomplish their tasks by following this principle. For example, if an employee needs only access to a specific EC2 instance, specify the instance in the IAM policy. Rather than granting an employee access to every instance in your AWS account.

6. Please explain the IAM Policy Structure.

We can create IAM policies from the AWS web console and by the visual editor using the JASON-based policy editor. If you take a look at the JASON policy document it basically consists of below elements:

- Effect — Decides whether the resource is allowed or denied (Allow/Deny)
- Action — A set of service-specific parameters
- Resource — Resource names
- Condition (Optional) — Grant conditions

7. Define AWS IAM roles.

An IAM role is a temporary way to access permissions through your identity.

8. What is a Root user?

The Root User is the Owner Account (administrator) that is created when the AWS Account is created. By default, it has access to all AWS services and resources. It is not possible for IAM Policies to explicitly deny this user access to AWS services or resources.

9. How do you revoke access rights?

If you need to revoke access rights from an existing user, it's simple. Simply click on Manage Permissions on his or her profile page and select Revoke Access. You'll be presented with a list of all services to which they are granted access; check each service that is correct and then click Revoke Access in the bottom right corner.

10. What is MFA in AWS IAM?

Multi-factor authentication (MFA) adds an extra layer of security for users accessing AWS resources. In addition to a username and password, an MFA-enabled user must provide a one-time code generated by an authenticator app or sent via SMS or voice call before gaining access. An MFA device can be enabled on your computer, phone, or tablet.

11. Is it possible to monitor the activity of IAM users?

Yes, IAM users' activities can be monitored. In case of a violation, you can remove the IAM user's access.

12. What are IAM users' access keys?

Each IAM user receives an access key along with a secret key. Users can use their access keys to authenticate themselves to Amazon Web Services when they launch an instance, run a command, or call an API. If you lose your access key, please make sure that you terminate all instances and delete any resources linked to them before creating a new one. If you lose your secret key, we recommend deleting all related resources in order to minimize potential harm.

13. What is Access control to AWS resources?

The first step in securing your resources is using access control lists (ACLs) to allow or deny access. An AWS account has an owner, so you need an access key and secret key when using ACLs with any service. Make sure you keep these keys safe! The first step in securing your resources is using access control lists (ACLs) to allow or deny access. An AWS account has an owner, so you need an access key and secret key when using ACLs with any service.

14. What are the key features of AWS IAM?

- Access control to AWS resources
- Multi-factor authentication (MFA)
- Federated access
- Analytics

15. Explain best practices to manage access to AWS resources?

- Do not use root accounts – Since root accounts have access to all the AWS resources and services, it is not a good idea to share or use them.
- Use Groups – Create groups, grant access to them, and add users to them – so that all users within the group have the same access.
- Enable Multi-factor Authentication (MFA) – MFA should be enabled for privileged users such as admins. MFA adds an additional layer of security.
- Grant least privileges – Only grant permissions that are necessary for the user or group.

Assignment 5:- Launch your Linux instance in IAM and update your machine.

Proof: