CREATING AZURE POLICY COMPLIANT BACKDOOR

Viktor Gazdag | Cloud Village | DefCon 32

WHOAMI

- Principal Security Consultant NCC Group
- Cloud and CI/CD Security
- Certs: AZ-104, AZ-400, AZ-500, SC-100, CARTP, CAWASP
- Black Hat USA 2022, DefCon31 Cloud Village, DoD CyberDT XSWG #15

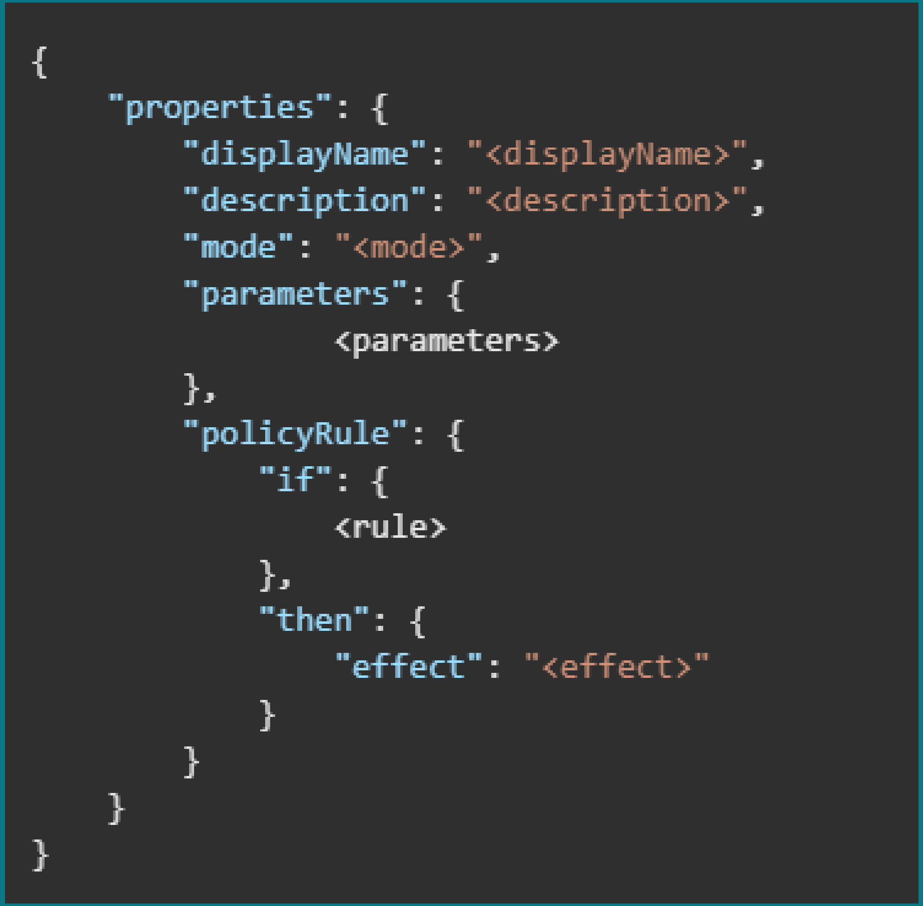
AZURE POLICY

- "Azure Policy evaluates resources and actions in Azure by comparing the properties of those resources to business rules." - Microsoft
- Business rules = I.e.: Resources Only in European Regions

AZURE POLICY

- Policy Definition – JSON
- Policy Initiative
- Assigned to scope (Mgmt group, Subscription, Resource group, Resource)
- Remediation
- Policy Initiative(Pol Defs, Rem)->Scope

AZURE POLICY

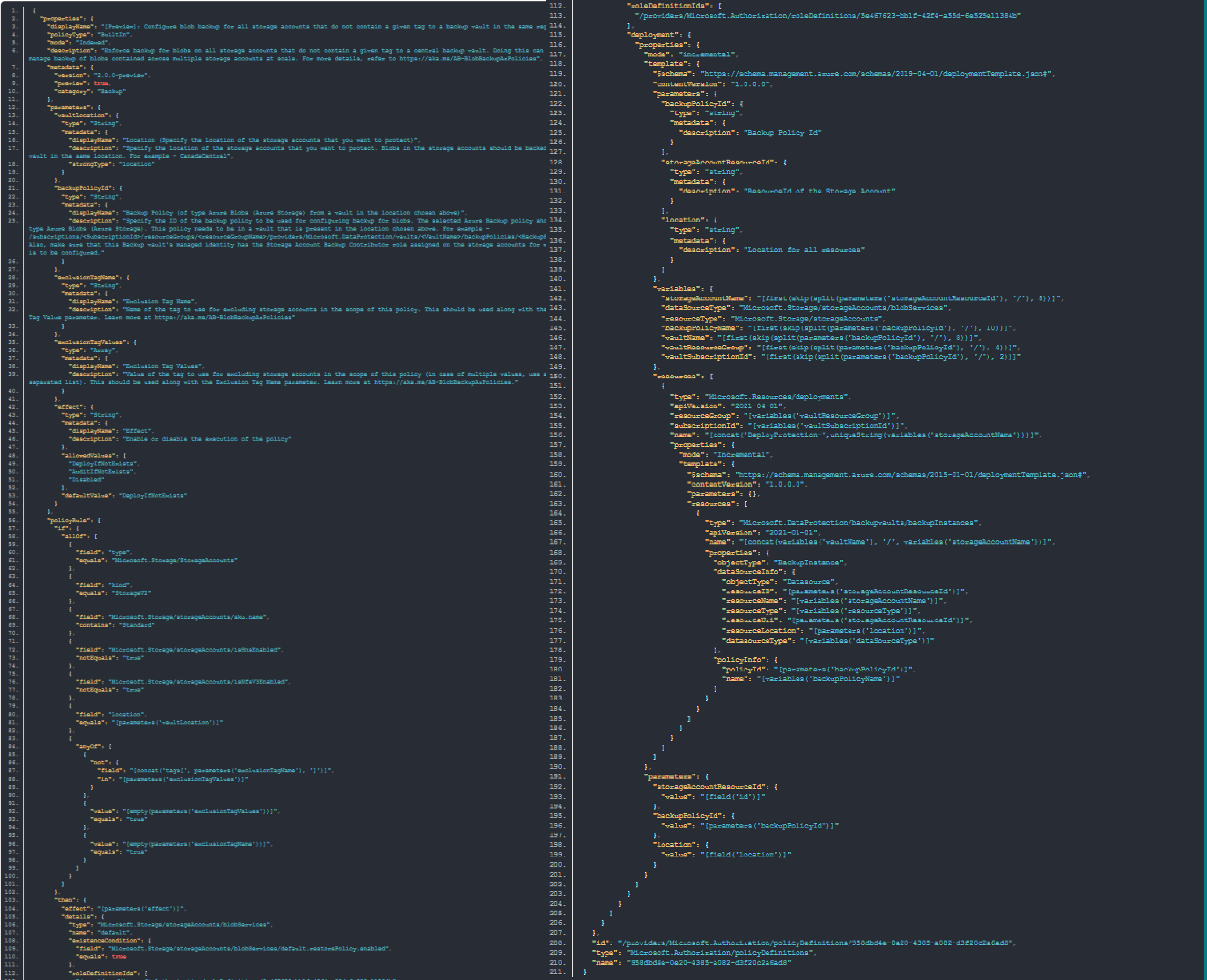


```
{
    "properties": {
        "displayName": "<displayName>",
        "description": "<description>",
        "mode": "<mode>",
        "parameters": {
                <parameters>
        },
        "policyRule": {
            "if": {
                <rule>
            },
            "then": {
                "effect": "<effect>"
            }
        }
    }
}
```

