Zero Credential Development in Azure

Rupesh Kumar

27th Feb 2020

About Me

- Freelancer
- working@Snelstart, Alkmaar
- MCSE

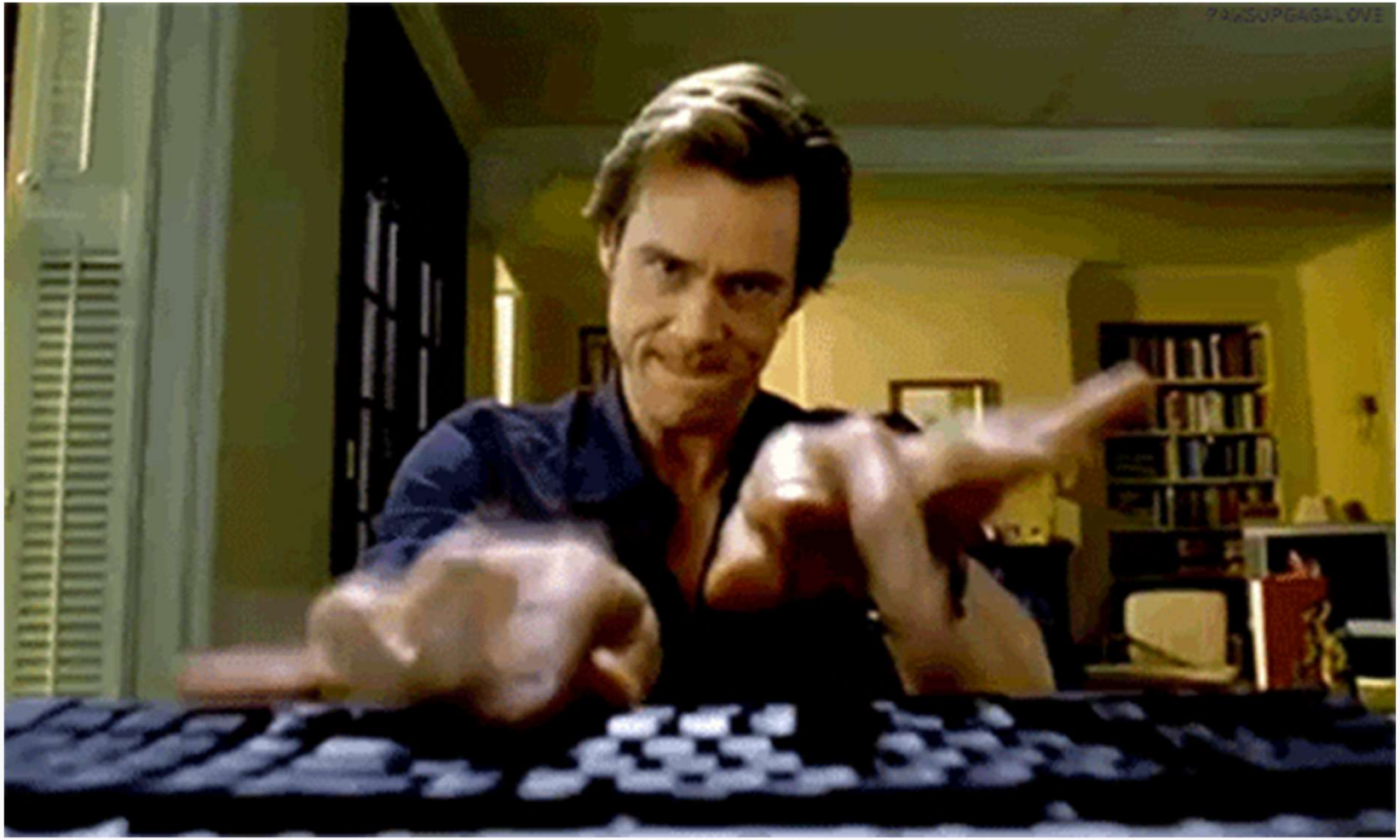
/rupeshtech
/rupeshtech
@rupeshwitter
https://rupesh.blog/

Sc ch/

Sample Application

- BookShop Application
- Built in .NET Core 3.1
- Connects to Azure sql db and displays data
- There are two environments (Dev and Prod)
- There are two app services (miDemo-Dev and miDemo-Prod)
- There are two Databases (one for each env)

