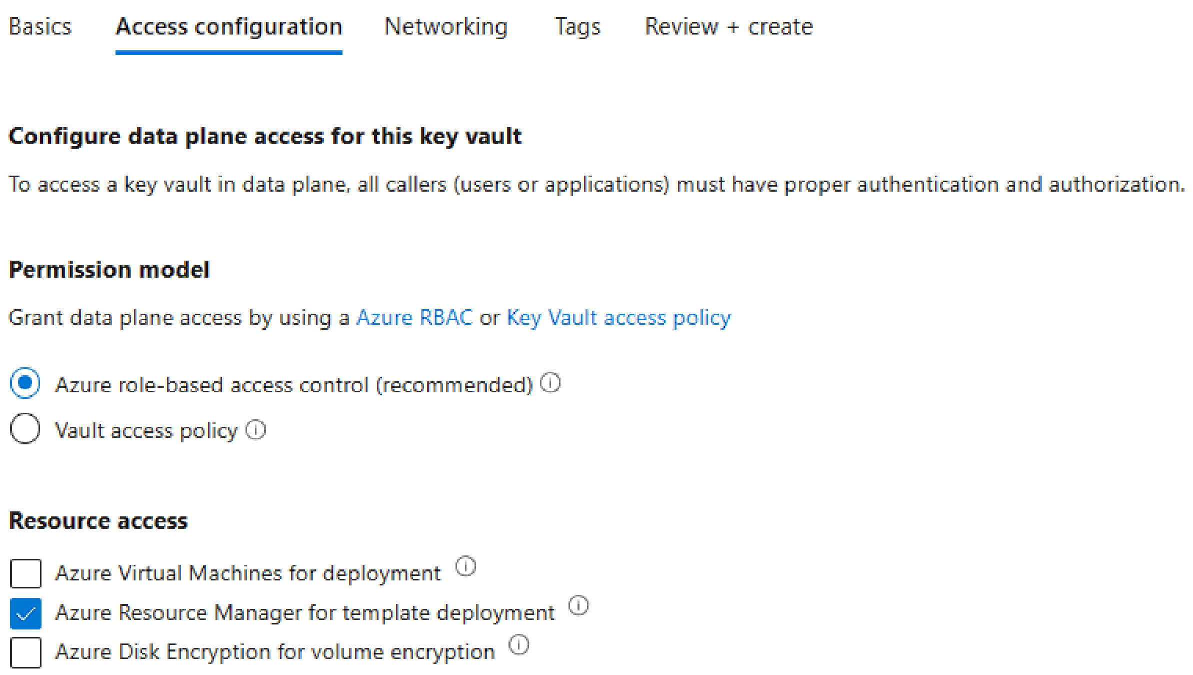
😊 Secure String Key Vault

1. In the previous lab we used the parameters as secure strings to secure our password.
2. Now in this lab we will see another method to secure our password by using the key vault service which we can find in Azure Portal.
3. First in your Azure Portal search for key vaults and create a vault for yourself.
4. Now while creating your Vault remember to turn on **Azure resource manager for template deployment.**
5. After that just create your vault.



Configure data plane access for this key vault

To access a key vault in data plane, all callers (users or applications) must have proper authentication and authorization.