ELEMENTS OF DEFINITION

- displayName
- description
- mode

- parameters
- policyRule
  - logical evaluations
  - effect



```
{
  "properties": {
    "displayName": "[Preview]: Configure blob backup for all storage accounts that do not contain a given tag to a backup vault in the same reg
    "policyType": "BuiltIn",
    "mode": "Indexed",
    "description": "Enforce backup for blobs on all storage accounts that do not contain a given tag to a central backup vault. Doing this can
manage backup of blobs contained across multiple storage accounts at scale. For more details, refer to https://aka.ms/AB-BlobBackupAzPolicies",
    "metadata": {
      "version": "2.0.0-preview",
      "preview": true,
      "category": "Backup"
    },
    "parameters": {
      "vaultLocation": {
        "type": "String",
        "metadata": {
          "displayName": "Location (Specify the location of the storage accounts that you want to protect)",
          "description": "Specify the location of the storage accounts that you want to protect. Blobs in the storage accounts should be backed
vault in the same location. For example - CanadaCentral",
          "strongType": "location"
        }
      },
      "backupPolicyId": {
        "type": "String",
        "metadata": {
          "displayName": "Backup Policy (of type Azure Blobs (Azure Storage) from a vault in the location chosen above)",
          "description": "Specify the ID of the backup policy to be used for configuring backup for blobs. The selected Azure Backup policy sho
type Azure Blobs (Azure Storage). This policy needs to be in a vault that is present in the location chosen above. For example -
/subscriptions/<SubscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.DataProtection/vaults/<VaultName>/backupPolicies/<BackupP
Also, make sure that this Backup vault's managed identity has the Storage Account Backup Contributor role assigned on the storage accounts for w
is to be configured."
        }
      },
      "exclusionTagName": {
        "type": "String",
        "metadata": {
          "displayName": "Exclusion Tag Name",
          "description": "Name of the tag to use for excluding storage accounts in the scope of this policy. This should be used along with the
Tag Value parameter. Learn more at https://aka.ms/AB-BlobBackupAzPolicies"
        }
      },
      "exclusionTagValues": {
        "type": "Array",
        "metadata": {
          "displayName": "Exclusion Tag Values",
          "description": "Value of the tag to use for excluding storage accounts in the scope of this policy (in case of multiple values, use a
separated list). This should be used along with the Exclusion Tag Name parameter. Learn more at https://aka.ms/AB-BlobBackupAzPolicies."
        }
      },
      "effect": {
        "type": "String",
        "metadata": {
          "displayName": "Effect",
          "description": "Enable or disable the execution of the policy"
        },
        "allowedValues": [
          "DeployIfNotExists",
          "AuditIfNotExists",
          "Disabled"
        ],
        "defaultValue": "DeployIfNotExists"
      }
    },
    "policyRule": {
      "if": {
        "allOf": [
          {
            "field": "type",
            "equals": "Microsoft.Storage/StorageAccounts"
          },
          {
            "field": "kind",
            "equals": "StorageV2"
          },
          {
            "field": "Microsoft.Storage/storageAccounts/sku.name",
            "contains": "Standard"
          },
          {
            "field": "Microsoft.Storage/storageAccounts/isHnsEnabled",
            "notEquals": "true"
          },
          {
            "field": "Microsoft.Storage/storageAccounts/isNfsV3Enabled",
            "notEquals": "true"
          },
          {
            "field": "location",
            "equals": "[parameters('vaultLocation')]"
          },
          {
            "anyOf": [
              {
                "not": {
                  "field": "[concat('tags[', parameters('exclusionTagName'), ']')]",
                  "in": "[parameters('exclusionTagValues')]"
                }
              },
              {
                "value": "[empty(parameters('exclusionTagValues'))]",
                "equals": "true"
              },
              {
                "value": "[empty(parameters('exclusionTagName'))]",
                "equals": "true"
              }
            ]
          }
        ]
      },
      "then": {
        "effect": "[parameters('effect')]",
        "details": {
          "type": "Microsoft.Storage/storageAccounts/blobServices",
          "name": "default",
          "existenceCondition": {
            "field": "Microsoft.Storage/storageAccounts/blobServices/default.restorePolicy.enabled",
            "equals": true
          },
          "roleDefinitionIds": [
            "/providers/Microsoft.Authorization/roleDefinitions/5e467623-bb1f-42f4-a55d-6e525e11384b"
          ],
          "deployment": {
            "properties": {
              "mode": "incremental",
              "template": {
                "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
                "contentVersion": "1.0.0.0",
                "parameters": {
                  "backupPolicyId": {
                    "type": "string",
                    "metadata": {
                      "description": "Backup Policy Id"
                    }
                  },
                  "storageAccountResourceId": {
                    "type": "string",
                    "metadata": {
                      "description": "ResourceId of the Storage Account"
                    }
                  },
                  "location": {
                    "type": "string",
                    "metadata": {
                      "description": "Location for all resources"
                    }
                  }
                },
                "variables": {
                  "storageAccountName": "[first(skip(split(parameters('storageAccountResourceId'), '/'), 8))]",
                  "dataSourceType": "Microsoft.Storage/storageAccounts/blobServices",
                  "resourceType": "Microsoft.Storage/storageAccounts",
                  "backupPolicyName": "[first(skip(split(parameters('backupPolicyId'), '/'), 10))]",
                  "vaultName": "[first(skip(split(parameters('backupPolicyId'), '/'), 8))]",
                  "vaultResourceGroup": "[first(skip(split(parameters('backupPolicyId'), '/'), 4))]",
                  "vaultSubscriptionId": "[first(skip(split(parameters('backupPolicyId'), '/'), 2))]"
                },
                "resources": [
                  {
                    "type": "Microsoft.Resources/deployments",
                    "apiVersion": "2021-04-01",
                    "resourceGroup": "[variables('vaultResourceGroup')]",
                    "subscriptionId": "[variables('vaultSubscriptionId')]",
                    "name": "[concat('DeployProtection-',uniqueString(variables('storageAccountName')))]",
                    "properties": {
                      "mode": "Incremental",
                      "template": {
                        "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
                        "contentVersion": "1.0.0.0",
                        "parameters": {},
                        "resources": [
                          {
                            "type": "Microsoft.DataProtection/backupvaults/backupInstances",
                            "apiVersion": "2021-01-01",
                            "name": "[concat(variables('vaultName'), '/', variables('storageAccountName'))]",
                            "properties": {
                              "objectType": "BackupInstance",
                              "dataSourceInfo": {
                                "objectType": "Datasource",
                                "resourceID": "[parameters('storageAccountResourceId')]",
                                "resourceName": "[variables('storageAccountName')]",
                                "resourceType": "[variables('resourceType')]",
                                "resourceUri": "[parameters('storageAccountResourceId')]",
                                "resourceLocation": "[parameters('location')]",
                                "datasourceType": "[variables('dataSourceType')]"
                              },
                              "policyInfo": {
                                "policyId": "[parameters('backupPolicyId')]",
                                "name": "[variables('backupPolicyName')]"
                              }
                            }
                          }
                        ]
                      }
                    }
                  }
                ]
              },
              "parameters": {
                "storageAccountResourceId": {
                  "value": "[field('id')]"
                },
                "backupPolicyId": {
                  "value": "[parameters('backupPolicyId')]"
                },
                "location": {
                  "value": "[field('location')]"
                }
              }
            }
          }
        }
      }
    }
  },
  "id": "/providers/Microsoft.Authorization/policyDefinitions/958dbd4e-0e20-4385-a082-d3f20c2a6ad8",
  "type": "Microsoft.Authorization/policyDefinitions",
  "name": "958dbd4e-0e20-4385-a082-d3f20c2a6ad8"
}
```

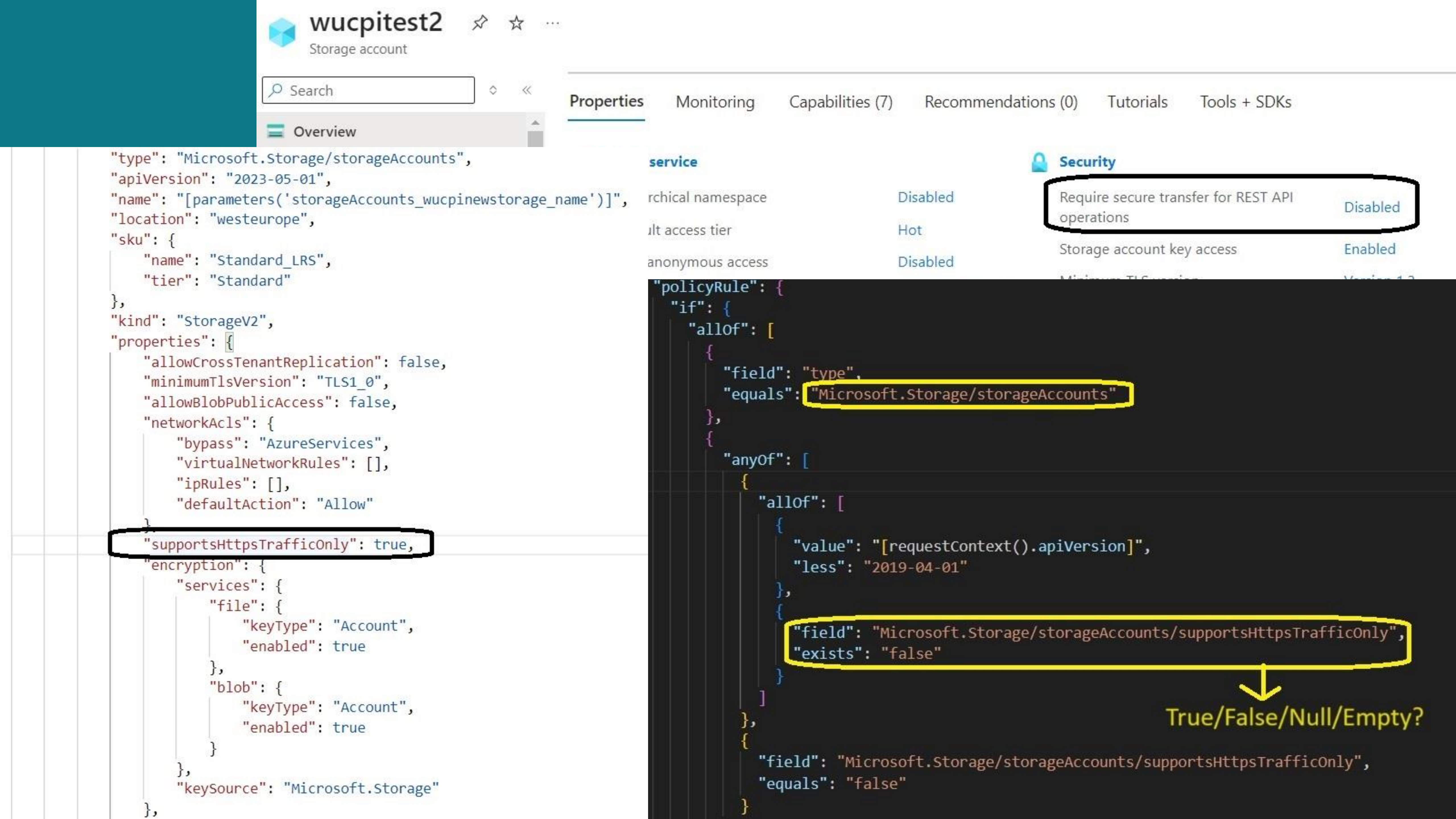
PROPERTIES & ALIAS

- Policy Alias and Resource Property and Modifiable

- Resourcetype

PROPERTIES & ALIAS

- Get-AZPolicyAlias
- AZ CLI
- ARM Template
- Deployment Template
- Azure Resource Explorer



service

rchical namespace	Disabled
ılt access tier	Hot
anonymous access	Disabled

Security

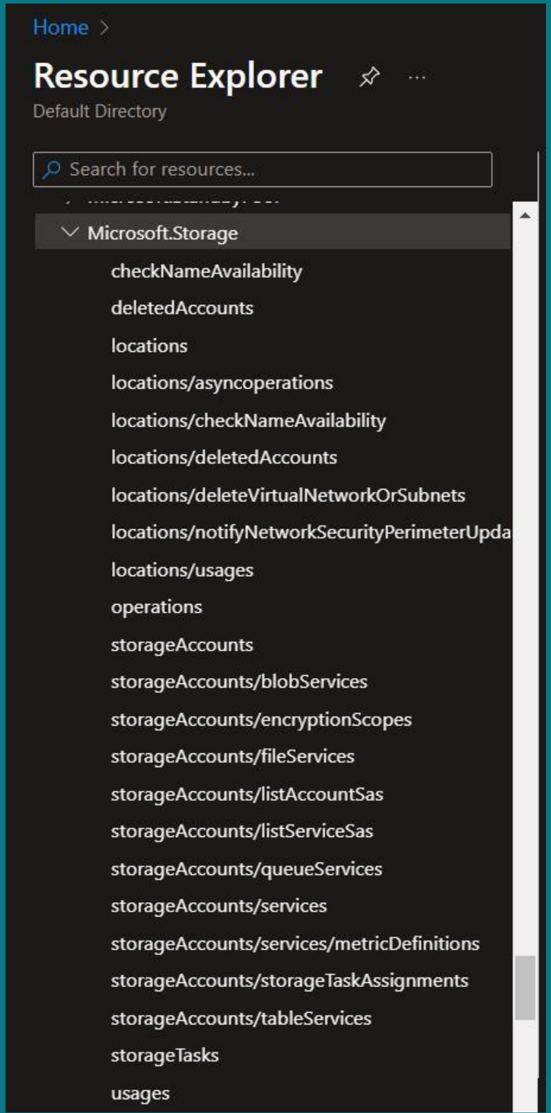
Require secure transfer for REST API operations	Disabled
Storage account key access	Enabled

```
"type": "Microsoft.Storage/storageAccounts",
"apiVersion": "2023-05-01",
"name": "[parameters('storageAccounts_wucpinewstorage_name')]",
"location": "westeurope",
"sku": {
    "name": "Standard_LRS",
    "tier": "Standard"
},
"kind": "StorageV2",
"properties": {
    "allowCrossTenantReplication": false,
    "minimumTlsVersion": "TLS1_0",
    "allowBlobPublicAccess": false,
    "networkAcls": {
        "bypass": "AzureServices",
        "virtualNetworkRules": [],
        "ipRules": [],
        "defaultAction": "Allow"
    },
    "supportsHttpsTrafficOnly": true,
    "encryption": {
        "services": {
            "file": {
                "keyType": "Account",
                "enabled": true
            },
            "blob": {
                "keyType": "Account",
                "enabled": true
            }
        },
        "keySource": "Microsoft.Storage"
    },
```

```
"policyRule": {
  "if": {
    "allOf": [
      {
        "field": "type",
        "equals": "Microsoft.Storage/storageAccounts"
      },
      {
        "anyOf": [
          {
            "allOf": [
              {
                "value": "[requestContext().apiVersion]",
                "less": "2019-04-01"
              },
              {
                "field": "Microsoft.Storage/storageAccounts/supportsHttpsTrafficOnly",
                "exists": "false"
              }
            ]
          },
          {
            "field": "Microsoft.Storage/storageAccounts/supportsHttpsTrafficOnly",
            "equals": "false"
          }
```

True/False/Null/Empty?

