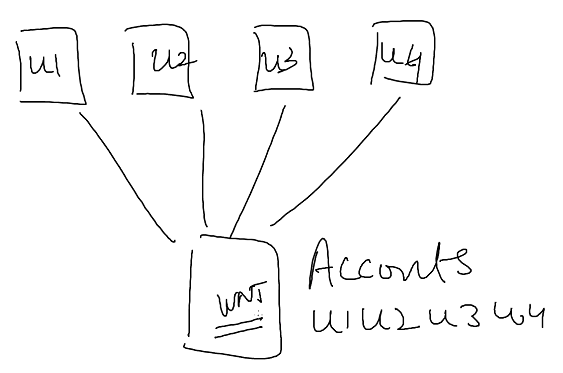
**Securing Azure Web Application**

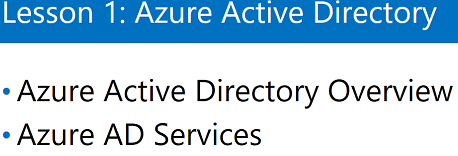
* Active directory

Reason for active directory: security

WindowNT: multiple systems connected with WNT server, each user has its own identity and security and can share information.



AD used for authentication and authorization.



Azure Active Directory Overview:

* Managed identity and access management solution in azure

Focus on managing your domain, users and applications

* It’s a PAAS service
* Rich single sign-on solution : we actually have to sign on once

e.g.

Suppose I have user 1 and he sign on once on a machine and

When this user sign in on two machine, he open an application which supports Single sign – on. This application will sign in with the credentials on which she/she has sign in.

**How single sign-on works in Azure:**

I have an app which in Azure, that app is connected to Azure AD. This App user access to his desktop as soon as he opens the application, application signs him in. AD which is there is using signing on the user using credentials, AD is used for authentication, and AD will check the password from local machine.

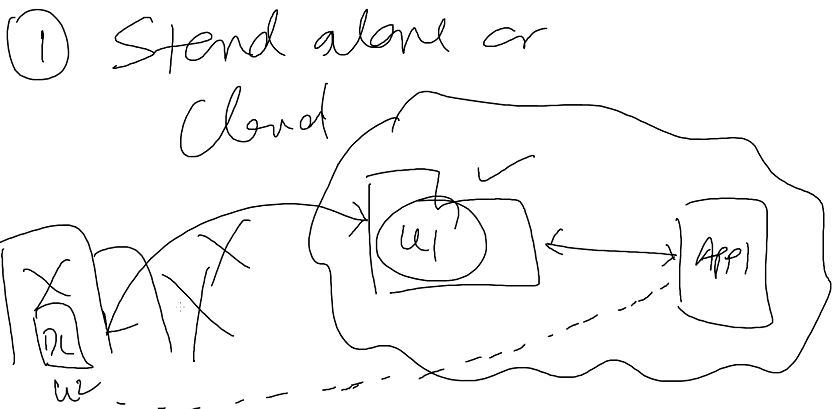
**AD works in three ways in cloud:**

1. Cloud only identity/ stand alone

I have my Active directory with user (u1) and have on prime AD also.

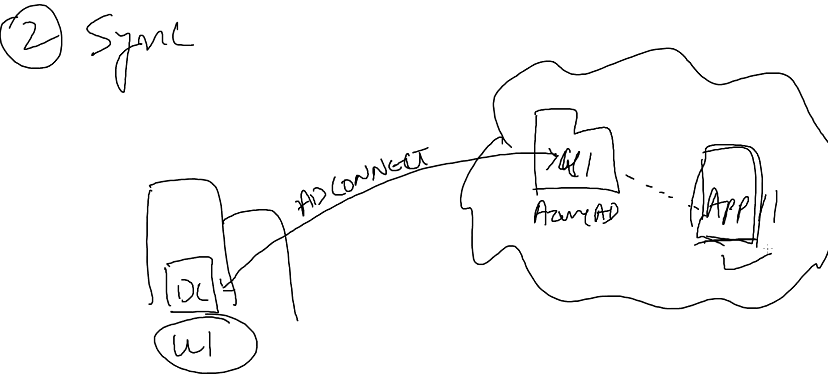
There is no connection between AD in Cloud and on prime AD.