Permission model

Grant data plane access by using a Azure RBAC or Key Vault access policy

● Azure role-based access control (recommended) ⓘ

○ Vault access policy ⓘ

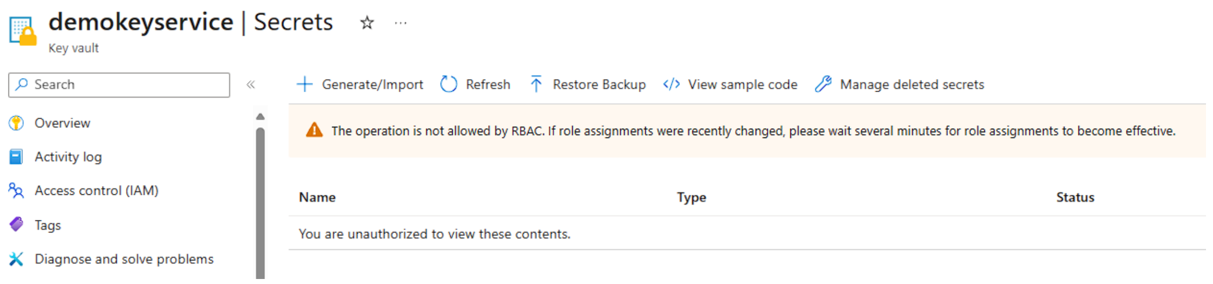
Resource access

☐ Azure Virtual Machines for deployment ⓘ

☑ Azure Resource Manager for template deployment ⓘ

☐ Azure Disk Encryption for volume encryption ⓘ

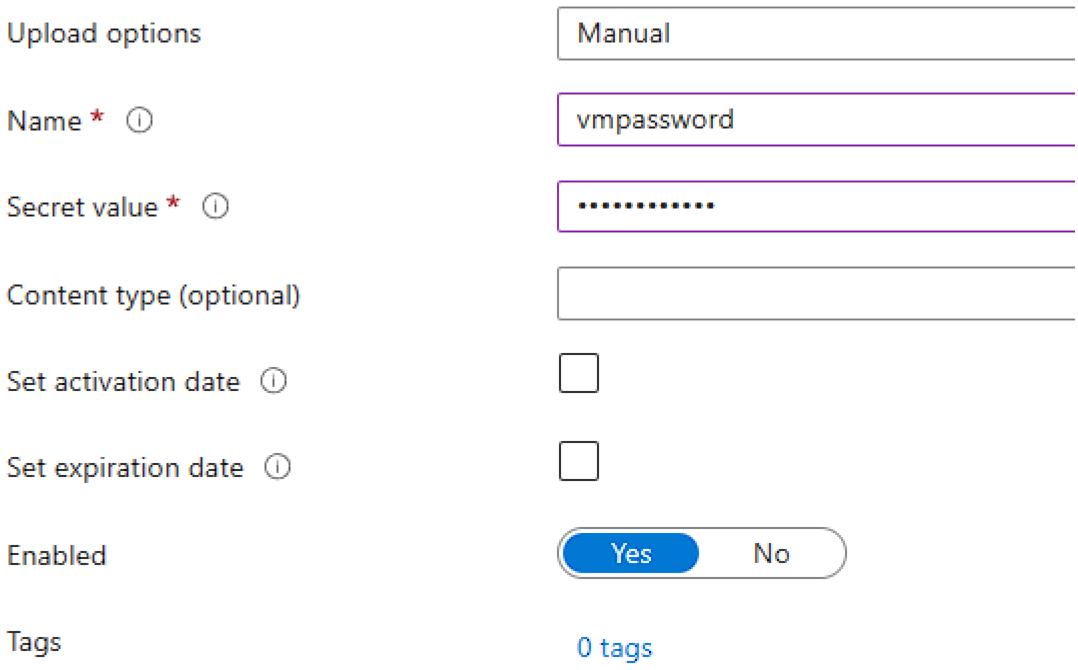
6. Once your deployment is complete click on go to resource.
7. Then from the secrets tab go towards it and generate a new secret.



⚠ The operation is not allowed by RBAC. If role assignments were recently changed, please wait several minutes for role assignments to become effective.

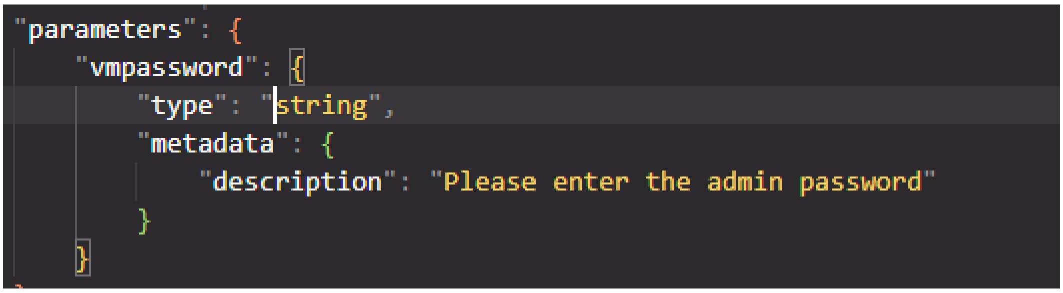
Name	Type	Status
You are unauthorized to view these contents.		