PROPERTIES & ALIAS



Home >

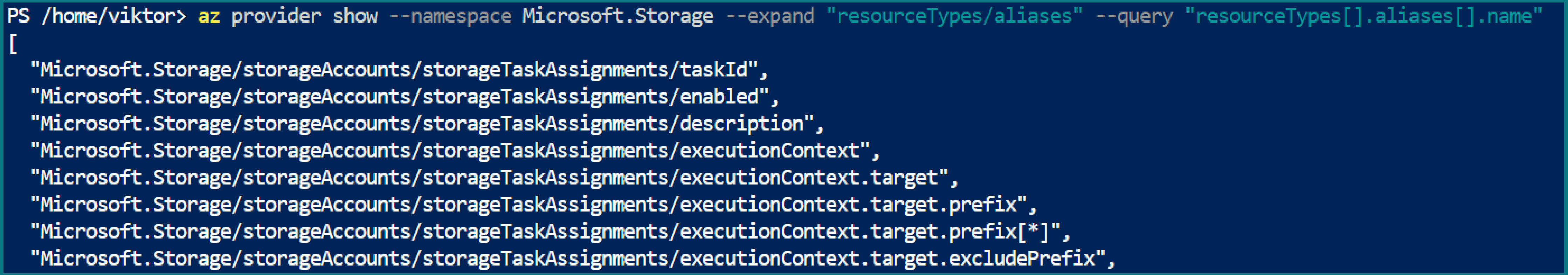
Resource Explorer

Default Directory

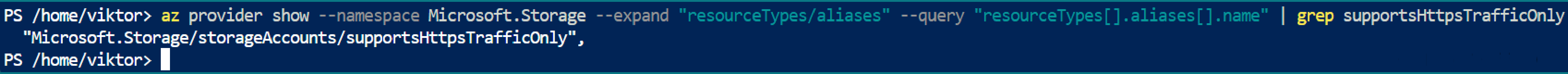
Search for resources...

- Microsoft.Storage
  - checkNameAvailability
  - deletedAccounts
  - locations
  - locations/asyncoperations
  - locations/checkNameAvailability
  - locations/deletedAccounts
  - locations/deleteVirtualNetworkOrSubnets
  - locations/notifyNetworkSecurityPerimeterUpda
  - locations/usages
  - operations
  - storageAccounts
  - storageAccounts/blobServices
  - storageAccounts/encryptionScopes
  - storageAccounts/fileServices
  - storageAccounts/listAccountSas
  - storageAccounts/listServiceSas
  - storageAccounts/queueServices
  - storageAccounts/services
  - storageAccounts/services/metricDefinitions
  - storageAccounts/storageTaskAssignments
  - storageAccounts/tableServices
  - storageTasks
  - usages

PROPERTIES & ALIAS

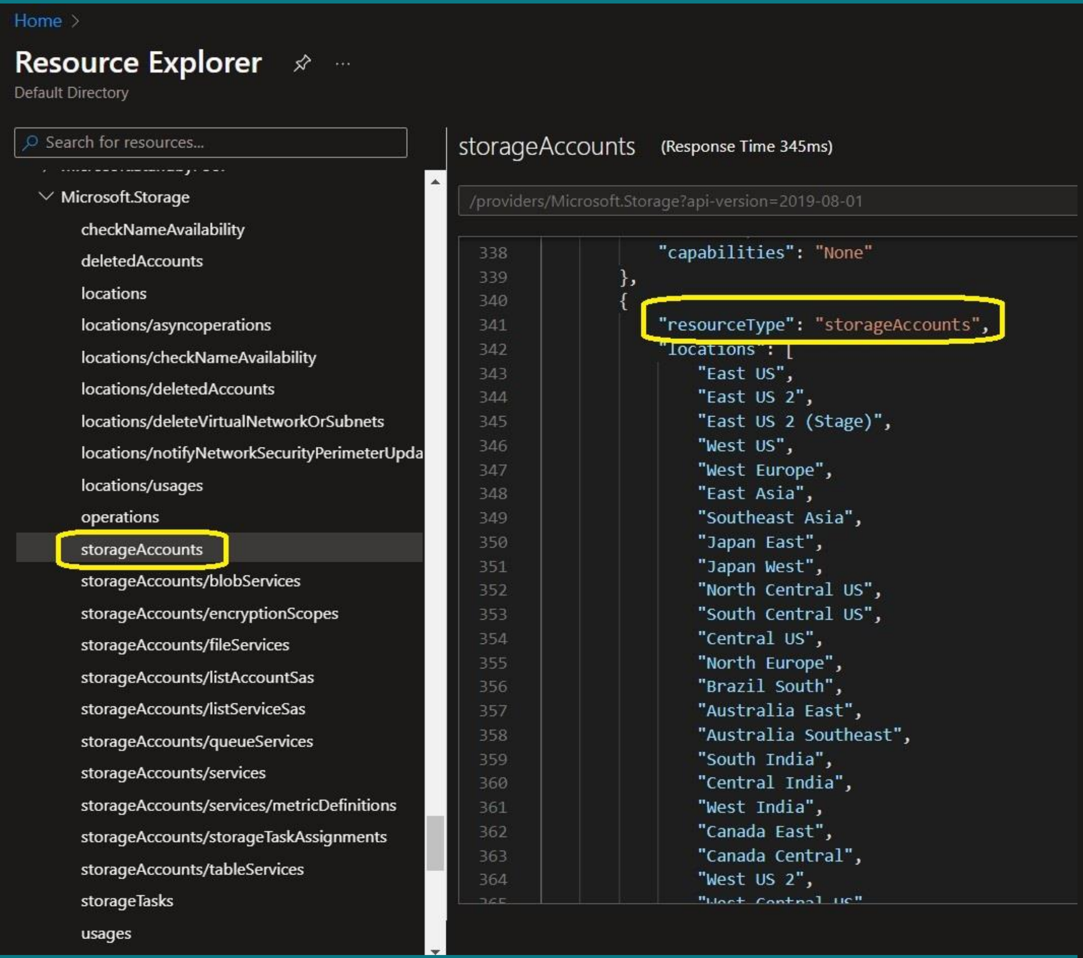


```
PS /home/viktor> az provider show --namespace Microsoft.Storage --expand "resourceTypes/aliases" --query "resourceTypes[].aliases[].name"
[
  "Microsoft.Storage/storageAccounts/storageTaskAssignments/taskId",
  "Microsoft.Storage/storageAccounts/storageTaskAssignments/enabled",
  "Microsoft.Storage/storageAccounts/storageTaskAssignments/description",
  "Microsoft.Storage/storageAccounts/storageTaskAssignments/executionContext",
  "Microsoft.Storage/storageAccounts/storageTaskAssignments/executionContext.target",
  "Microsoft.Storage/storageAccounts/storageTaskAssignments/executionContext.target.prefix",
  "Microsoft.Storage/storageAccounts/storageTaskAssignments/executionContext.target.prefix[*]",
  "Microsoft.Storage/storageAccounts/storageTaskAssignments/executionContext.target.excludePrefix",
```



```
PS /home/viktor> az provider show --namespace Microsoft.Storage --expand "resourceTypes/aliases" --query "resourceTypes[].aliases[].name" | grep supportsHttpsTrafficOnly
  "Microsoft.Storage/storageAccounts/supportsHttpsTrafficOnly",
PS /home/viktor>
```

PROPERTIES & ALIAS



Home >

Resource Explorer

Default Directory

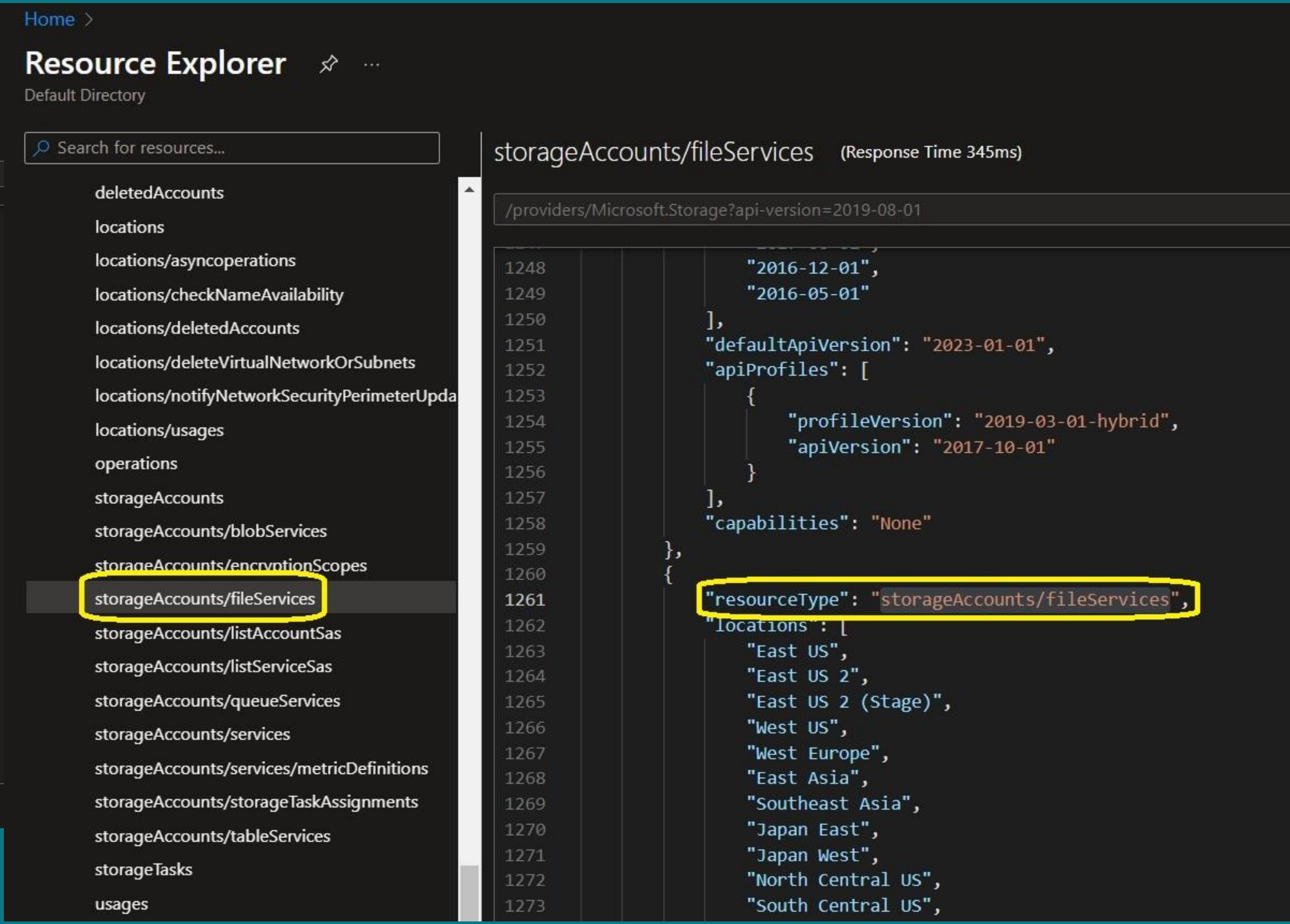
Search for resources...

