



**TECH
STARTER**

AWS - IAM

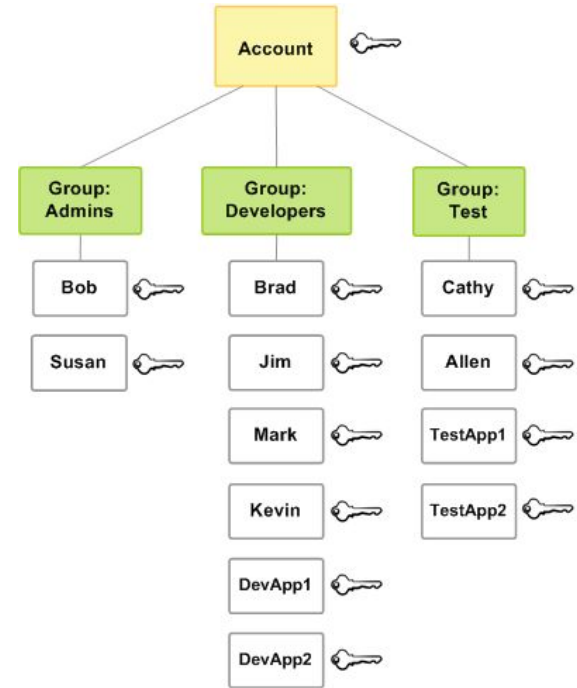
IAM



**TECH
STARTER**

IAM Benutzer und Gruppen

- IAM = Identity und Access Management
- Globaler Service
- Root Account wird per default erstellt und sollte nicht geteilt werden
- Benutzer sind Personen innerhalb des Unternehmens
- Gruppen können keine Untergruppen haben



Quelle: <https://docs.aws.amazon.com>

IAM Permissions

- Benutzer und Gruppen können Berechtigungen in JSON Format erteilt werden
- In AWS immer nach dem least privilege principle arbeiten
→ Nicht mehr Erlaubnis erteilen, als benötigt wird

```
{
  "Version": "2012-10-17",
  "Id": "S3-Account-Permissions",
  "Statement": [
    {
      "Sid": "1",
      "Effect": "Allow",
      "Principal": {
        "AWS": ["arn:aws:iam::123456789012:root"]
      },
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": ["arn:aws:s3:::mybucket/*"]
    }
  ]
}
```

IAM Policy Struktur

- Version: Definiert, wie das Dokument interpretiert werden soll
- Id (optional) Identifier
- Statement: Ein oder mehrere Berechtigungen
- Sid(optional): Identifier für das Statement
- Effect → Allow oder Deny
- Principal: account/user/role, dem die Policy zugeordnet werden soll
- Action: Erlaubte Aktionen

```
{
  "Version": "2012-10-17",
  "Id": "S3-Account-Permissions",
  "Statement": [
    {
      "Sid": "1",
      "Effect": "Allow",
      "Principal": {
        "AWS": ["arn:aws:iam::123456789012:root"]
      },
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": ["arn:aws:s3:::mybucket/*"]
    }
  ]
}
```

IAM - Passwort Policy

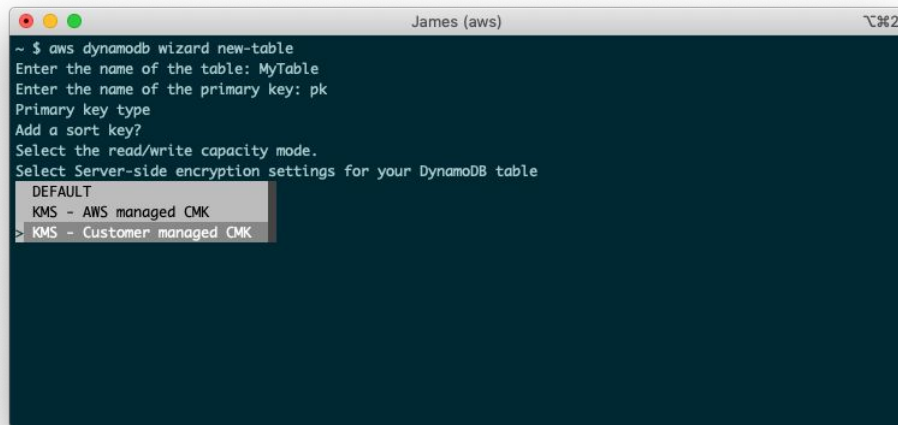
- Je stärker das Passwort, desto sicherer der Account
- z.B. Min. Passwort-Länge
- Benötigt bestimmte Zeichen
- Fordere an, dass Benutzer ihr Passwort nach geraumer Zeit ändern

Zugriff auf AWS

- Es gibt 3 Möglichkeiten
 - AWS Management Console (kennen wir)
 - AWS Command Line Interface (CLI): Geschützt durch Access key
 - AWS Software Developer Kit (SDK)
- Access Keys werden über AWS Console generiert
- Benutzer müssen ihre Access Keys verwalten
- Access Key ID ~= Benutzername
- Secret Access Key ~= Passwort

Was ist AWS CLI?