If I have user u1 in Cloud AD and its connected to an Application that user can access that application.



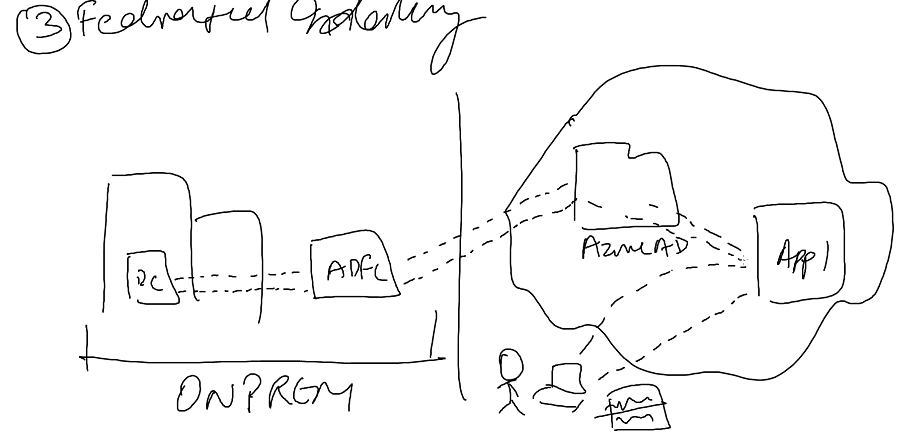
1. Synchronize identity

I have and on prime AD with domain controller, have my Azure AD in cloud. Both Ad is connected with AD connect. If I have user u1 in on prime AD, same user will be synchronize to Azure AD. And If I have an Application in cloud the U1 can login to Application. So U1 is the part of Azure AD.



1. Federated identity

Domain controller and ADFC server, on prime. Cloud Azure AD and Application. I works like synchronization works but authentication works in a way where user u1 try to access the application, as soon as he enters the password, it is sent to Azure AD which instead of validating send to ADFC server. ADFC server send the request to domain controller which in term sends the authorization token which gives the access token.