- Microsoft.Storage
  - checkNameAvailability
  - deletedAccounts
  - locations
  - locations/asyncoperations
  - locations/checkNameAvailability
  - locations/deletedAccounts
  - locations/deleteVirtualNetworkOrSubnets
  - locations/notifyNetworkSecurityPerimeterUpda
  - locations/usages
  - operations
  - storageAccounts
  - storageAccounts/blobServices
  - storageAccounts/encryptionScopes
  - storageAccounts/fileServices
  - storageAccounts/listAccountSas
  - storageAccounts/listServiceSas
  - storageAccounts/queueServices
  - storageAccounts/services
  - storageAccounts/services/metricDefinitions
  - storageAccounts/storageTaskAssignments
  - storageAccounts/tableServices
  - storageTasks
  - usages

storageAccounts (Response Time 345ms)

/providers/Microsoft.Storage?api-version=2019-08-01

```
            "capabilities": "None"
        },
        {
            "resourceType": "storageAccounts",
            "locations": [
                "East US",
                "East US 2",
                "East US 2 (Stage)",
                "West US",
                "West Europe",
                "East Asia",
                "Southeast Asia",
                "Japan East",
                "Japan West",
                "North Central US",
                "South Central US",
                "Central US",
                "North Europe",
                "Brazil South",
                "Australia East",
                "Australia Southeast",
                "South India",
                "Central India",
                "West India",
                "Canada East",
                "Canada Central",
                "West US 2",
```



Home >

Resource Explorer

Default Directory

Search for resources...

- deletedAccounts
- locations
- locations/asyncoperations
- locations/checkNameAvailability
- locations/deletedAccounts
- locations/deleteVirtualNetworkOrSubnets
- locations/notifyNetworkSecurityPerimeterUpda
- locations/usages
- operations
- storageAccounts
- storageAccounts/blobServices
- storageAccounts/encryptionScopes
- storageAccounts/fileServices
- storageAccounts/listAccountSas
- storageAccounts/listServiceSas
- storageAccounts/queueServices
- storageAccounts/services
- storageAccounts/services/metricDefinitions
- storageAccounts/storageTaskAssignments
- storageAccounts/tableServices
- storageTasks
- usages

storageAccounts/fileServices (Response Time 345ms)

/providers/Microsoft.Storage?api-version=2019-08-01

```
                "2016-12-01",
                "2016-05-01"
            ],
            "defaultApiVersion": "2023-01-01",
            "apiProfiles": [
                {
                    "profileVersion": "2019-03-01-hybrid",
                    "apiVersion": "2017-10-01"
                }
            ],
            "capabilities": "None"
        },
        {
            "resourceType": "storageAccounts/fileServices",
            "locations": [
                "East US",
                "East US 2",
                "East US 2 (Stage)",
                "West US",
                "West Europe",
                "East Asia",
                "Southeast Asia",
                "Japan East",
                "Japan West",
                "North Central US",
                "South Central US",
```

PROPERTIES & ALIAS & MODIFABLE

- az provider show --namespace Microsoft.Storage --expand "resourceTypes/aliases" --query "resourceTypes[].aliases[].name”

- (Get-AzPolicyAlias -NamespaceMatch 'Microsoft.Storage').Aliases

PROPERTIES & ALIAS & MODIFABLE

- Get-AzPolicyAlias -NamespaceMatch Microsoft.Sql | where ResourceType -like ‘servers/databases*’ | Select-Object -ExpandProperty ‘Aliases’ | where name -like ‘*backup*’ | Select Name, defaultPath

- Get-AzPolicyAlias -NamespaceMatch Microsoft.Sql | where ResourceType -like ‘*servers/databases*’ | Select-Object -ExpandProperty Aliases | Where-Object { $_.DefaultMetadata.Attributes -eq ‘Modifiable’ } | where name -like ‘*backup*’ |Select Name,DefaultPath

AZURE VS ENTRA ID

MANAGED VS CUSTOM POLICY

PERMISSION

- Microsoft.Authorization
- Microsoft.PolicyInsights

EVALUATION

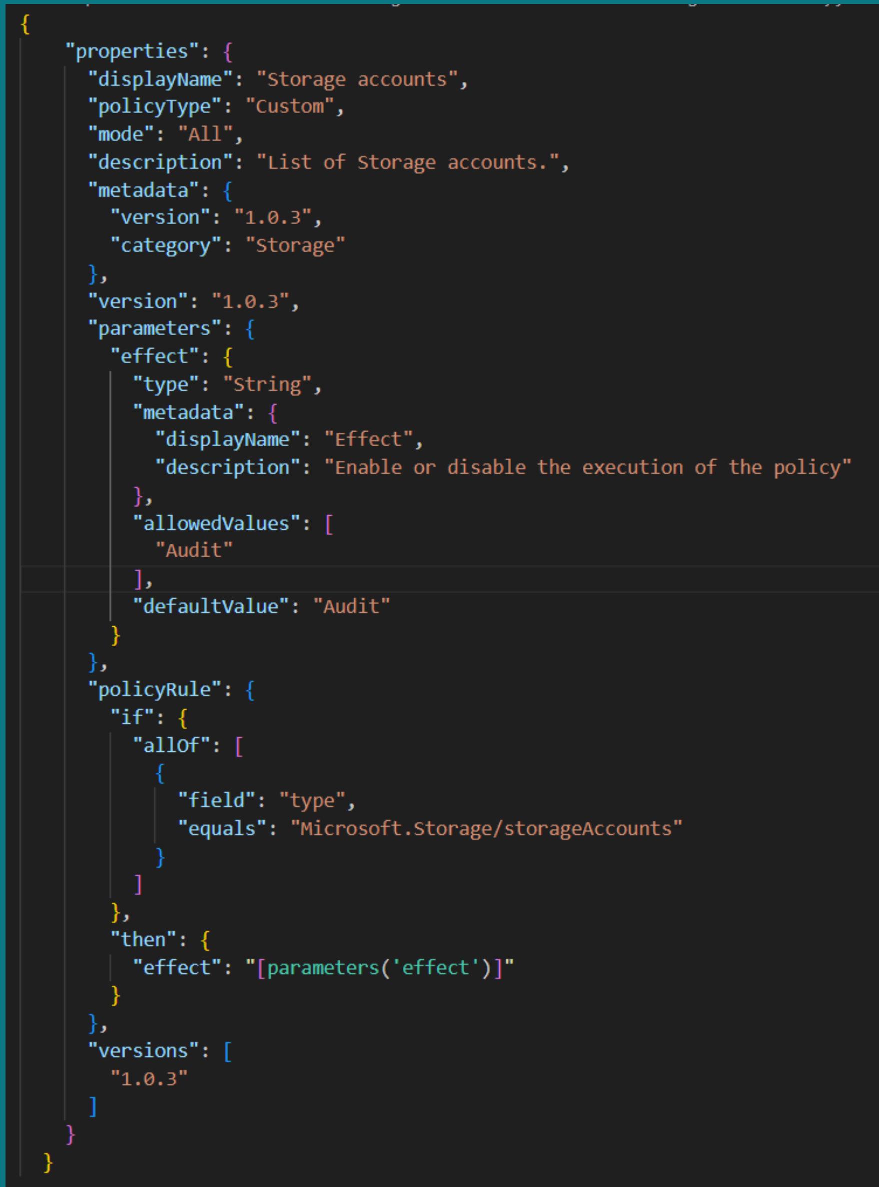
- A resource is deployed to or updated within a scope
- Once every 24 hours, assignments are automatically reevaluated.
- On-Demand Possible (I.e.: Call in Automation Account)

FASTER TRIGGER

- az policy state trigger-scan --resource-group "demo-rg"