Demo



What is problem with this setup

- Developers can see password/secrets
- It stays in repo
- It stays in their laptop

Thru pipeline

Demo

What is problem with this setup

Password is still part of config

Problems related to Passwords/secrets

- Have to
  - Protect
  - Manage
  - Rotate
  - Revoke
- Gives blanket access to resource e.g Azure storage account
- Cannot tie access to single service

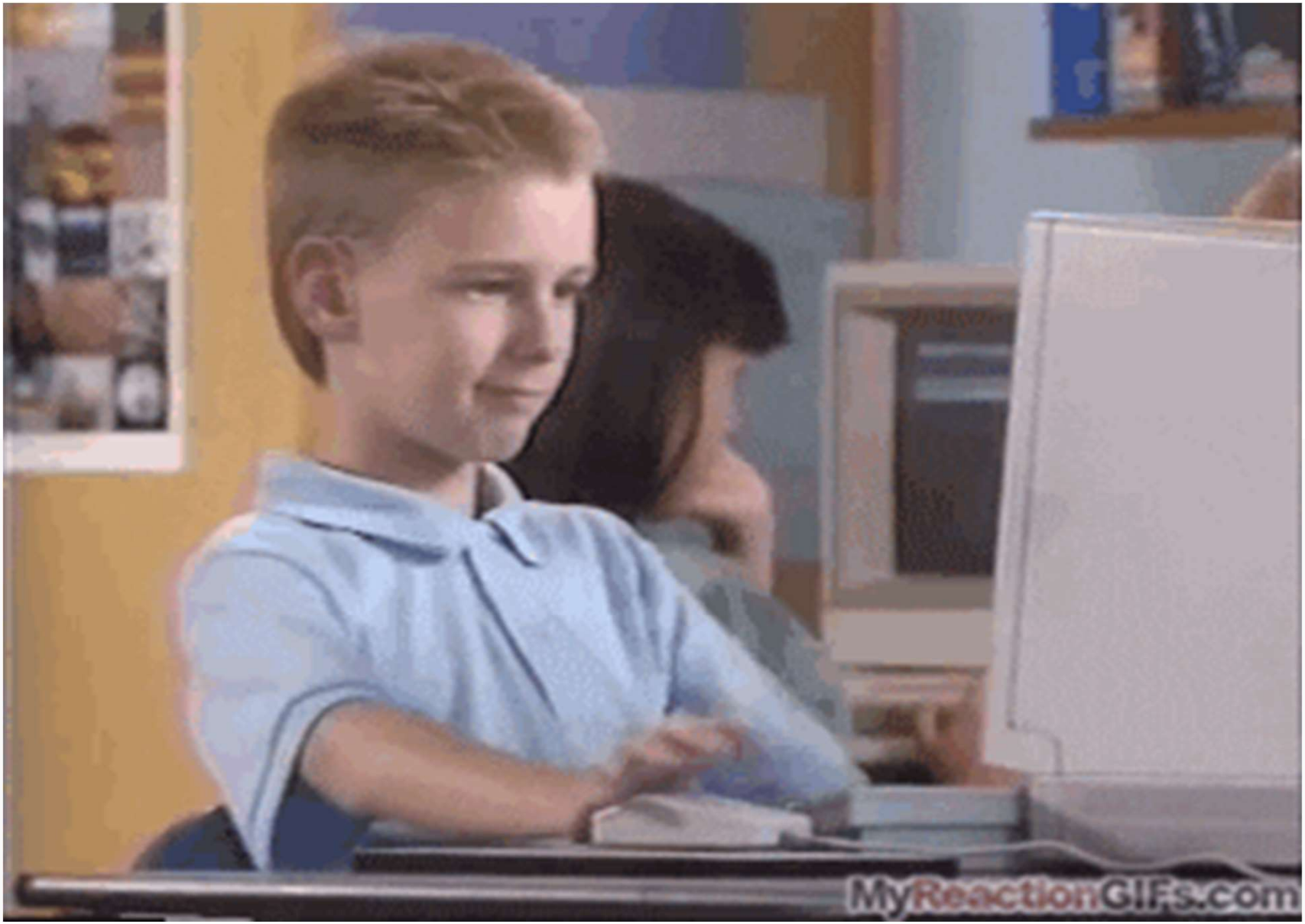
Managed Identity

What is Managed Identity

- It is an Identity, like any other ID and stays in AAD
- It also has UserName(ClientId) and Password(ClientSecret)
- But it is Managed
- Password encryption, key rotation, Life cycle is managed by AAD automatically

MI Features

- Previously known as Managed Service Identity (MSI)
- Free
- Automatic Service Identity Creation
- Key Management and rotation
- Life Cycle Management
- Facilitates Zero cred


MyReactionGIFs.com

Implementation??