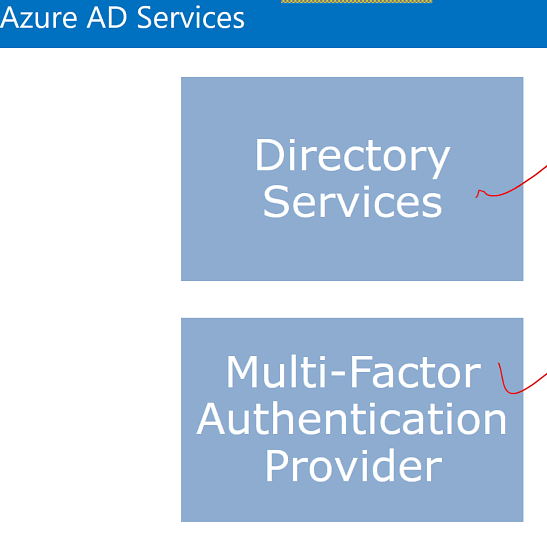
* Supports existing standard protocols

SAML 2.0

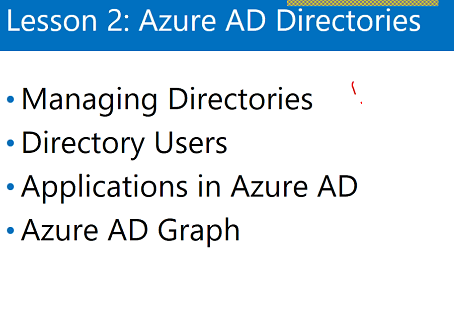
WS-Federation

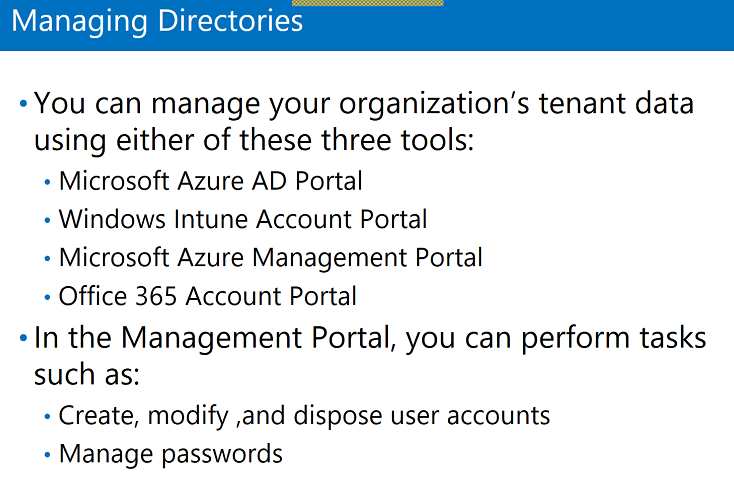
OpenId Connect

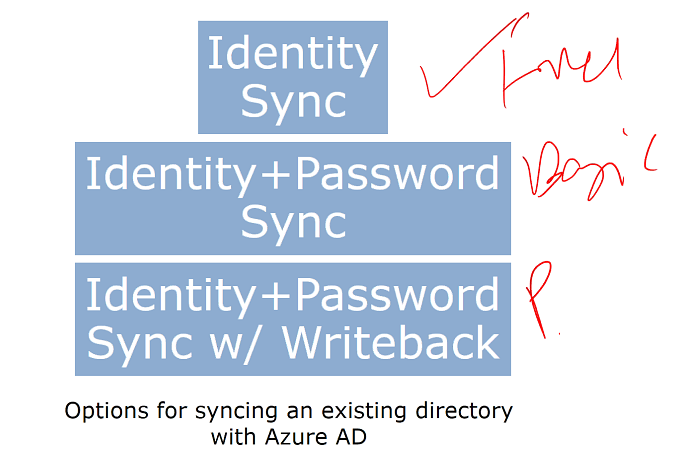
OAuth 2.0

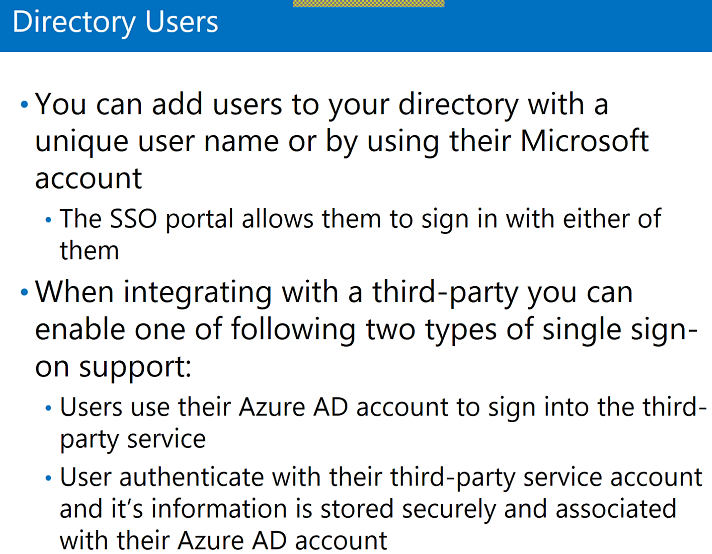


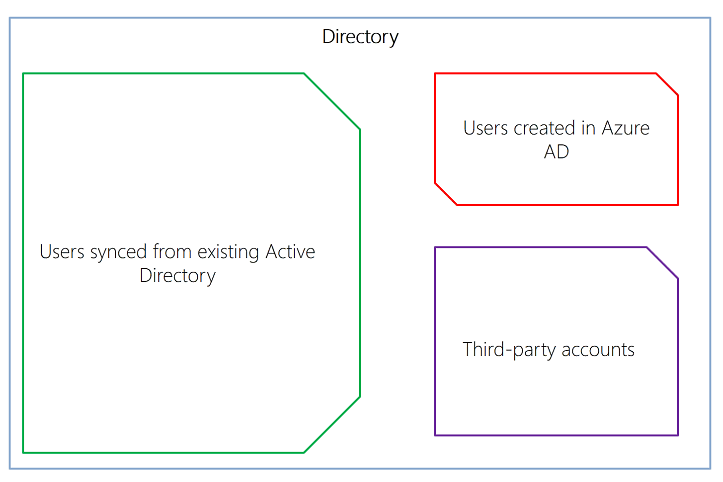
MFA: RSA application. IDEA: this generates a random number and before logining system asked for that random number.