ENUMERATION

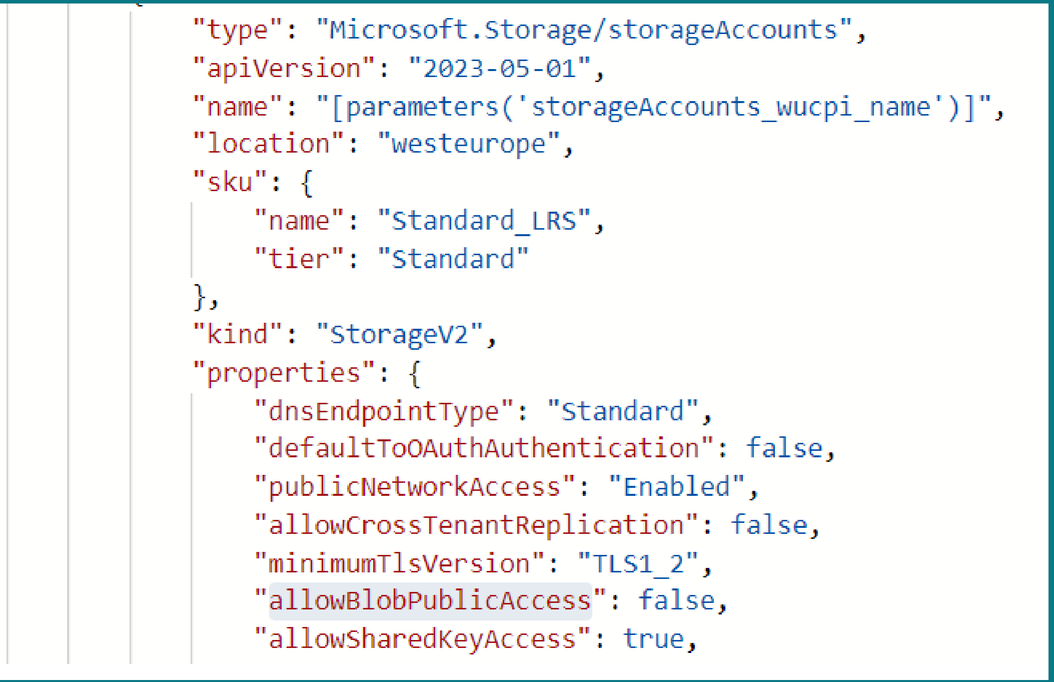
- List of Storage Accounts



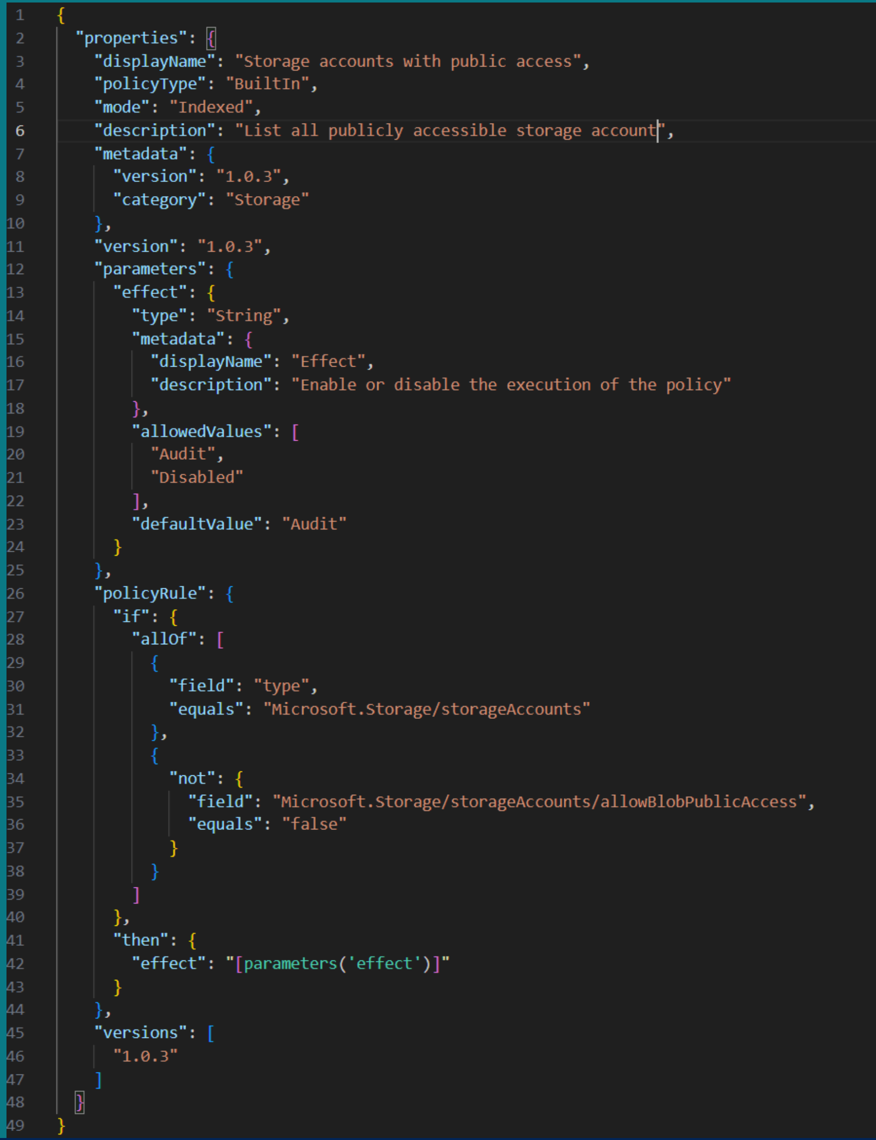
```
{
  "properties": {
   "displayName": "Storage accounts",
   "policyType": "Custom",
   "mode": "All",
   "description": "List of Storage accounts.",
   "metadata": {
    "version": "1.0.3",
    "category": "Storage"
   },
   "version": "1.0.3",
   "parameters": {
    "effect": {
     "type": "String",
     "metadata": {
      "displayName": "Effect",
      "description": "Enable or disable the execution of the policy"
     },
     "allowedValues": [
      "Audit"
     ],
     "defaultValue": "Audit"
    }
   },
   "policyRule": {
    "if": {
     "allOf": [
      {
       "field": "type",
       "equals": "Microsoft.Storage/storageAccounts"
      }
     ]
    },
    "then": {
     "effect": "[parameters('effect')]"
    }
   },
   "versions": [
    "1.0.3"
   ]
  }
 }
```

ENUMERATION

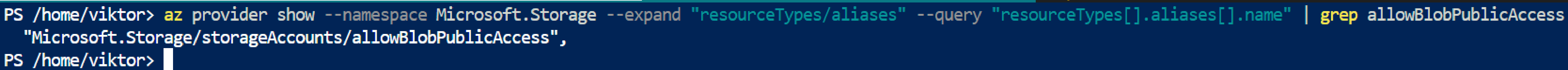
- If resource has anonymous access



```
"type": "Microsoft.Storage/storageAccounts",
"apiVersion": "2023-05-01",
"name": "[parameters('storageAccounts_wucpi_name')]",
"location": "westeurope",
"sku": {
    "name": "Standard_LRS",
    "tier": "Standard"
},
"kind": "StorageV2",
"properties": {
    "dnsEndpointType": "Standard",
    "defaultToOAuthAuthentication": false,
    "publicNetworkAccess": "Enabled",
    "allowCrossTenantReplication": false,
    "minimumTlsVersion": "TLS1_2",
    "allowBlobPublicAccess": false,
    "allowSharedKeyAccess": true,
```



```
{
  "properties": {
    "displayName": "Storage accounts with public access",
    "policyType": "BuiltIn",
    "mode": "Indexed",
    "description": "List all publicly accessible storage account",
    "metadata": {
      "version": "1.0.3",
      "category": "Storage"
    },
    "version": "1.0.3",
    "parameters": {
      "effect": {
        "type": "String",
        "metadata": {
          "displayName": "Effect",
          "description": "Enable or disable the execution of the policy"
        },
        "allowedValues": [
          "Audit",
          "Disabled"
        ],
        "defaultValue": "Audit"
      }
    },
    "policyRule": {
      "if": {
        "allOf": [
          {
            "field": "type",
            "equals": "Microsoft.Storage/storageAccounts"
          },
          {
            "not": {
              "field": "Microsoft.Storage/storageAccounts/allowBlobPublicAccess",
              "equals": "false"
            }
          }
        ]
      },
      "then": {
        "effect": "[parameters('effect')]"
      }
    },
    "versions": [
      "1.0.3"
    ]
  }
}
```

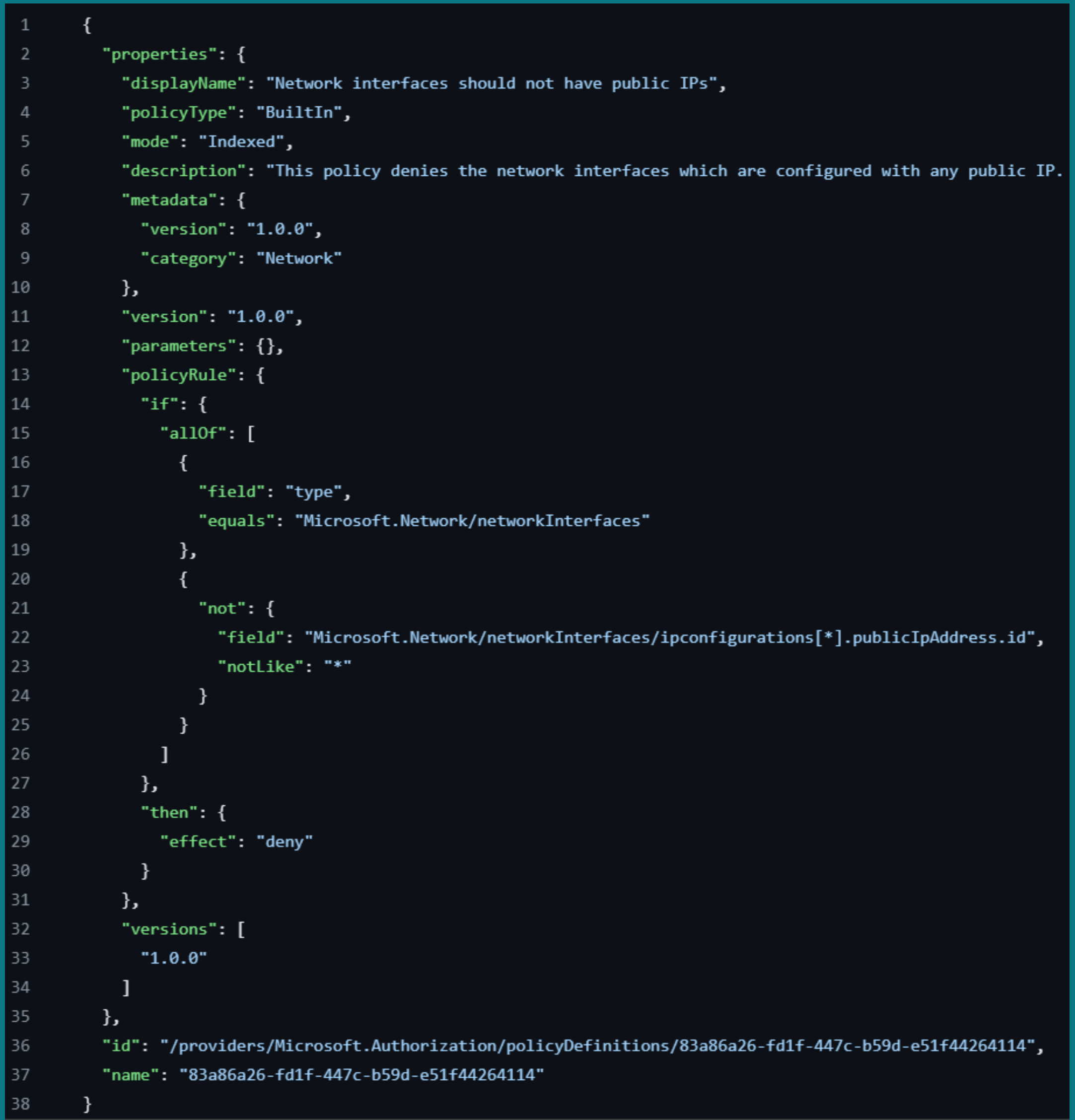


```
PS /home/viktor> az provider show --namespace Microsoft.Storage --expand "resourceTypes/aliases" --query "resourceTypes[].aliases[].name" | grep allowBlobPublicAccess
  "Microsoft.Storage/storageAccounts/allowBlobPublicAccess",
PS /home/viktor> 
```

ENUMERATION

- All The Virtual Machines With External IP – Built-In Policy
- https://github.com/Azure/azure-policy/blob/master/built-in-policies/policyDefinitions/Network/NetworkPublicIPNic_Deny.json

ENUMERATION



```
{
  "properties": {
    "displayName": "Network interfaces should not have public IPs",
    "policyType": "BuiltIn",
    "mode": "Indexed",
    "description": "This policy denies the network interfaces which are configured with any public IP.
    "metadata": {
      "version": "1.0.0",
      "category": "Network"
    },
    "version": "1.0.0",
    "parameters": {},
    "policyRule": {
      "if": {
        "allOf": [
          {
            "field": "type",
            "equals": "Microsoft.Network/networkInterfaces"
          },
          {
            "not": {
              "field": "Microsoft.Network/networkInterfaces/ipconfigurations[*].publicIpAddress.id",
              "notLike": "*"
            }
          }
        ]
      },
      "then": {
        "effect": "deny"
      }
    },
    "versions": [
      "1.0.0"
    ]
  },
  "id": "/providers/Microsoft.Authorization/policyDefinitions/83a86a26-fd1f-447c-b59d-e51f44264114",
  "name": "83a86a26-fd1f-447c-b59d-e51f44264114"
}
```