8. Give it a name and password of your choice and then just create your secret.



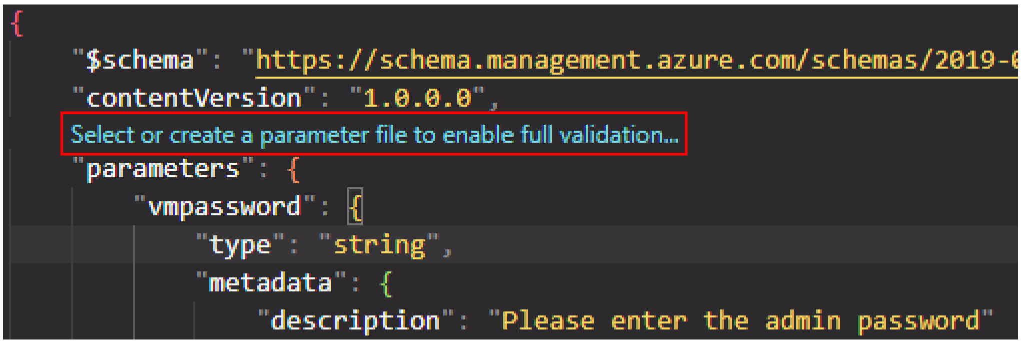
Upload options	Manual
Name *	vmpassword
Secret value *	•••••••••••••
Content type (optional)	
Set activation date	□
Set expiration date	□
Enabled	[✓ Yes] No
Tags	0 tags

9. Now come back to VS Code and create a new file for this lab.
10. Then copy the code from the previous lab.
11. There is a small change instead of secure string this time we will use only string.



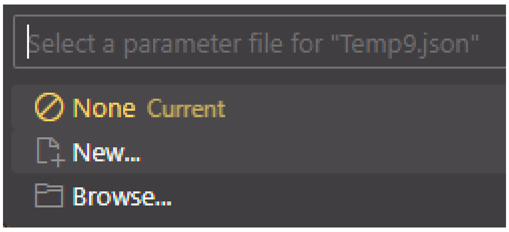
```
"parameters": {
    "vmpassword": {
        "type": "string",
        "metadata": {
            "description": "Please enter the admin password"
        }
    }
```

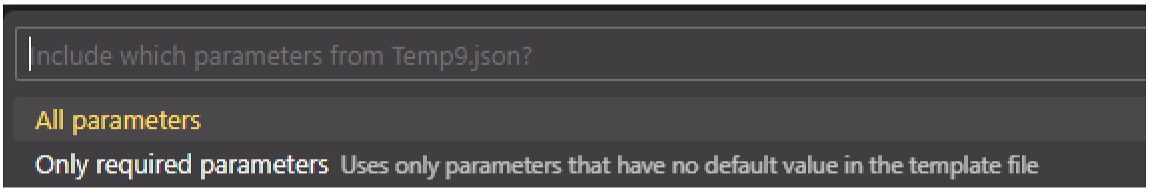
12. The only thing now is we need to pass the value. So, instead of us entering a value, the template needs to pick up the value from the Azure Key vault.
13. For this, we need to again create a parameter file. I'll click here in Visual Studio code to create a new parameter file.



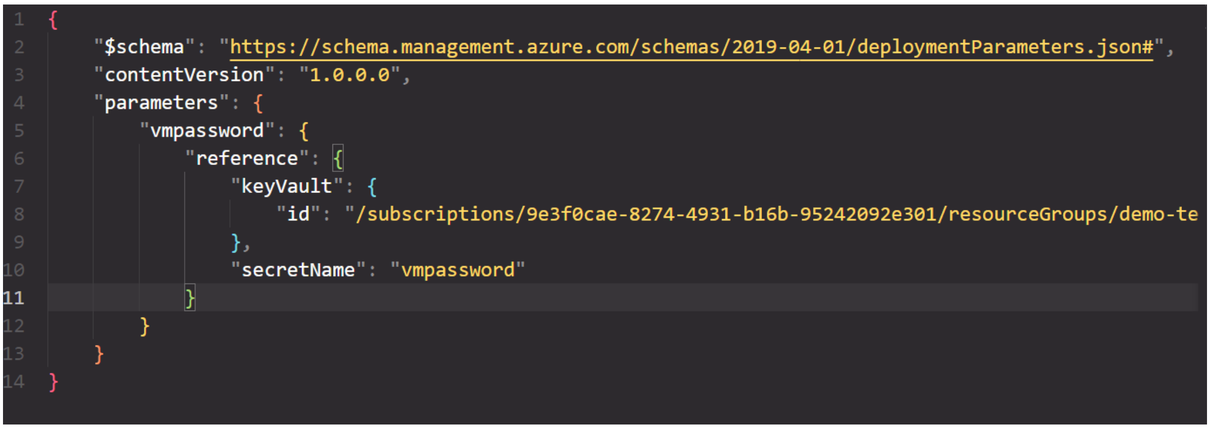
```
{
    "$schema": "https://schema.management.azure.com/schemas/2019-
    "contentVersion": "1.0.0.0",
    Select or create a parameter file to enable full validation...
    "parameters": {
        "vmpassword": {
            "type": "string",
            "metadata": {
                "description": "Please enter the admin password"
```

