

AWS IAM Privilege Escalation Redux

int eighty (of Dual Core)

- Thank you for joining me
- Showcase collection of ways I have escalated privileges in AWS environments
- No fancy tooling (Pacu, Endgame, Cloudpslaining) — control plane, APIs
- Pentest stories: Things that should not have worked but did

Mom: Son why is the gas bill so high?

Me:



- Who here operates in an AWS environment?
- Elevate your power level so that you can protect your AWS bill
- Learn new tricks plus methodology
- Present some IAM priv esc paths I have not seen documented online AS SUCH*
- * Not SME on using search engines, best effort



int eighty

int eighty (he/him) is a computer crime enthusiast, and the rapper in Dual Core.

Occasional memes and hacking content on Mastodon and Twitter as [@int0x80](#).



@wafflesweekly

- Led public cloud security at large tech company
- Angel Dagger
- You belong here

Prior Art

- <https://rhinosecuritylabs.com/aws/aws-privilege-escalation-methods-mitigation/>
- <https://cloudsplaining.readthedocs.io/en/latest/glossary/privilege-escalation/>
- <https://bishopfox.com/blog/aws-iam-privilege-escalation-playground>
- https://hackingthe.cloud/aws/exploitation/iam_privilege_escalation/
- <https://makosecblog.com/aws-pentest/aws-managed-policies-that-allow-privesc/>
- <https://ermetic.com/blog/aws/auditing-passrole-a-problematic-privilege-escalation-permission/>

- Ermetic: methodology — how to research, learn, understand in AWS
- WARNING: cloud bills; use billing alarms, turn off unneeded resources
 - All resources, even AWS account, used for this talk are deleted

**You must tightly control
any mechanism(s) used
for authorization.**

- Failure to do so can be financially damaging to the extent of business-ending
- Trust is incredibly difficult to gain, and incredibly easy to lose
- Any gap is an opportunity for privilege escalation.
- Different models:
 - Managed policies attached to principals or groups
 - Principal paths and naming schemes
 - Tagging
- Common themes
 - There's an attempt at least privilege, usually
 - You must tightly control any mechanism(s) used for authorization.

Core Concepts

Terminology

 IAM user

IAM user

Identity on AWS used for web console access and/or programmatic API access

Core Concepts

Terminology

 IAM user

 IAM group

IAM group

Collection of one or more IAM users, unauthenticated entity

Core Concepts

Terminology

 IAM user

 IAM group

 IAM role

IAM role

Similar to IAM user, used by both humans and applications to interact with AWS services

To “assume” a role means to access and use the role

Core Concepts

Terminology

 IAM user

 IAM group

 IAM role

 IAM policy

IAM policy

Determine who can perform what actions on which resources, and under which conditions

Does your AWS environment:

☒ Prevent

☒ Detect

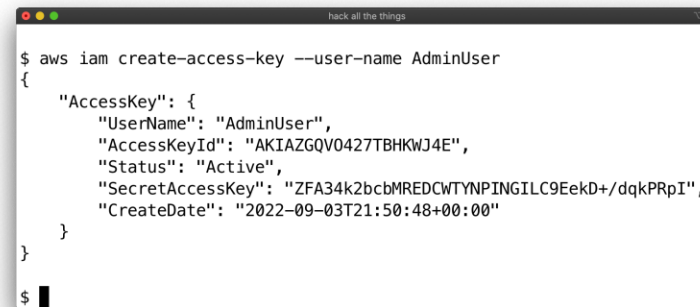
☒ Alert

- IAM users

IAM Users

iam:CreateAccessKey

- API credentials for IAM users
- Max TWO sets of access keys
- LimitExceeded



```
hack all the things
$ aws iam create-access-key --user-name AdminUser
{
  "AccessKey": {
    "UserName": "AdminUser",
    "AccessKeyId": "AKIAZGQV0427TBHKWJ4E",
    "Status": "Active",
    "SecretAccessKey": "ZFA34k2bcbMREDCWTYPINGILC9EekD+/dqkPRpI",
    "CreateDate": "2022-09-03T21:50:48+00:00"
  }
}
```