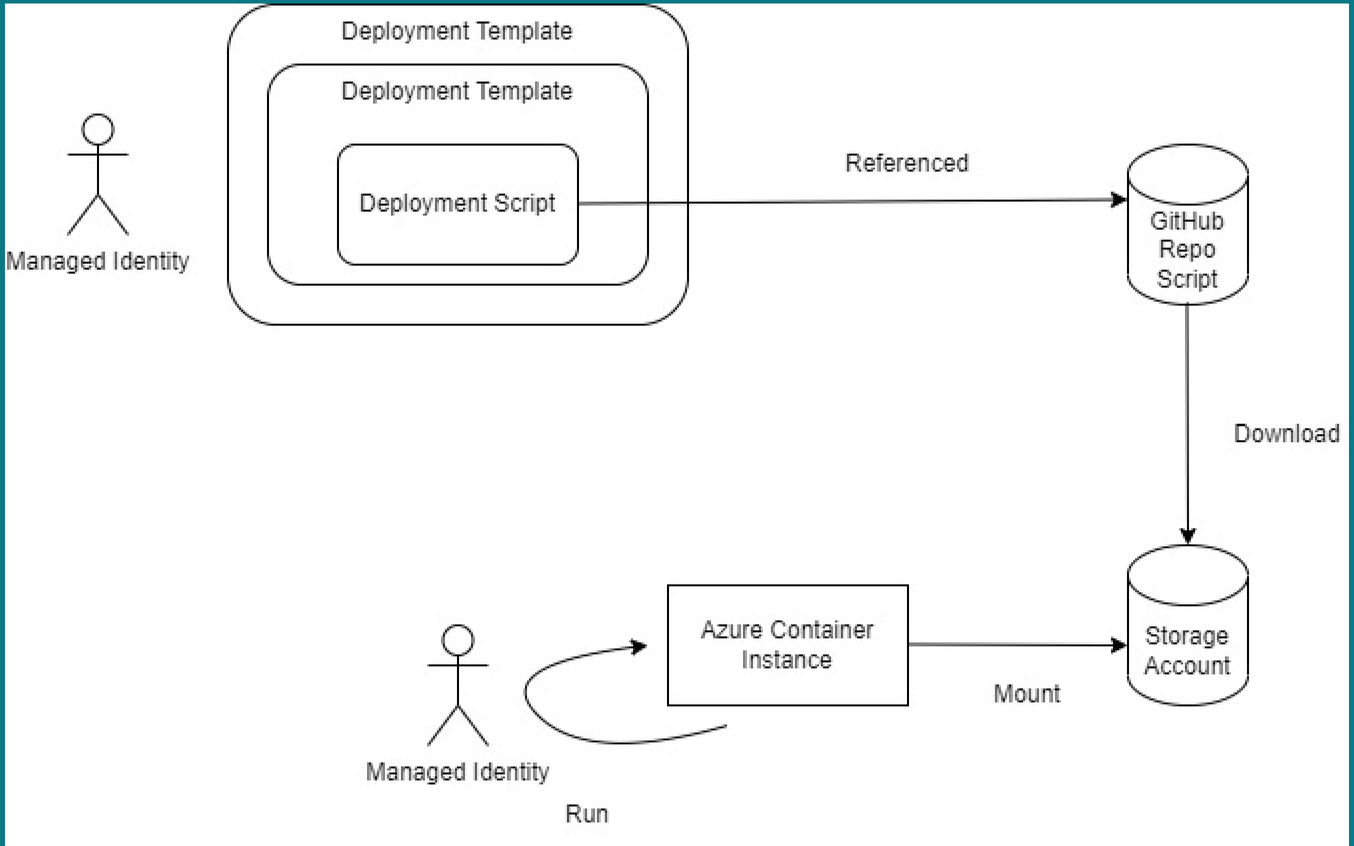
PROOF OF CONCEPT

- Check If Storage Account Has HTTPS (supportsHttpsTrafficOnly)
- Remediation Task – DeployIfNotExists Effect
- Role Assignment To The Managed Identity
- Download Deployment Script (PowerShell) From External URL

PROOF OF CONCEPT

- Create A Service Principal
- Call Controlled Website With Secret
- Triggered By Default Every 24 Hours, But Not The Remediation
- The Remediation Task Will Do Other Things As I Modified It

PROOF OF CONCEPT



PROOF OF CONCEPT ELEMENTS

- Storage Account
- Managed Identity (For SP Creation – App Admin, User Admin, Owner and Not Contributor)
- Azure Container Instance
- Deployment
- Deployment Script
- Website / GitHub

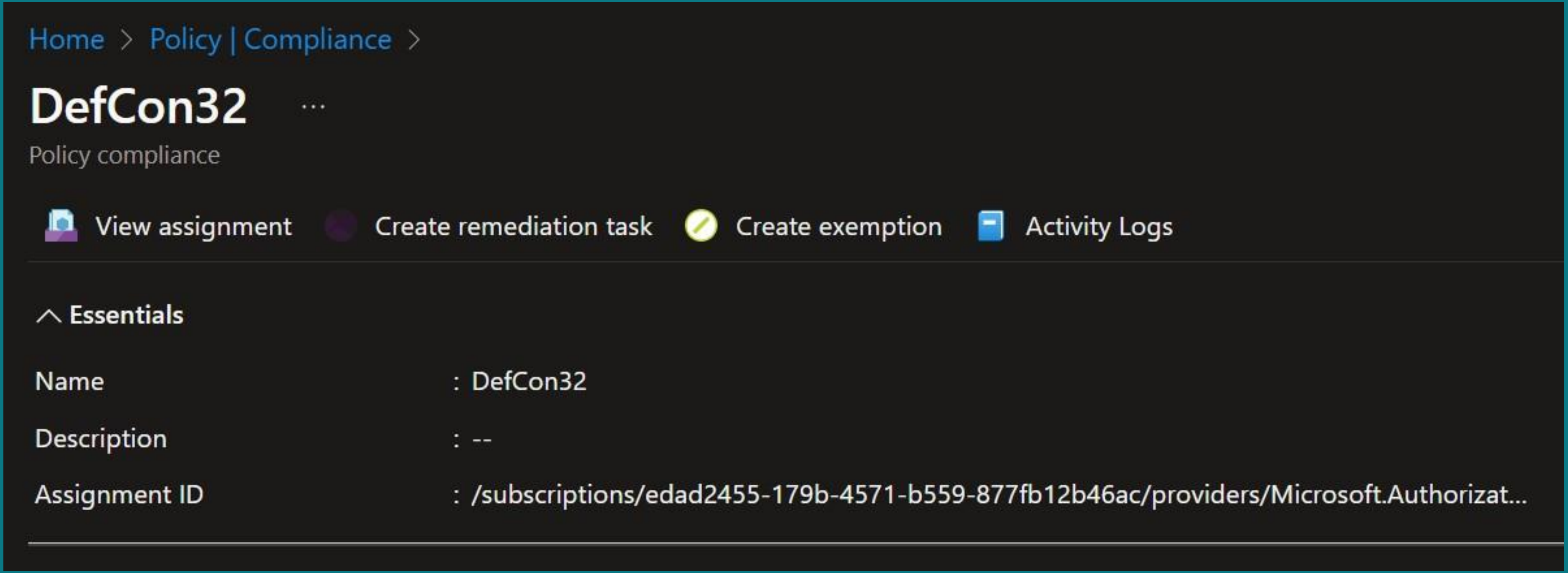
DEMO

- Video - 8:30 mins

PROOF OF CONCEPT - PROBLEMS

- Requires A "LOT"
- Basically Everything Is In Audit Log
- Deployment And Script Stored In The Storage Account
- Remediation Task Run Only Once

AUDIT LOG



DefCon32 ...

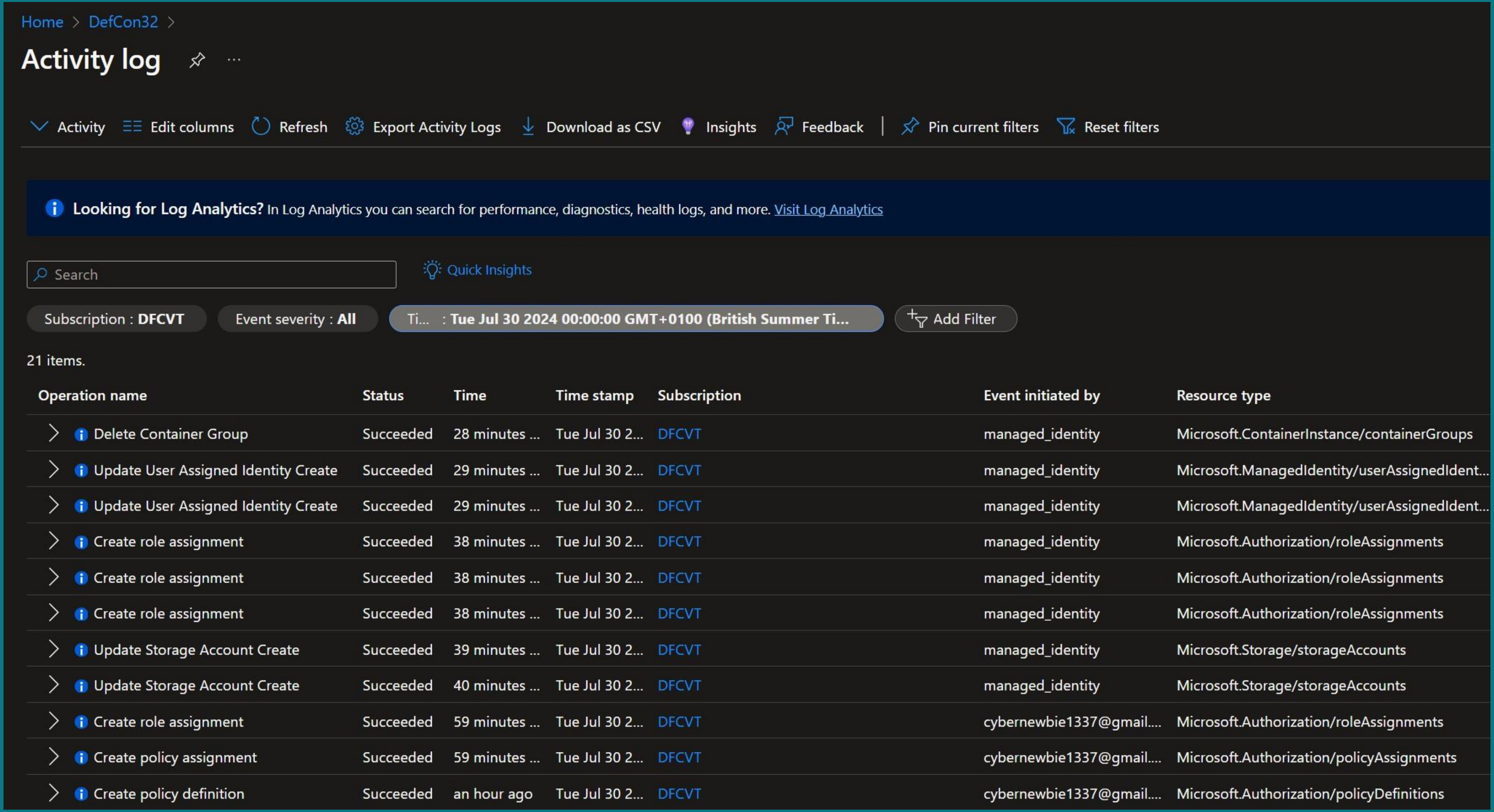
Policy compliance

Name : DefCon32

Description : --

Assignment ID : /subscriptions/edad2455-179b-4571-b559-877fb12b46ac/providers/Microsoft.Authorizat...