14. Then you have to click on new after that choose all parameters then click on save.



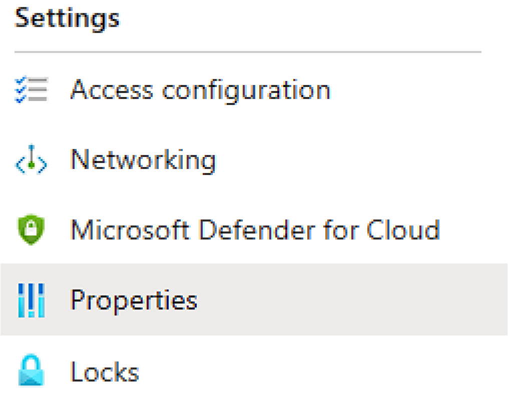

Include which parameters from Temp9.json?
All parameters
Only required parameters Uses only parameters that have no default value in the template file

15. Below you can see in the parameter file we have defined a password which is referring to our key vault and we have mentioned its ID.

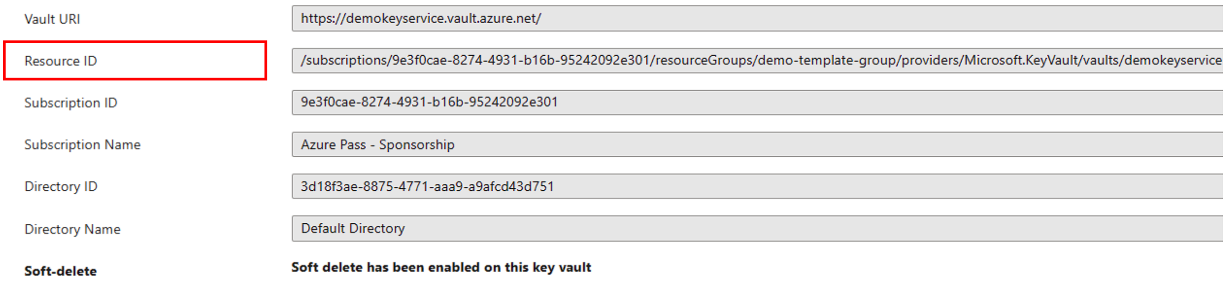


```
{
    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentParameters.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
        "vmpassword": {
            "reference": {
                "keyVault": {
                    "id": "/subscriptions/9e3f0cae-8274-4931-b16b-95242092e301/resourceGroups/demo-te
                },
                "secretName": "vmpassword"
            }
        }
    }
}
```