- Overlooked: Two sets of access keys
- Benefits
 - Persist when env rotates tracked access keys frequently
 - Less conspicuous than new IAM user
- LimitExceeded
 - iam:GetAccessKeyLastUsed displays N/A if never used
 - iam:ListAccessKeys for creation date & in/active status
 - Could remove one with iam>DeleteAccessKey



IAM Users

iam:CreateLoginProfile

- Credentials for AWS web console
- EntityAlreadyExists
- Avoid alerting
 - iam:GetAccountPasswordPolicy
 - iam:List[Virtual]MFADevices
 - iam:CreateVirtualMFADevice

```
hack all the things
$ aws iam create-login-profile \
  --user-name AdminUser \
  --password 'a better password than Hack All The Things1!' \
  --no-password-reset-required
{
  "LoginProfile": {
    "UserName": "AdminUser",
    "CreateDate": "2022-09-03T21:57:22+00:00",
    "PasswordResetRequired": false
  }
}
```

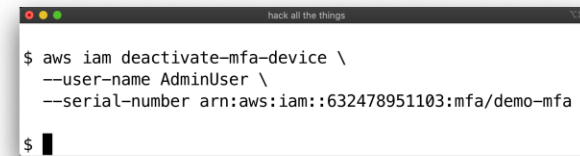
- Benefits

- Persist when env rotates tracked access keys frequently
- Does not create a new IAM user

IAM Users

iam:DeactivateMFADevice

- You compromise credentials for AWS web console
- aws:MultiFactorAuthPresent
- Avoid alerting
 - iam:CreateVirtualMFADevice
 - iam:EnableMFADevice



```
hack all the things 1/31
$ aws iam deactivate-mfa-device \
  --user-name AdminUser \
  --serial-number arn:aws:iam::632478951103:mfa/demo-mfa
$
```

- You somehow acquire user/pass for IAM user
- IAM user has MFA — you do not

Does your AWS environment:

- ☒ Prevent
- ☒ Detect
- ☒ Alert

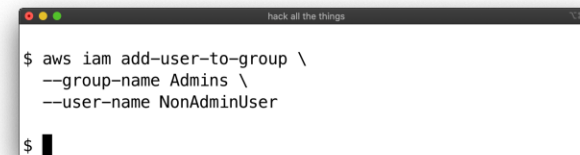
- IAM groups



IAM Groups

iam:AddUserToGroup

- Add non-admin user to admin group



```
hack all the things 1/20/21
$ aws iam add-user-to-group \
  --group-name Admins \
  --user-name NonAdminUser
$
```

- Bypass intended trust boundary
- More context up next

IAM Groups

iam:CreateGroup

- Administrative group renamed or removed
- Policies reference original group
- iam:UpdateGroup

```
hack all the things 1.38.3
$ aws iam create-group \
  --group-name ITSecurityAdmins
{
  "Group": {
    "Path": "/",
    "GroupName": "ITSecurityAdmins",
    "GroupId": "AGPAZGQV0427XWSITX6C6",
    "Arn": "arn:aws:iam::632478951103:group/ITSecurityAdmins",
    "CreateDate": "2022-09-03T23:58:43+00:00"
  }
}
$ aws iam add-user-to-group \
  --group-name ITSecurityAdmins \
  --user-name NonAdminUser
$
```

- Scenario: IT Security Admins rebranded to Cloud Security Admins
- Group named CloudSecurityAdmins added to IAM policies
- CloudSecurityAdmins group created, ITSecurityAdmins removed
- IAM policies not updated
- Same reason iam:AddUserToGroup has worked — policies out of sync w/ env state

IAM Groups

iam:RemoveUserFromGroup

- Group's policies use Effect of Deny
- Restricts members from actions



```
hack all the things 1/31/3
$ aws iam remove-user-from-group \
  --group-name Developers \
  --user-name NonAdminUser
$
```

Methodology

Give a mana fish, receive 30 mana over 6 seconds



Methodology

Recon

- Identify preventive controls
- Understand the AWS service
 - Search: <service> Actions Resources Conditions
 - ~~RTFM~~ RTFUG (User Guide)

- Identify preventive controls
 - Examine IaC in source control (tf, cft, et al)
 - Immutable APIs: describe, get, list
- Understand the AWS service
 - Most services have a Security section
 - Also valuable from remediation/engineering perspective