AZURE AUDIT LOG



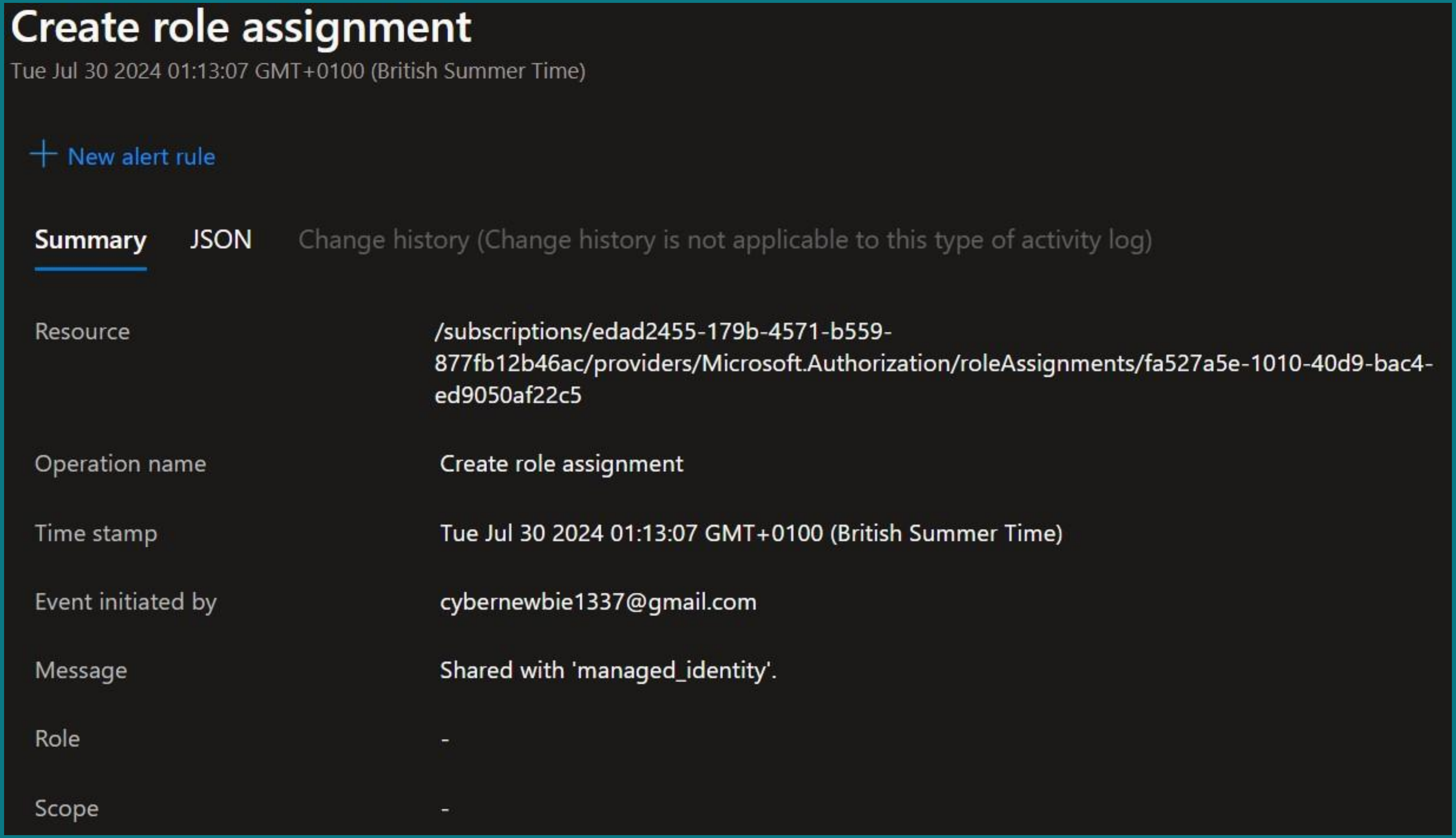
Looking for Log Analytics? In Log Analytics you can search for performance, diagnostics, health logs, and more. Visit Log Analytics

Subscription : **DFCVT**   Event severity : **All**   Ti... : **Tue Jul 30 2024 00:00:00 GMT+0100 (British Summer Ti...**

21 items.

Operation name	Status	Time	Time stamp	Subscription	Event initiated by	Resource type
Delete Container Group	Succeeded	28 minutes ...	Tue Jul 30 2...	DFCVT	managed_identity	Microsoft.ContainerInstance/containerGroups
Update User Assigned Identity Create	Succeeded	29 minutes ...	Tue Jul 30 2...	DFCVT	managed_identity	Microsoft.ManagedIdentity/userAssignedIdent...
Update User Assigned Identity Create	Succeeded	29 minutes ...	Tue Jul 30 2...	DFCVT	managed_identity	Microsoft.ManagedIdentity/userAssignedIdent...
Create role assignment	Succeeded	38 minutes ...	Tue Jul 30 2...	DFCVT	managed_identity	Microsoft.Authorization/roleAssignments
Create role assignment	Succeeded	38 minutes ...	Tue Jul 30 2...	DFCVT	managed_identity	Microsoft.Authorization/roleAssignments
Create role assignment	Succeeded	38 minutes ...	Tue Jul 30 2...	DFCVT	managed_identity	Microsoft.Authorization/roleAssignments
Update Storage Account Create	Succeeded	39 minutes ...	Tue Jul 30 2...	DFCVT	managed_identity	Microsoft.Storage/storageAccounts
Update Storage Account Create	Succeeded	40 minutes ...	Tue Jul 30 2...	DFCVT	managed_identity	Microsoft.Storage/storageAccounts
Create role assignment	Succeeded	59 minutes ...	Tue Jul 30 2...	DFCVT	cybernewbie1337@gmail....	Microsoft.Authorization/roleAssignments
Create policy assignment	Succeeded	59 minutes ...	Tue Jul 30 2...	DFCVT	cybernewbie1337@gmail....	Microsoft.Authorization/policyAssignments
Create policy definition	Succeeded	an hour ago	Tue Jul 30 2...	DFCVT	cybernewbie1337@gmail....	Microsoft.Authorization/policyDefinitions

AZURE AUDIT LOG



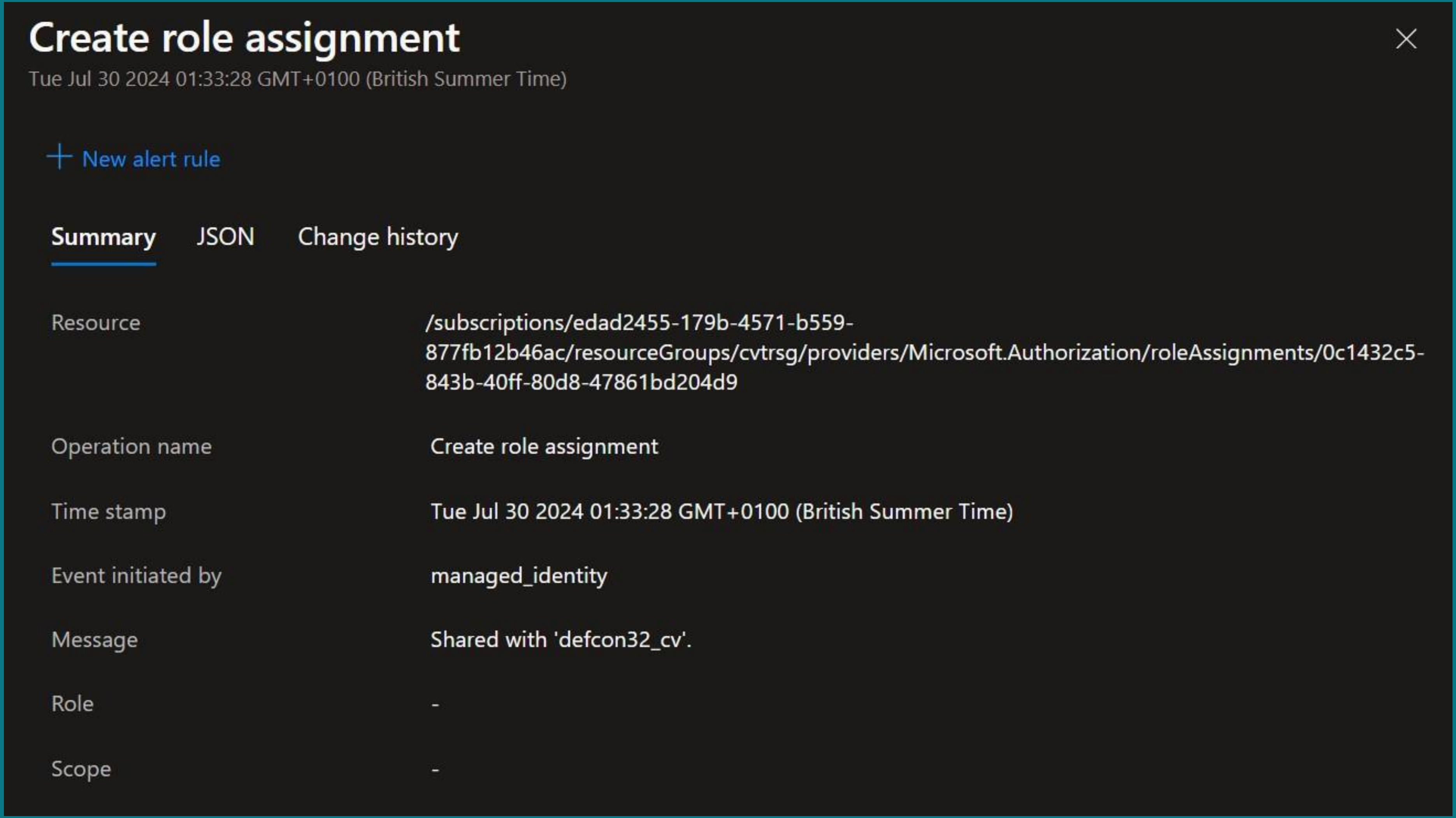
Create role assignment

Tue Jul 30 2024 01:13:07 GMT+0100 (British Summer Time)

+ New alert rule
Summary
JSON
Change history (Change history is not applicable to this type of activity log)

Resource	/subscriptions/edad2455-179b-4571-b559-877fb12b46ac/providers/Microsoft.Authorization/roleAssignments/fa527a5e-1010-40d9-bac4-ed9050af22c5
Operation name	Create role assignment
Time stamp	Tue Jul 30 2024 01:13:07 GMT+0100 (British Summer Time)
Event initiated by	cybernewbie1337@gmail.com
Message	Shared with 'managed_identity'.
Role	-
Scope	-