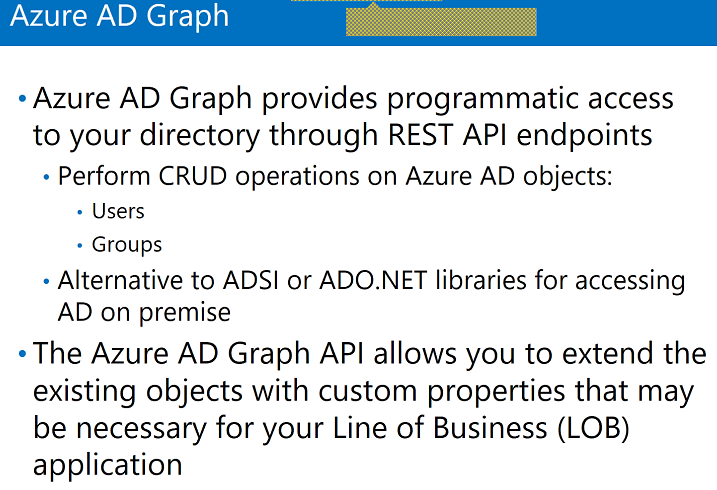




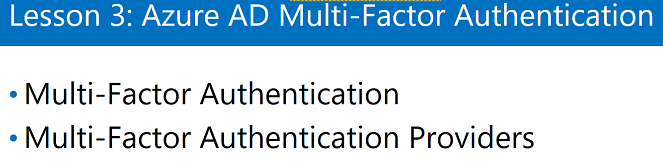




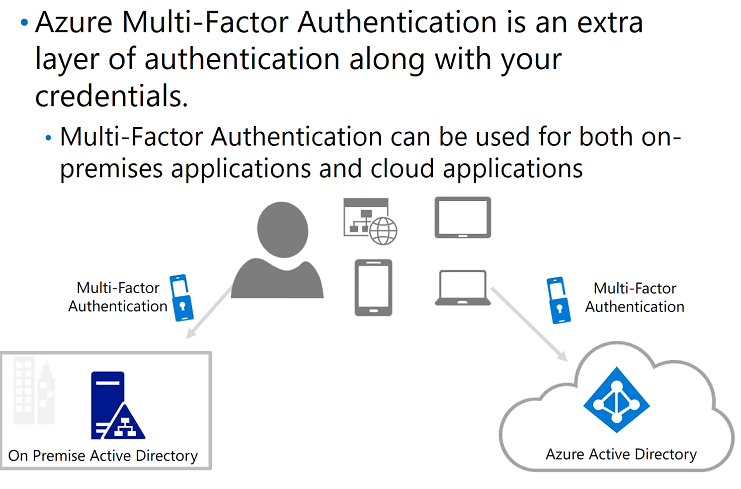


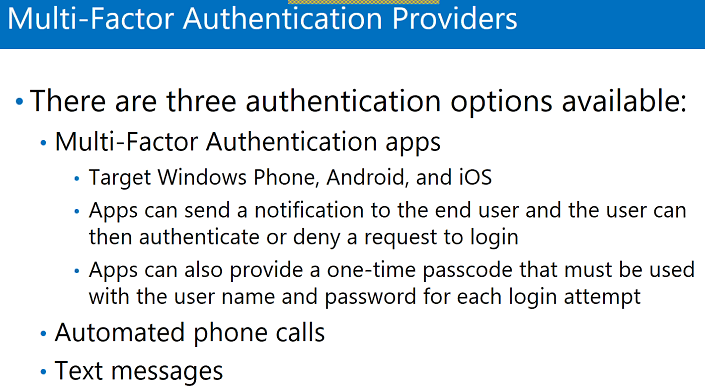


ADSI: tool to control Azure AD.



Multi factor Authentication:





Labs:

1. Create app service

This is the application which is deployed in cloud and we have to add authentication to this app.