Easy or Difficult

Enable MI

Using Portal

Arm template

```
"identity": {
    "type": "SystemAssigned"
}
```

Powershell

```
Set-AzWebApp -AssignIdentity $true -Name $webappname –ResourceGroupName $rgname
```

Code Changes

Types of MI

- There are two types of MI
- System assigned
- User assigned

System Assigned MI?

- Automatically created
- Tied to a service
- Unique to every service
- Deleted When service is deleted

Demo

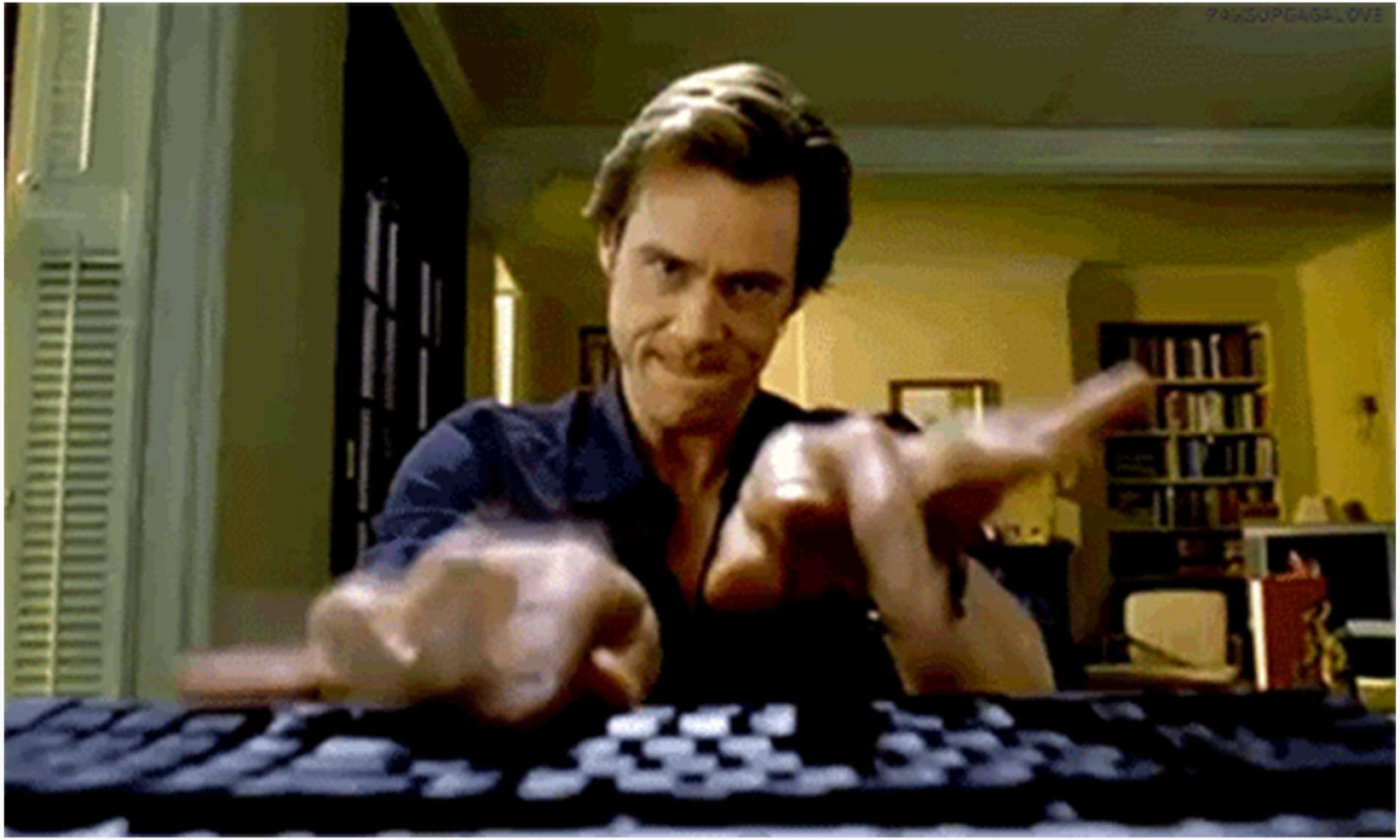
User Assigned MI?