AZURE AUDIT LOG

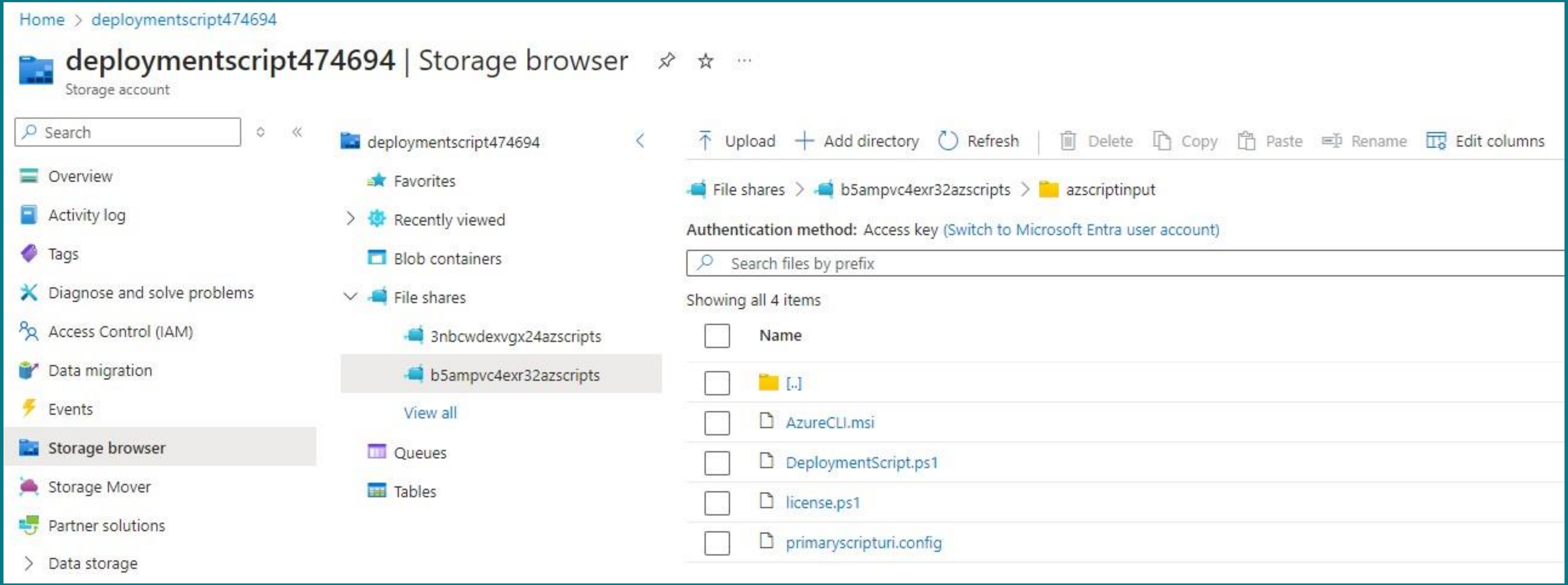


Create role assignment

Tue Jul 30 2024 01:33:28 GMT+0100 (British Summer Time)

Resource	/subscriptions/edad2455-179b-4571-b559-877fb12b46ac/resourceGroups/cvtrsg/providers/Microsoft.Authorization/roleAssignments/0c1432c5-843b-40ff-80d8-47861bd204d9
Operation name	Create role assignment
Time stamp	Tue Jul 30 2024 01:33:28 GMT+0100 (British Summer Time)
Event initiated by	managed_identity
Message	Shared with 'defcon32_cv'.
Role	-
Scope	-

AZURE AUDIT LOG



Home > deploymentscript474694

deploymentscript474694 | Storage browser

Storage account

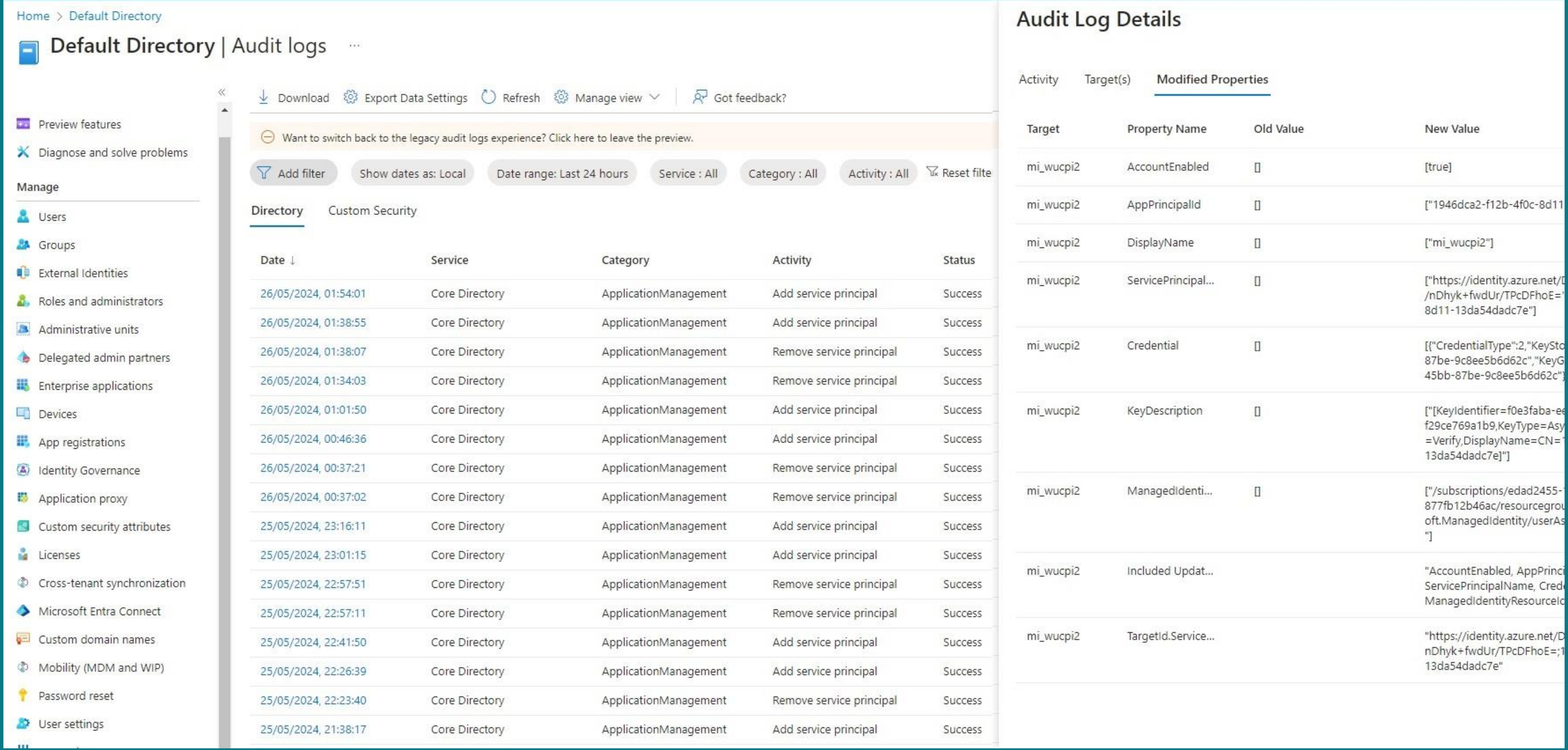
File shares > b5ampvc4exr32azscripts > azscriptinput

Authentication method: Access key (Switch to Microsoft Entra user account)

Showing all 4 items

☐	Name
☐	[..]
☐	AzureCLI.msi
☐	DeploymentScript.ps1
☐	license.ps1
☐	primaryscripturi.config

ENTRA ID AUDIT LOG



Want to switch back to the legacy audit logs experience? Click here to leave the preview.

Show dates as: Local · Date range: Last 24 hours · Service : All · Category : All · Activity : All

Date ↓	Service	Category	Activity	Status
26/05/2024, 01:54:01	Core Directory	ApplicationManagement	Add service principal	Success
26/05/2024, 01:38:55	Core Directory	ApplicationManagement	Add service principal	Success
26/05/2024, 01:38:07	Core Directory	ApplicationManagement	Remove service principal	Success
26/05/2024, 01:34:03	Core Directory	ApplicationManagement	Remove service principal	Success
26/05/2024, 01:01:50	Core Directory	ApplicationManagement	Add service principal	Success
26/05/2024, 00:46:36	Core Directory	ApplicationManagement	Add service principal	Success
26/05/2024, 00:37:21	Core Directory	ApplicationManagement	Remove service principal	Success
26/05/2024, 00:37:02	Core Directory	ApplicationManagement	Remove service principal	Success
25/05/2024, 23:16:11	Core Directory	ApplicationManagement	Add service principal	Success
25/05/2024, 23:01:15	Core Directory	ApplicationManagement	Add service principal	Success
25/05/2024, 22:57:51	Core Directory	ApplicationManagement	Remove service principal	Success
25/05/2024, 22:57:11	Core Directory	ApplicationManagement	Remove service principal	Success
25/05/2024, 22:41:50	Core Directory	ApplicationManagement	Add service principal	Success
25/05/2024, 22:26:39	Core Directory	ApplicationManagement	Add service principal	Success
25/05/2024, 22:23:40	Core Directory	ApplicationManagement	Remove service principal	Success
25/05/2024, 21:38:17	Core Directory	ApplicationManagement	Add service principal	Success

Audit Log Details

Target	Property Name	Old Value	New Value
mi_wucpi2	AccountEnabled	[]	[true]
mi_wucpi2	AppPrincipalId	[]	["1946dca2-f12b-4f0c-8d11
mi_wucpi2	DisplayName	[]	["mi_wucpi2"]
mi_wucpi2	ServicePrincipal...	[]	["https://identity.azure.net/D /nDhyk+fwdUr/TPcDFhoE=" 8d11-13da54dadc7e"]
mi_wucpi2	Credential	[]	[{"CredentialType":2,"KeySto 87be-9c8ee5b6d62c","KeyG 45bb-87be-9c8ee5b6d62c"}
mi_wucpi2	KeyDescription	[]	["[KeyIdentifier=f0e3faba-ee f29ce769a1b9,KeyType=Asy =Verify,DisplayName=CN=1 13da54dadc7e]"]
mi_wucpi2	ManagedIdenti...	[]	["/subscriptions/edad2455-1 877fb12b46ac/resourcegrou oft.ManagedIdentity/userAs "]
mi_wucpi2	Included Updat...		"AccountEnabled, AppPrinci ServicePrincipalName, Crede ManagedIdentityResourceIc
mi_wucpi2	TargetId.Service...		"https://identity.azure.net/D nDhyk+fwdUr/TPcDFhoE=;1 13da54dadc7e"

LINKS 1

- Original Policy - https://techcommunity.microsoft.com/t5/core-infrastructure-and-security/azure-policy-remediation-with-deployment-scripts/ba-p/1594674
- Azure Policy Examples - https://github.com/Azure/azure-policy
- Azure Community Policy Examples - https://github.com/Azure/Community-Policy
- Azure Policy Examples - https://www.azadvertizer.net/azpolicyadvertizer_all.html

LINKS 2

- https://codingfor.beer/creating-an-azure-policy-remediation-task-for-an-azure-policy-initiative-e75214373c99
- https://marcogerber.ch/the-power-of-azure-policy-deployifnotexists/
- https://itnext.io/implementing-and-troubleshooting-a-custom-azure-policy-definition-1ab8d0634bb2

LINKS 3

- https://stefanstranger.github.io/2021/01/22/LessonsLearnedDevelopingACustomPolicy/
- https://journeyofthegeek.com/2021/03/07/writing-a-custom-azure-policy/
- https://msftplayground.com/2022/03/building-your-azure-policies-part-1/

LINKS 4

- https://securitylabs.datadoghq.com/articles/azure-policy-privilege-escalation/
- https://medium.com/cyesec/azure-ad-privilege-escalation-through-auto-assignment-policies-2d0422be4f25
- https://medium.com/@vladimir.tul/elevating-privileges-through-azure-policy-872298cf673f

QUESTIONS

- @wucpi
- viktor.gazdag@nccgroup.com