**Testing Instructions: Cisco Umbrella Data Connector with Private Storage Account**

**Overview**

This document provides step-by-step instructions for testing the Cisco Umbrella data connector deployment with private storage account configuration.

**Prerequisites**

Before beginning the deployment, ensure you have:

* An Azure subscription with permissions to create resources
* A Cisco Umbrella account with permissions to access the S3 bucket
* **Network Contributor** permissions on the Virtual Network and subnets
* An existing Virtual Network (VNet) with **two dedicated subnets**:
  + **Function App subnet**: Must be delegated to Microsoft.Web/serverFarms for VNet integration
  + **Private Endpoints subnet**: Must NOT be delegated (for storage account private endpoints)
* Required information:
  + Cisco Umbrella S3 bucket name
  + AWS Access Key ID with permissions to read from the S3 bucket
  + AWS Secret Access Key
  + Log Analytics Workspace ID and Key
  + Existing VNet name and resource group
  + Function App subnet name (delegated to Microsoft.Web/serverFarms)
  + Private endpoints subnet name (no delegation)

**Deployment Steps**

**1. Repository Setup**

git clone https://github.com/Azure/Azure-Sentinel.git

cd Azure-Sentinel

git checkout cisco\_umbrella\_private\_storageaccount

**2. Network Setup (Critical Step)**

Before deployment, you must prepare the network infrastructure:

a. **Create or identify an existing VNet** with sufficient address space b. **Create two dedicated subnets**:

**Function App Subnet (with delegation):**

# Create subnet for Function App with delegation

az network vnet subnet create \

--resource-group <your-resource-group> \

--vnet-name <your-vnet-name> \

--name function-subnet \

--address-prefixes 10.0.1.0/26 \

--delegations Microsoft.Web/serverFarms

**Private Endpoints Subnet (no delegation):**

# Create subnet for Private Endpoints without delegation

az network vnet subnet create \

--resource-group <your-resource-group> \

--vnet-name <your-vnet-name> \

--name private-endpoints-subnet \

--address-prefixes 10.0.2.0/26

**Important Notes:**

* Private endpoints **cannot** be created in a delegated subnet
* Function Apps **require** subnet delegation for VNet integration
* Use different address ranges that don't overlap with existing subnets

**3. Deployment Process**

a. Navigate to Azure Portal (https://portal.azure.com) b. Open Azure Cloud Shell or use Azure CLI locally c. Navigate to the connector directory:

cd Solutions/CiscoUmbrella/Data\ Connectors/

d. Execute the deployment command with **all required network parameters**:

az deployment group create --resource-group <your-resource-group> \

--template-file azuredeploy\_CiscoUmbrella\_API\_FunctionApp\_elasticpremium.json \

--parameters FunctionName=<unique-function-name> \

WorkspaceID=<your-workspace-id> \

WorkspaceKey=<your-workspace-key> \

S3Bucket=<your-s3-bucket-name> \

AWSAccessKeyId=<your-aws-access-key-id> \

AWSSecretAccessKey=<your-aws-secret-key> \

AppInsightsWorkspaceResourceID=<your-app-insights-workspace-resource-id> \

existingVnetName=<your-vnet-name> \

existingVnetResourceGroupName=<vnet-resource-group> \

existingSubnetName=<function-app-subnet-name> \

existingPrivateEndpointSubnetName=<private-endpoints-subnet-name>

**Example with actual values:**

az deployment group create --resource-group test\_hfh\_rg \

--template-file azuredeploy\_CiscoUmbrella\_API\_FunctionApp\_elasticpremium.json \

--parameters FunctionName=UmbrellaV2 \

WorkspaceID=bb557eee-7f69-47bd-ab82-21e28aabba6d \

WorkspaceKey="your-workspace-key" \

S3Bucket=ciscoumbrella-test1 \

AWSAccessKeyId=AKIARB2PKUDQTR6ZXSFX \

AWSSecretAccessKey="your-aws-secret" \

AppInsightsWorkspaceResourceID="/subscriptions/082909d8-d042-4ef9-99f9-dd4a4ed283de/resourceGroups/DefaultResourceGroup-CUS/providers/Microsoft.OperationalInsights/workspaces/DefaultWorkspace-082909d8-d042-4ef9-99f9-dd4a4ed283de-CUS" \

existingVnetName=testhfh1-vnet \

existingVnetResourceGroupName=test\_hfh\_rg \

existingSubnetName=function-subnet \

existingPrivateEndpointSubnetName=private-endpoints-subnet

**Verification Steps**

**1. Resource Verification**

Navigate to the Azure Portal and verify the creation of:

* **App Service Plan** (ElasticPremium EP1)
* **Azure Function App** (with VNet integration enabled)
* **Storage Account** (with private endpoint and public access disabled)
* **Four Private Endpoints** for storage services:
  + Blob private endpoint
  + File private endpoint
  + Queue private endpoint
  + Table private endpoint
* **Four Private DNS Zones** (automatically created):
  + privatelink.blob.core.windows.net
  + privatelink.file.core.windows.net
  + privatelink.queue.core.windows.net
  + privatelink.table.core.windows.net
* **VNet Links** for each private DNS zone
* **DNS Zone Groups** connecting private endpoints to DNS zones
* **Application Insights** for monitoring
* **Storage containers**: azure-webjobs-hosts, azure-webjobs-secrets
* **File share** for Function App storage

**2. Network Verification**

Verify the network configuration:

* Function App is connected to the **delegated subnet** (function-subnet)
* Private endpoints are deployed in the **non-delegated subnet** (private-endpoints-subnet)
* Storage account has **"Public network access: Disabled"**
* Private endpoint connections show as **"Approved"** status
* **Private DNS zones** are created and linked to the VNet
* **DNS zone groups** connect private endpoints to DNS zones for proper resolution

**3. Function App Monitoring**

a. Access the Function App in Azure Portal b. Go to "Functions" > "CiscoUmbrella" > "Monitor" c. Check logs for any errors d. First invocation should occur within 5-10 minutes e. **Verify VNet Integration**: Check "Networking" > "VNet Integration" shows connected status

**4. Data Ingestion Verification**

a. Open your Log Analytics workspace b. Navigate to "Logs" c. Run the following KQL query:

Cisco\_Umbrella

| take 10

d. Expect data to appear within 10-15 minutes of deployment

**Security Verification**

**Storage Account Security**

Verify the following settings in the Storage Account:

* **Public network access**: Disabled
* **Private endpoints**: 4 endpoints connected (blob, file, queue, table)
* **Network access**: Only via private endpoint
* **No public IP access** is allowed
* **Default action**: Deny (under Networking > Firewalls and virtual networks)

**Function App Security**

Confirm the following in the Function App:

* **VNet integration**: Active and connected to delegated subnet
* **System-assigned managed identity**: Enabled
* **Private endpoints**: Can access storage via private network
* **Outbound IP addresses**: Routed through VNet (not public IPs when accessing storage)

**Network Security**

Verify network configuration:

* **Subnet delegation**: Function App subnet delegated to Microsoft.Web/serverFarms
* **Private endpoint subnet**: No delegation, contains 4 private endpoints
* **DNS resolution**: Private endpoints resolve to private IP addresses (10.0.2.x range)
* **NSG rules**: Allow traffic between Function App and private endpoint subnets

**Troubleshooting Guide**

**Common Deployment Errors**

**Error: "Encountered an error (InternalServerError) from host runtime"**

* This is a common issue with Function Apps using VNet integration and private endpoints
* **Root causes and solutions:**
  1. **Function App cannot access storage account via private endpoints**:
     + Check if Function App subnet can reach private endpoint subnet
     + Verify NSG rules allow traffic between subnets
     + Test DNS resolution: private endpoints should resolve to 10.0.2.x IPs
  2. **DNS resolution issues**:
     + **This is now automatically handled by the ARM template**
     + The template creates private DNS zones for all storage services
     + DNS zones are automatically linked to your VNet
     + Private endpoints are configured with DNS zone groups for proper resolution
  3. **Storage account connection string issues**:
     + Function App may be trying to access storage via public endpoint
     + Verify AzureWebJobsStorage setting uses private endpoint
  4. **App Service Plan insufficient resources**:
     + ElasticPremium EP1 may need scaling up
     + Check App Service Plan metrics for resource constraints

**Troubleshooting steps for InternalServerError:**

# Check Function App logs

az functionapp log tail --name <function-app-name> --resource-group <rg>

# Test DNS resolution from Function App (if SSH/Console access available)

nslookup <storage-account-name>.blob.core.windows.net

# Check storage account network settings

az storage account show --name <storage-account-name> --resource-group <rg> --query "networkRuleSet"

# Verify private endpoint connections

az network private-endpoint list --resource-group <rg> --query "[].{Name:name,State:privateLinkServiceConnections[0].privateLinkServiceConnectionState.status}"

# Verify private DNS zones were created automatically

az network private-dns zone list --resource-group <rg> --query "[].{Name:name,LinkedVnets:numberOfVirtualNetworkLinks}"

**Error: "The content for this response was already consumed"**

* This typically masks ARM template validation issues
* Run template validation first: az deployment group validate
* Check for empty strings in dependsOn arrays
* Clear Azure CLI cache: az cache purge

**Error: "Subnet delegation cannot be changed"**

* Subnet is already in use by other resources
* Create a new dedicated subnet for Function App
* Ensure the subnet is delegated to Microsoft.Web/serverFarms

**Error: "Private endpoint creation not allowed as subnet is delegated"**

* Private endpoints cannot be created in delegated subnets
* Use separate subnets: one delegated (Function App), one non-delegated (Private Endpoints)