Separate Service

Multiple services can share one identity

One service can be assigned more than one identity

Demo



Benefits of using MI

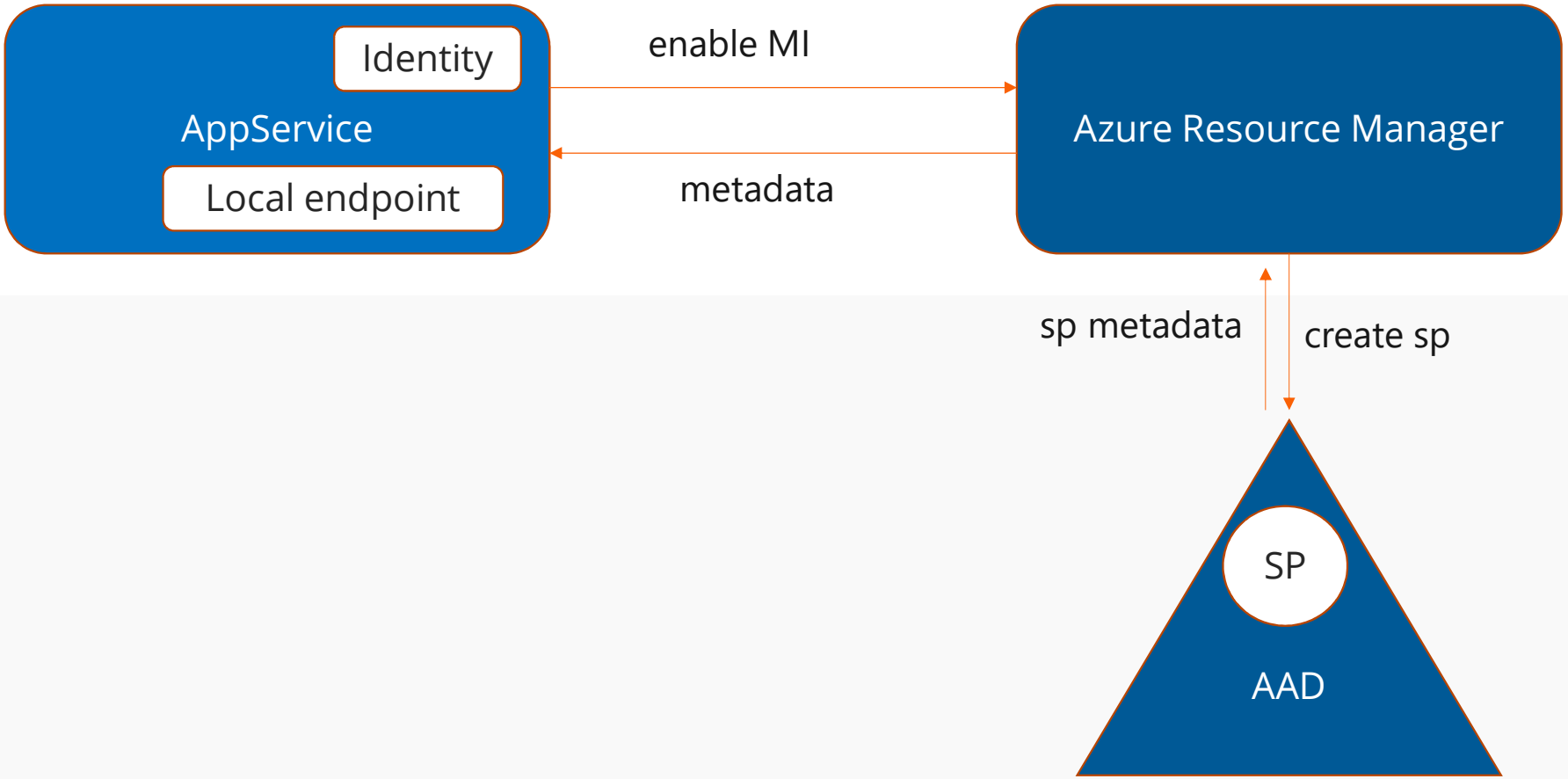
- No Secrets to worry
- No Key rotation to remember
- More granular access
- Auditable
- Revocation can be done quickly

