AWS Security Series

Part 2.2: AWS Identity & Access Management: The Risk Perspective

Nitin Sharma
CyberSecurity and DevSecOps Engineer
LinkedIn: linkedin.com/in/nitins87
Quora: quora.com/profile/NitinS-1
Blog: 4hathacker.in

Contents

$ whoami

$ IAM - Principle of Least Privilege

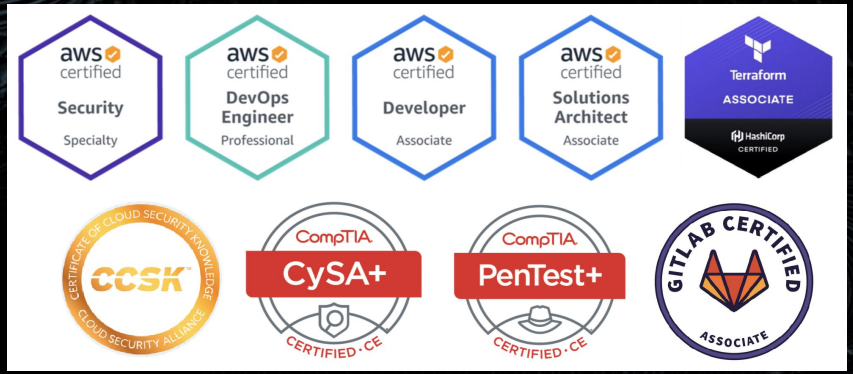
$ AWS IAM - Different Attack Patterns

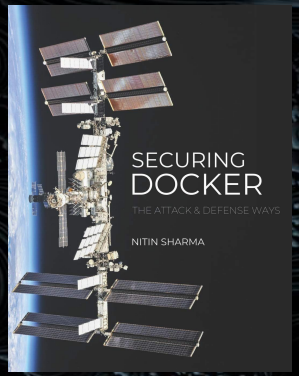
$ AWS IAM - Triaging Violations

$ Demo - AWS IAM Assessment with Cloudsplaining

$ whoami

- Cybersecurity and DevSecOps professional experienced in Cloud Security, Container Security and DevOps Research
- Certifications:





- Published author for "Securing Docker - The Attack & Defense Ways" book under CyberSecrets Publication
- Half Marathon runner, Cyclist and Fitness Enthusiast
- Helping out beginners in Cloud, DevOps and CyberSec at Quora

$ whoami (Mordac in the past)

$ IAM - Principle of Least Privilege

- According to CISA (US-CERT website archive),

> *"Only the minimum necessary rights should be assigned to a subject that requests access to a resource and should be in effect for the shortest duration necessary (remember to relinquish privileges). Granting permissions to a user beyond the scope of the necessary rights of an action can allow that user to obtain or change information in unwanted ways. Therefore, careful delegation of access rights can limit attackers from damaging a system."*

(Source: US-CERT Archive)

$ IAM - Principle of Least Privilege

- **Scenario**:
  - Let's say, you were to go on vacation,
  - And give your friend
    - The key to to your home,
      - To feed pets
      - Collect mail, etc.

$ IAM - Principle of Least Privilege

- **Associated Risks:**
  - **While you may trust a friend, there is always the possibility that there will be a party in your house without your consent, or that something else will happen that you don't like.**

$ IAM - Principle of Least Privilege

- **Associated Risks/Attack Patterns:**
  - **Additional abuse of house property.**
  - **Whenever a key to your house is out of your control, there's a risk of that key getting duplicated. If there's a key outside your control, and you're not home, then there's the risk that the key is being used to enter your house. Any length of time when someone has your key and is not being supervised by you constitutes a window of time in which you are vulnerable to an attack.**

$ IAM - Principle of Least Privilege

- **Associated Risk Triage and Best Practices:**
  - Whether or not you trust your friend, there's really no need to put yourself at risk by giving more access than necessary.
  - For example,
    - If you don't have pets, you should relinquish only the mailbox key. **[Least Privilege - Reducing Additional Abuse]**
    - If you do get a house sitter while you're on vacation, you aren't likely to let that person keep your keys when you're not on vacation. **[Least Privilege - Temporary/Time bound Access]**