**Error: "App Service Plan is required to use Regional VNET Integration"**

* ARM template must include App Service Plan resource
* Use ElasticPremium SKU (EP1) for VNet integration support

**If No Data Appears or Function App Shows InternalServerError**

Check the following in order:

1. **Function App detailed logs** for specific error messages:
2. az functionapp log tail --name <function-app-name> --resource-group <rg>
3. **Storage account connectivity test**:
   * Go to Function App → Development Tools → Console
   * Run: nslookup <storage-account-name>.blob.core.windows.net
   * Should resolve to private IP (10.0.2.x), not public IP
4. **Private DNS zones** (if DNS resolution fails):
   * **The ARM template automatically creates these resources:**
     + privatelink.blob.core.windows.net
     + privatelink.file.core.windows.net
     + privatelink.queue.core.windows.net
     + privatelink.table.core.windows.net
   * **DNS zones are automatically linked to your VNet**
   * **Private endpoints include DNS zone group configurations**
   * You can **verify creation** with: az network private-dns zone list --resource-group <rg>
5. **AWS credentials** and S3 bucket permissions
6. **VNet integration status** (Networking > VNet Integration)
7. **Private endpoint connectivity** to storage account
8. **S3 bucket** contains recent Cisco Umbrella logs
9. **NSG rules** between Function App and private endpoint subnets

**Network Connectivity Issues**

1. **DNS resolution**: Verify private endpoints resolve to private IPs (10.0.2.x range)
2. # From Function App console or SSH
3. nslookup <storage-account-name>.blob.core.windows.net
4. # Should return private IP, not public
5. **NSG rules**: Ensure traffic allowed between Function App and private endpoint subnets
6. **Route tables**: Check custom routes don't block private endpoint traffic
7. **Function App outbound connectivity**: Should use VNet, not public IPs
8. **Private DNS zones missing**: **Automatically created by ARM template**
   * The template now creates all required private DNS zones
   * DNS zones are linked to your VNet automatically
   * Private endpoints include DNS zone group configurations
   * Verify with: az network private-dns zone list --resource-group <rg>
9. **Storage account firewall**: Verify it's set to "Deny" public access
10. **VNet integration**: Check Function App → Networking → VNet Integration status

**Additional Verification Points**

* **ARM template validation**: Always validate before deployment
* **Resource deployment status**: Check all resources deployed successfully
* **Subnet address spaces**: Ensure no overlap between subnets
* **Private endpoint status**: All 4 endpoints should show "Approved"
* **Storage account metrics**: Monitor access patterns via Azure Monitor

**Reporting Results**

Please provide feedback on:

1. **Deployment status** (success/failure with specific error messages)
2. **Network setup** (subnet creation and delegation process)
3. **Template validation** results before deployment
4. **Resource creation** status (all expected resources deployed)
5. **Data flow status** in Log Analytics (time to first data ingestion)
6. **Private endpoint connectivity** (storage access via private network)
7. **VNet integration** functionality (Function App network isolation)
8. **Any networking, permission, or configuration issues encountered**

**Support and Additional Resources**

For any issues during testing, please check:

* **Function App logs** (Monitor > Logs)
* **ARM template validation** output
* **Network configuration** (VNet, subnets, private endpoints)
* **Azure Portal deployment status** (Resource Group > Deployments)
* **AWS credentials and S3 permissions**
* **Private endpoint DNS resolution** (should resolve to private IPs)

**Useful CLI commands for troubleshooting:**

# Check deployment status

az deployment group list --resource-group <rg> --query "[].{Name:name,State:properties.provisioningState}"

# Validate template before deployment

az deployment group validate --resource-group <rg> --template-file azuredeploy\_CiscoUmbrella\_API\_FunctionApp\_elasticpremium.json --parameters @azuredeploy.parameters.all.json

# Check subnet delegation

az network vnet subnet show --resource-group <rg> --vnet-name <vnet> --name <subnet> --query "delegations"

# List private endpoints and their status

az network private-endpoint list --resource-group <rg> --query "[].{Name:name,Subnet:subnet.id,Status:privateLinkServiceConnections[0].privateLinkServiceConnectionState.status}"

# Get Function App logs (for InternalServerError troubleshooting)

az functionapp log tail --name <function-app-name> --resource-group <rg>

# Check VNet integration status

az functionapp vnet-integration list --name <function-app-name> --resource-group <rg>

# Test storage account network access

az storage account show --name <storage-account-name> --resource-group <rg> --query "{PublicAccess:publicNetworkAccess,DefaultAction:networkRuleSet.defaultAction}"

# Check private DNS zone configuration (automatically created by ARM template)

az network private-dns zone list --resource-group <rg> --query "[].{Name:name,LinkedVnets:numberOfVirtualNetworkLinks}"