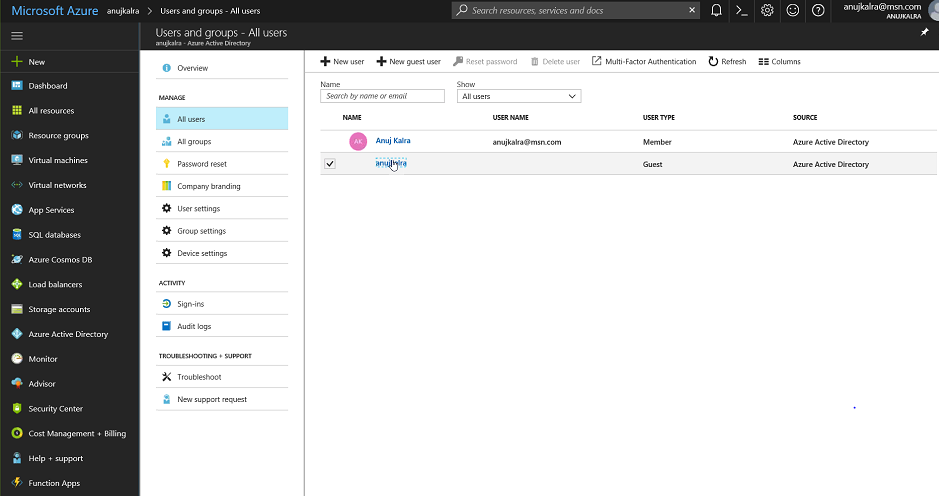
1. Go to authentication and authorization
2. Select AD

Express

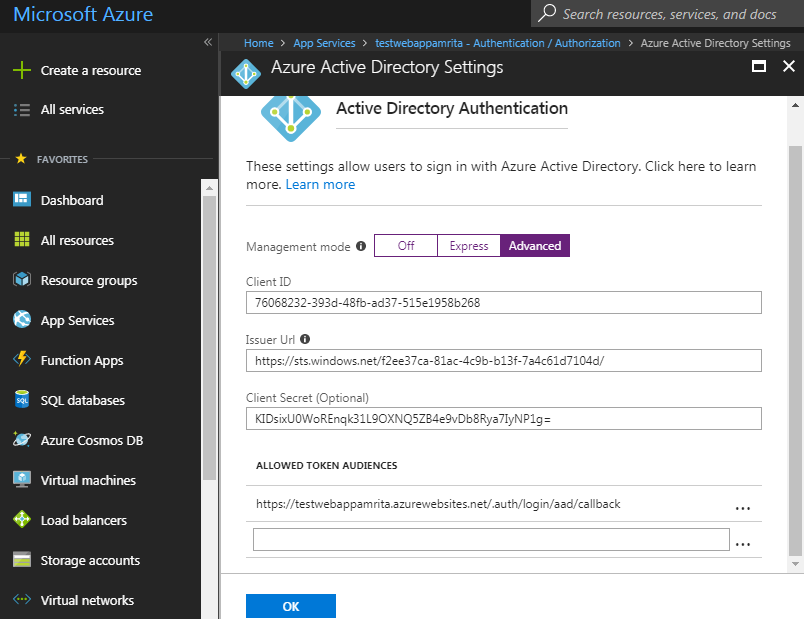
Advance

1. Create
2. Set App service authentication – on
3. Click Active Directory Authentication – Express
4. Create App
5. Ok

Now will create a user account



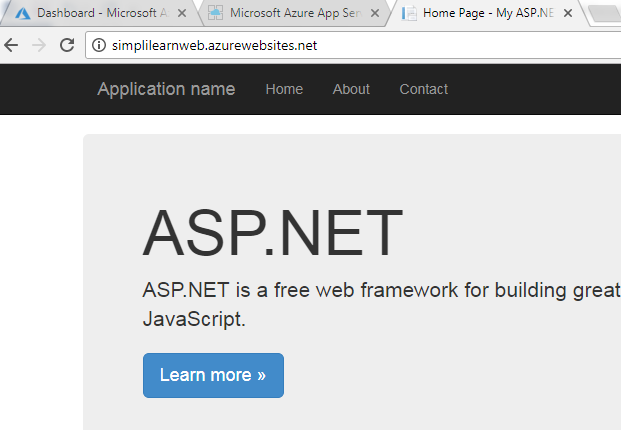
1. Select Azure Active Directory
2. All users
3. Create guest user and activate the user
4. Now browse the application
5. And login using new user
6. Create new Web App
7. To Add app- go to AD
8. Enterprise Application
9. New application
10. Ok take me to new app registration
11. Create new app registration
12. Single sign on url – copy the url of app
13. Check the registered app in Active directory
14. App registrations
15. Add end points
16. Copy any end point and run in browser
17. Now select advance