$ IAM - Principle of Least Privilege

- **Associated Risk Triage and Best Practices:**
  - Whether or not you trust your friend, there's really no need to put yourself at risk by giving more access than necessary.
  - For example,
    - If you don't have pets, you should relinquish only the mailbox key. **[Least Privilege - Reducing Additional Abuse]**
    - If you do get a house sitter while you're on vacation, you aren't likely to let that person keep your keys when you're not on vacation. **[Least Privilege - Temporary/Time bound Access]**

$ AWS IAM - Risks/Attack Patterns

- Privilege Escalation
- Data Exfiltration
- Resource Exposure
- Credentials Exposure
- Infrastructure Modification

$ AWS IAM - Risks/Attack Patterns

#1 Privilege Escalation:

When the IAM Policy allows a combination of IAM actions that allow a 'Principal' with these permissions to escalate their privileges - for example,

- By creating an access key for another IAM user, or modifying their own permissions.

$ AWS IAM - Risks/Attack Patterns

#1 Privilege

Pitfalls while IAM.

(Source: BishopFox AWS Cheat Sheet)

01	**Allowing IAM Permission on Policies** No normal user should be able to change policies that apply to themselves or others.	**Affected Permissions** iam:CreateAccessKey iam:CreateLoginProfile iam:UpdateLoginProfile iam:AddUserToGroup
02	**Allowing IAM Permissions on Other Users** No normal user should be able to change the properties of other users, either directly or through modification of groups or roles.	**Affected Permissions** iam:CreatePolicyVersion iam:SetDefaultPolicyVersion iam:AttachUserPolicy iam:AttachGroupPolicy iam:AttachRolePolicy iam:PutUserPolicy iam:PutGroupPolicy iam:PutRolePolicy
03	**Updating an AssumeRolePolicy** No normal user should be able to change a role's AssumeRolePolicy.	**Affected Permissions** iam:AssumeRolePolicy
04	**Allowing iam:PassRole with Wildcards** Using wildcards will generally lead to dangerous privileges, but iam:PassRole in particular can cause problems as it may allow users to pass privileged roles to AWS services under their control.	**Affected Permissions** iam:PassRole
05	**Allowing Priv Esc through AWS Services** Limit the permissions of users on all AWS services, since specific combinations of permissions and services can lead to privilege escalation.	**Affected Permissions** Lambda Glue Cloud formation Data pipeline

avoid

onfiguring

$ AWS IAM - Risks/Attack Patterns

#2 Data Exfiltration:

When the IAM Policy contains actions that could allow certain read-only IAM actions without resource constraints such as,

- **s3:GetObject**, **ssm:GetParameter***, or **secretsmanager:GetSecretValue**. Unrestricted s3:GetObject permissions has a long history of customer data leaks.
- **rds:CopyDBSnapshot** and **rds:CreateDBSnapshot** can be used to exfiltrate RDS database contents.

$ AWS IAM - Risks/Attack Patterns

#3 Resource Exposure:

When the IAM Policy allows actions that permit modification of resource-based policies or can otherwise expose AWS resources to the public via similar actions, lead to resource exposure.

- s3:PutObjectAcl grants permission to modify the access control list (ACL) permissions for new or existing objects in an S3 bucket, which could expose objects to rogue actors or to the internet.

- The ability to modify AWS Resource Access Manager, which could allow a malicious actor to share a VPC hosting sensitive or internal services to rogue AWS accounts.

$ AWS IAM - Risks/Attack Patterns

#4 Credential Exposure:

When the IAM Policy allows actions that could end up returning credentials as part of the API response, such as **ecr:GetAuthorizationToken**, **iam:UpdateAccessKey**, and others.

```
340    "upload artifacts":
341      - command: s3.put
342        params:
343          optional: true
344          aws_key: "AKIAWFKNZA74UT3FFXHB"
345          aws_secret: "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"
346          local_file: mongosql-auth-c/test/artifacts/release.tgz
347          remote_file: mongosql-auth-c/artifacts/${build_variant}/${task_id}/mo
```

