

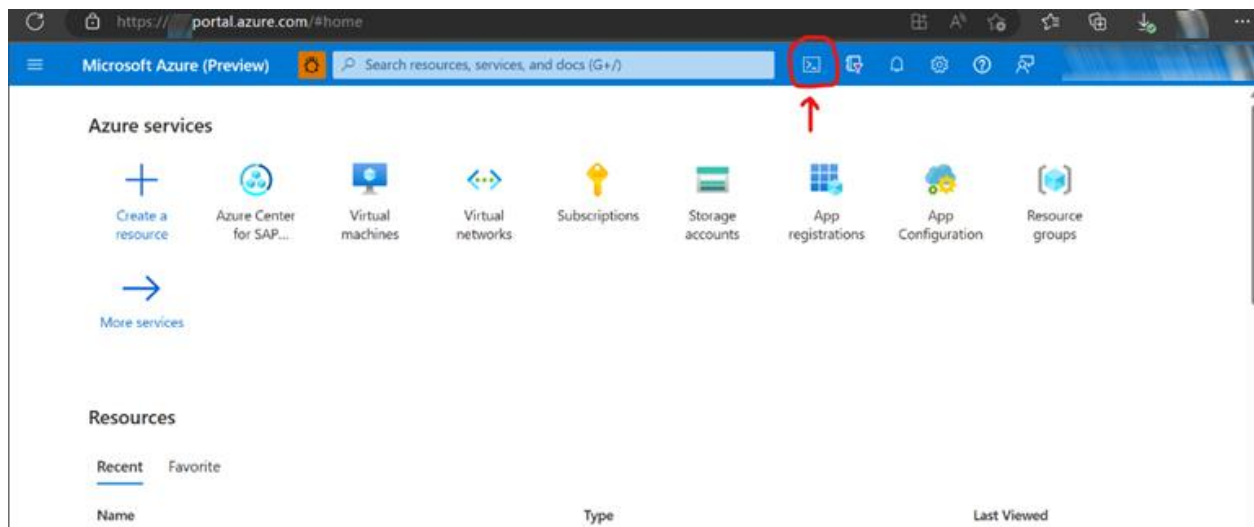
Monitoring Extension Windows manual upgrade

Please follow these steps to manually upgrade the Windows Monitoring Extension on all VMs running the extension, in a particular subscription,

Pre-requisite: To be able to run the necessary commands, you will require **Contributor** access or higher on the subscription.

1. Open Cloud shell

- Sign into the Azure Portal (<https://portal.azure.com>) using your Azure account credentials.
- Once you are signed in, click on the Cloud Shell icon located in the top right corner of the Azure Portal, next to your username.



- If you are using Cloud Shell for the first time, you will be prompted to choose a storage account for your Cloud Shell environment. Select "Create storage" to create a new storage account or select an existing one.
- After you have selected a storage account, the Cloud Shell environment will start initializing. This may take a few seconds.

2. Execute below steps in CloudShell,
 - a. Replace YOUR-SUBSCRIPTION-ID with the actual subscriptionId
 - b. Copy paste below content into CloudShell and hit Enter

```
# Update below with correct subscription Id.
inputSubscriptionId="YOUR-SUBSCRIPTION-ID"
# Set the subscription of interest as active.
az account set --subscription $inputSubscriptionId
# Confirm that the subscription is set as active.
az account show --output table
# Upgrade all monitoring extensions.
az vm extension set -n MonitoringExtensionWindows \
--publisher Microsoft.Azure.Workloads \
--version 1.0.21.0 --ids \
$(az resource list --query "[?(contains(type, \
'Microsoft.Compute/virtualMachines/extensions')) \
&& contains(name, 'MonitoringExtensionWindows')) ].id" \
-o tsv) --force-update
# End of script.
```

=====

FAQ:

1. Sample Input and sample output

Input:

```
inputSubscriptionId="12345678-90ab-cde1-2345-6789abcde123"
az account set --subscription $inputSubscriptionId
az account show --output table
az vm extension set -n MonitoringExtensionWindows --publisher
Microsoft.Azure.Workloads --version 1.0.21.0 --ids $(az resource list
--query "[?(contains(type,
'Microsoft.Compute/virtualMachines/extensions')) && contains(name,
'MonitoringExtensionWindows')) ].id" -o tsv) --force-update
```

Output:

Will get a "running" prompt when the install is in progress.

EnvironmentName	HomeTenantId	IsDefault	Name	State	TenantId
AzureCloud		True	Az WaaS Test 1	Enabled	
/ Running ..					

```
[
{
  "autoUpgradeMinorVersion": true,
  "enableAutomaticUpgrade": true,
  "forceUpdateTag": "d3f2b931-92a6-4982-be0b-8d031e09372a",
  "id": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/opx-e2e-rg/providers/Microsoft.Compute/virtualMachines/opxascsvm1/extensions/MonitoringExtensionWindows,
  "instanceView": null,
  "location": "eastus",
  "name": "MonitoringExtensionWindows",
  "protectedSettings": null,
  "protectedSettingsFromKeyVault": null,
  "provisioningState": "Succeeded",
  "publisher": "Microsoft.Azure.Workloads",
  "resourceGroup": "opx-e2e-rg",
  "settings": {
    "commandToExecute": "Enable",
    "module": "powershell",
    "taskId": "a7c2ef3a-1f06-4221-96d4-d580855a5f58",
    "timeStamp": "0001-01-01T00:00:00Z"
  },
  "suppressFailures": null,
  "tags": null,
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "typeHandlerVersion": "1.0",
  "typePropertiesType": "MonitoringExtensionWindows"
}
]
```

2. Error: “argument --ids: expected at least one argument”

When script execution fails with above error, then the input subscription does not have any impacted resources that need this mitigation. Please ignore.

3. Is there an alternative to using Azure Cloud Shell for running the commands?

Yes. The script can be run on any machine with Azure CLI installed. Please follow the steps below,

- Ensure Azure CLI is installed. If not, follow the documentation to install Azure CLI: [How to install the Azure CLI | Microsoft Docs](#)
- Run "az login" command
- Execute the steps mentioned in Step2, replacing 'YOUR-SUBSCRIPTION-ID' with the actual subscriptionId.

4. Commands ran successfully but the VIS is showing 'FailedToInstallMonitoringExtension' error.

The above error can be seen if the Monitoring extension was in a failed provisioning state before the upgrade. To fix this, you can delete and register the VIS again.