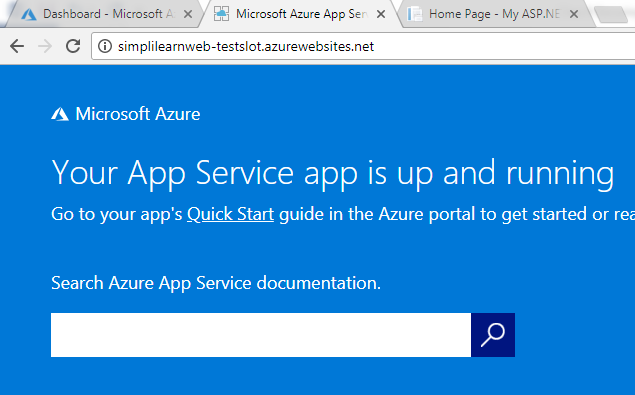
1. Save
2. Open with chrome

**Swap between web Application and its development slot:**

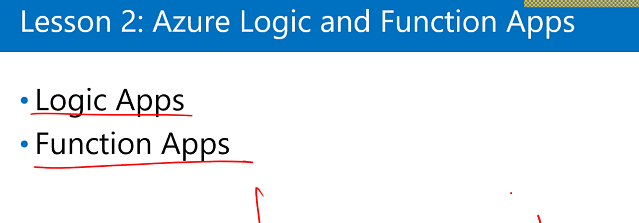
This is web application which is loaded using VS

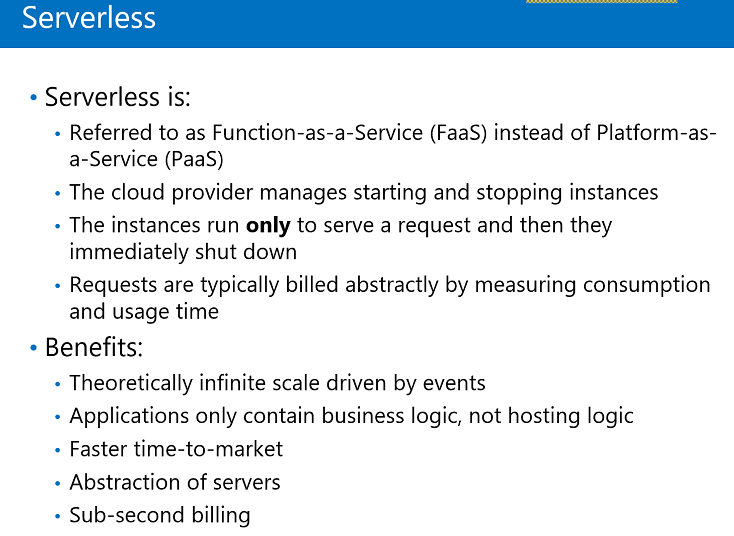


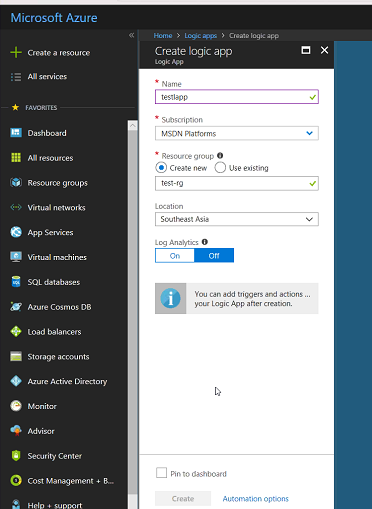
This is Web Development slot



It will just swap the production to slot and slot to production.







Once create we can add triggers, actions, etc.