Services that support MI

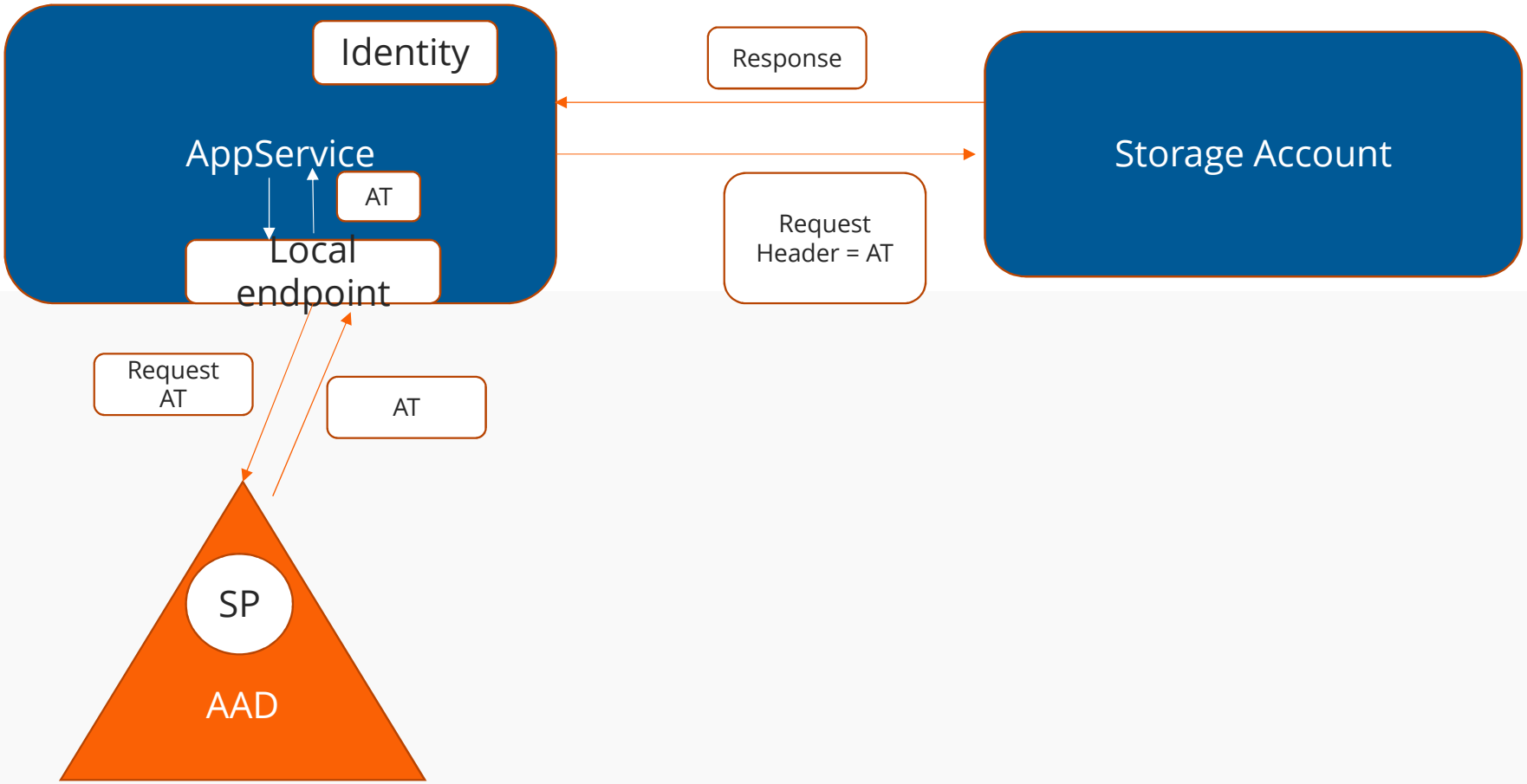
- Virtual Machine
- App Service
- Azure Function
- Data Factory
- Logic Apps
- Api Mangement
- Many More