$ AWS IAM - Risks/Attack Patterns

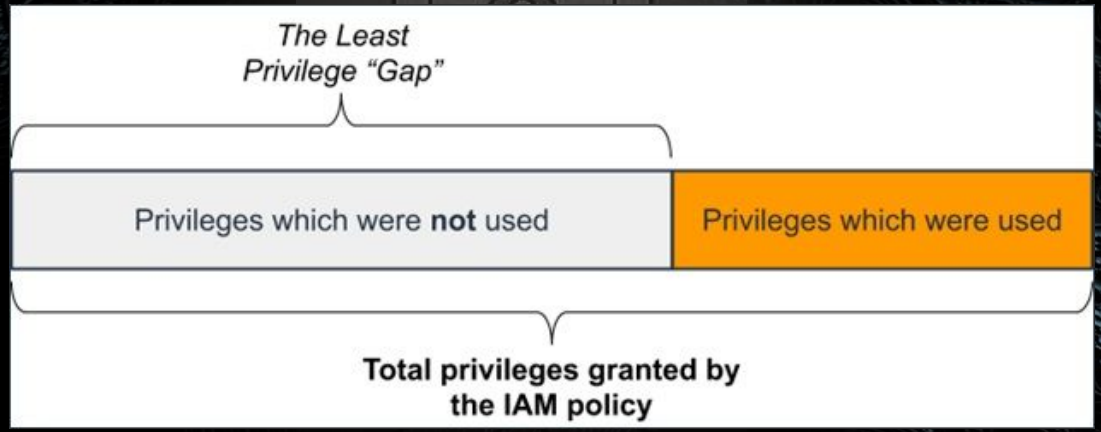
#5 Infrastructure Modification:

This describes IAM actions with “modify” capabilities, and can therefore lead to Resource Hijacking, unauthorized creation of resources, backdoor creation, and /or modification of existing resources resulting in downtime - for example,

- **ec2:AuthorizeSecurityGroupIngress** grants the permission to add one or more inbound rules to a security group; malicious usage of this IAM action could potentially lead to unintentional exposure of EC2 compute resources.

$ AWS IAM - Best Practices for Triaging

The Security team's role in the DevOps world is to enable developers/engineers/admins to do their jobs within the guardrails of what's acceptable and with the tools needed to make secure design decisions.



(Source: Approaching Least Privileges - AWS APN Blog)

$ AWS IAM - Best Practices for Triaging

1. **Segregate identities in separate environments.**

   Use DEV, PROD and QA environments along with different identities in each.

2. **Prefer AWS Custom Policies over AWS Managed Policies.**

   *AWS managed policies always include access to * resources because AWS provides these same policies universally to all customer accounts.*

3. **Differentiate System Roles with User Roles**

   IAM System roles should have limited and specific 'Write' permissions while no User role should have ':*' permissions.*

4. **Utilize IAM SCP or Permission boundaries as guardrails.**

   It's always recommended to scope-in IAM Policy actions with effective guardrails.

5. **Try to use "Conditions" logic in IAM policies intelligently.**

   Condition statement in IAM policies often result in resource exposure due to dynamic IPs and other things.

AWS USER GROUP NCR

$ AWS IAM Assessment - CloudSplaining

Cloudsplaining (Open Source tool by K. McQuade, Salesforce) is an AWS IAM Security Assessment tool that identifies violations of least privilege and generates a risk-prioritized HTML report. The tool aims at:

- Map out your risk landscape of IAM identity-based policies, enumerating the potential risks for a full IAM threat model
- Identify where you can reduce the blast radius in the case of credentials compromise
- Help you prioritize which ones to remediate
- Provide a straightforward workflow to remediate
- Provide a sufficient exclusions mechanism to programmatically define where deviations from resource constraints are by design

$ AWS IAM Assessment - CloudSplaining

Demo Time...

- How to install Cloudsplaining ?
- How to run an IAM Assessment ?
- Analyzing IAM Assessment Reports...

Thank You...

For queries feel free to connect with AWS Delhi User Group at:

1. AWS Delhi User Group (MeetUp) - Learn Together, Grow Together
2. AWS User Group Delhi NCR (Follow us on LinkedIn Page)