Methodology

Advance

- Find a gap and take advantage
- Expand your sphere of presence

- This is “exploit” in traditional networks
- Find a gap and take advantage
 - No exploit here, simply using the control plane
 - Components need to communicate, effectively Live off the Land
- Expand your sphere of presence
 - Use new access to iterate and progress



Methodology

Loot

- Investigate with your new access
- Learn more about the environment

- Look for new:
 - Targets / Attack Surface
 - Credentials
 - Keys
 - Secrets
 - Source Code



Methodology

Miscellaneous

- Test in separate AWS account(s)
- Infrastructure as Code (IaC)
- NOTES

- Test in separate AWS account(s)
 - Prepare for prod
 - Utilize the AWS free tier
 - Leverage billing alarms
- Infrastructure as Code (IaC)
 - Easy to reproduce and iterate
- NOTES
 - Find what works for you
 - Aggregate, search, synthesize
- All this makes it easy to compose a talk!

Does your AWS environment:

- ☒ Prevent
- ☒ Detect
- ☒ Alert

- IAM roles



IAM Roles

iam:AddRoleToInstanceProfile

- Instance profiles
- Permissions defined by passed IAM role
- DEMO

- An instance profile allows an EC2 instance to comm w/ other AWS services
- IAM role passed to profile augment IAM permissions

IAM Roles

`iam:AddRoleToInstanceProfile`

- Calling principal needs `iam:PassRole`
- Instance profile max ONE role
 - Use `iam:CreateInstanceProfile`
 - Target instance profile with no associated role

- Caveats of `iam:AddRoleToInstanceProfile`

IAM Roles

iam:CreateRole

- Scenario: Privileged paths
- `arn:aws:iam::111122223333:role/admin/iam-admin`

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "iam:CreateAccessKey",
      "Resource": "*",
      "Condition": {
        "ArnNotLike": {
          "aws:PrincipalArn": "arn:aws:iam::*:role/admin/*"
        }
      }
    }
  ]
}
```

- Reliance on paths means protecting access to paths



IAM Roles

iam:CreateRole

- Scenario: Privileged paths — mitigation
 - Allow `iam:CreateRole`
 - Deny administrative path for **non-administrative** principals

- Reliance on paths means protecting access to paths

Does your AWS environment:

☒ Prevent

☒ Detect

☒ Alert

- IAM policies



IAM Policies

`iam:CreatePolicyVersion`

- Scenario: Deny `iam:Attach*Policy` and `iam:Put*Policy`
- Create new version of existing policy
- DEMO

- Vector: modify policy



IAM Policies

`iam:CreatePolicyVersion`

- Alternatively
 - `iam:GetPolicyVersion` + `iam:SetDefaultPolicyVersion`
 - `iam:Detach*Policy` + `iam>DeletePolicyVersion`
+ `iam>DeletePolicy` + `iam>CreatePolicy` *
- Potential mitigations
 - Deny `iam:*PolicyVersion`
 - Use a different mechanism (SCPs)

- Alt 1:
 - Find overly permissive previous versions with `iam:GetPolicyVersion`
 - Revert permissions with `iam:SetDefaultPolicyVersion`
- Alt 2 *:
 - Scenario: Governance mechanism automatically managed policy attachments
 - Remove policy, create new permissive policy
- Mitigation
 - Attaching `AdministratorAccess` should not allow priv esc



IAM Policies

iam:Detach*Policy

- Scenario: IAM policy restricts access using **Effect of Deny**
- Gap in controls
- DEMO

- Vector: detach the restrictive policy
- Priv esc does not always mean getting root/DA
- Sometimes just extra access



IAM Policies

iam:Detach*Policy

- Inverse of **Attaching** or **Putting** a more permissive policy
- Prevent unauthorized modifications
 - Policy content
 - Policy attachment

- Similar authorization bypass when leveraging implicit deny by default
 - Attach permissive managed policy
 - Put permissive inline policy

**You must tightly control
any mechanism(s) used
for authorization.**



- As attack surface increases, Priv Esc vectors increase
- Full admin access rarely needed to complete objectives
- Questions
- Call to Action