Source: https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/services-support-managed-identities#azure-services-that-support-azure-ad-authentication

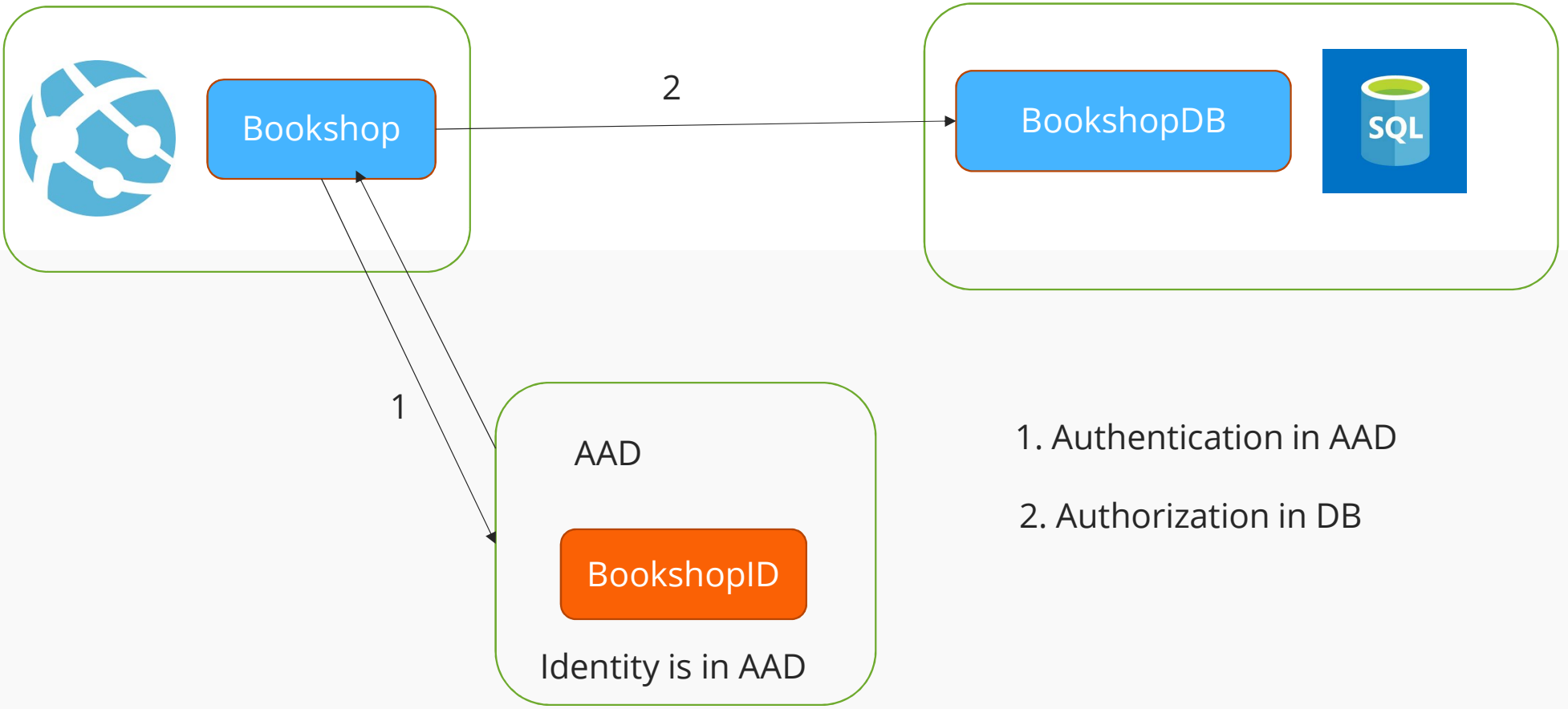
When MI is enabled



How Auth happens



Authentication & Authorization



1. Authentication in AAD
2. Authorization in DB

Services that AD Auth

What if, I have a custom API

What if, The API I use in external

- ARM API
- DataLake
- Azure Sql
- Storage
- ServiceBus
- EventHub
- KeyVault
- Many More.....

Source: https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/services-support-managed-identities#azure-services-that-support-azure-ad-authentication

And finally

- All code, scripts and slide will be shared

- Feedback form (http://bit.do/rupes)

Thanks

Questions?