- Ein Tool, das es ermöglicht mit AWS über die Shell zu interagieren
- Kommunikation läuft über das AWS API ab
- Dadurch können Skript erstellt werden, um Ressourcen zu managen
- Open Source



```
~ $ aws dynamodb wizard new-table
Enter the name of the table: MyTable
Enter the name of the primary key: pk
Primary key type
Add a sort key?
Select the read/write capacity mode.
Select Server-side encryption settings for your DynamoDB table
  DEFAULT
  KMS - AWS managed CMK
> KMS - Customer managed CMK
```

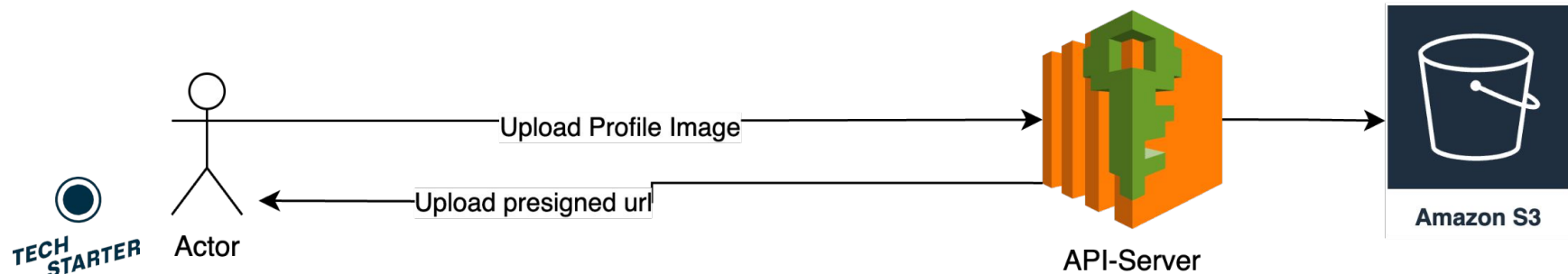
Quelle: <https://docs.aws.amazon.com>

Was ist AWS SDK?

- AWS Software Development Kit
- Programmiersprachen basierter Zugriff
- Auf AWS Services via Programmiersprache zugreifen
- In Applikationen integriert (Demo folgt)
- JavaScript, Python, PHP, .NET, Ruby, Java, Go, Node.js, C++)

IAM Service Rollen

- Service benötigt Berechtigungen, um auf anderen Service zuzugreifen
- Erlaubnis wird über IAM roles erteilt
- Beispiele: EC2 Instance Roles, Lambda Function Roles



IAM Sicherheit Tool

- IAM Credentials Report
 - Auflistung und Status der Credentials innerhalb eines Accounts
- IAM Access Advisor
 - Zeigt Service Berechtigungen, welche der Benutzer besitzt an und wann er diese zuletzt genutzt hat

IAM Best Practices

- Benutze den Root Account nur für Konfigurationen
- Für jede Person leg einen IAM Benutzer an
- Verwalte Benutzer über Gruppen und deren Berechtigungen
- Benutze MFA
- Benutze IAM Rollen, um Services Berechtigungen zu geben
- Benutze Access Keys für CLI/SDK
- Teile niemals deinen Access Key & Secret Access Key

Shared Responsibility Model IAM

AWS

- Infrastruktur
- Konfiguration und Sicherheitsanalysen

WIR

- Benutzer, Gruppen, Rollen, Richtlinien
- Rotation der Keys
- Analyse von Zugriffs patterns

IAM-Bereich - Zusammenfassung

- Benutzer(Users): Abgebildet auf einen physischen Benutzer, hat ein Passwort für die AWS-Konsole
- Gruppen(Groups): Enthält nur Benutzer
- Richtlinien(Policies): JSON-Dokument, das die Berechtigungen für Benutzer oder Gruppen festlegt
- Rollen(Roles): Für EC2-Instanzen oder AWS-Dienste
- Sicherheit(Security): MFA + Passwortrichtlinie
- AWS CLI: Verwalten Sie Ihre AWS-Dienste über die Befehlszeile
- AWS SDK: Verwalten Sie Ihre AWS-Dienste mithilfe einer Programmiersprache
- Zugangsschlüssel(Access Keys): Zugriff auf AWS über die CLI oder SDK
- Audit: IAM Credential Reports & IAM-Access Advisor