16. Now to get your key vault ID, from your left pane choose properties.



17. Then choose the resource ID.

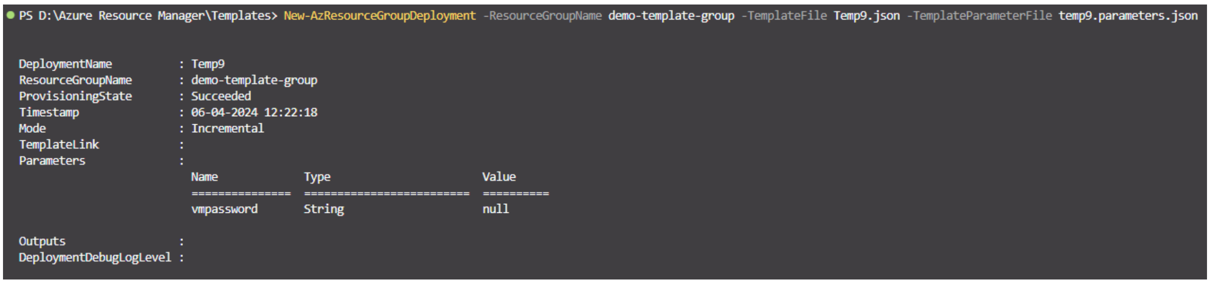


Vault URI	https://demokeyservice.vault.azure.net/
Resource ID	/subscriptions/9e3f0cae-8274-4931-b16b-95242092e301/resourceGroups/demo-template-group/providers/Microsoft.KeyVault/vaults/demokeyservice
Subscription ID	9e3f0cae-8274-4931-b16b-95242092e301
Subscription Name	Azure Pass - Sponsorship
Directory ID	3d18f3ae-8875-4771-aaa9-a9afcd43d751
Directory Name	Default Directory
Soft-delete	**Soft delete has been enabled on this key vault**

18. Once all this is done now you are going to deploy this template using power shell.
19. Click on terminal and choose new terminal and there you need to run this command mentioned below.

    New-AzResourceGroupDeployment -ResourceGroupName demo-template-group -TemplateFile Temp9.json -TemplateParameterFile temp9.parameters.json

20. Now after some time you can see that the deployment was successful.

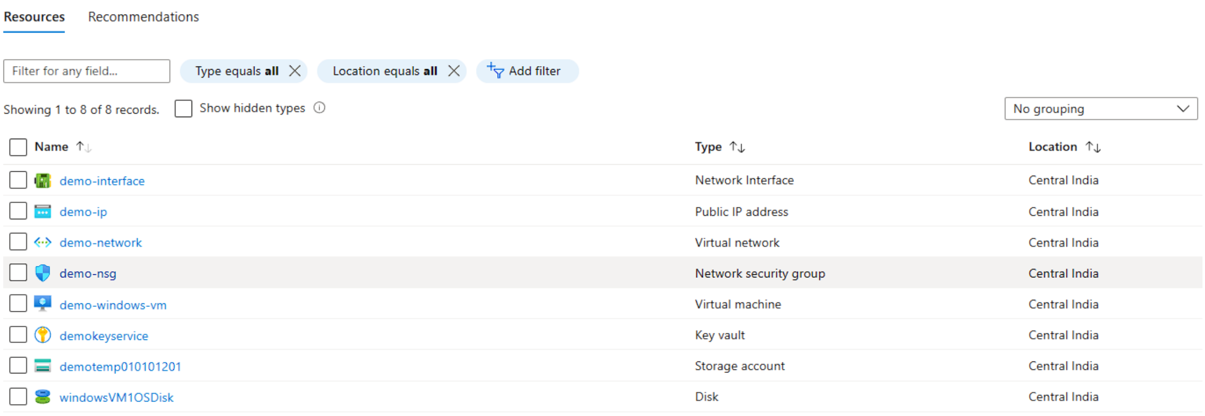


```
PS D:\Azure Resource Manager\Templates> New-AzResourceGroupDeployment -ResourceGroupName demo-template-group -TemplateFile Temp9.json -TemplateParameterFile temp9.parameters.json

DeploymentName          : Temp9
ResourceGroupName       : demo-template-group
ProvisioningState       : Succeeded
Timestamp               : 06-04-2024 12:22:18
Mode                    : Incremental
TemplateLink            :
Parameters              :
                          Name             Type                       Value
                          ===============  =========================  ==========
                          vmpassword       String                     null

Outputs                 :
DeploymentDebugLogLevel :
```

21. Now you have to go back to Portal and check for your resources and there you can see everything.



Showing 1 to 8 of 8 records. □ Show hidden types

□ Name ↑↓	Type ↑↓	Location ↑↓
□ demo-interface	Network Interface	Central India
□ demo-ip	Public IP address	Central India
□ demo-network	Virtual network	Central India
□ demo-nsg	Network security group	Central India
□ demo-windows-vm	Virtual machine	Central India
□ demokeyservice	Key vault	Central India
□ demotemp010101201	Storage account	Central India
□ windowsVM1OSDisk	Disk	Central India

22. Now you can try to login to your VM and there you will see that everything executed successfully.