

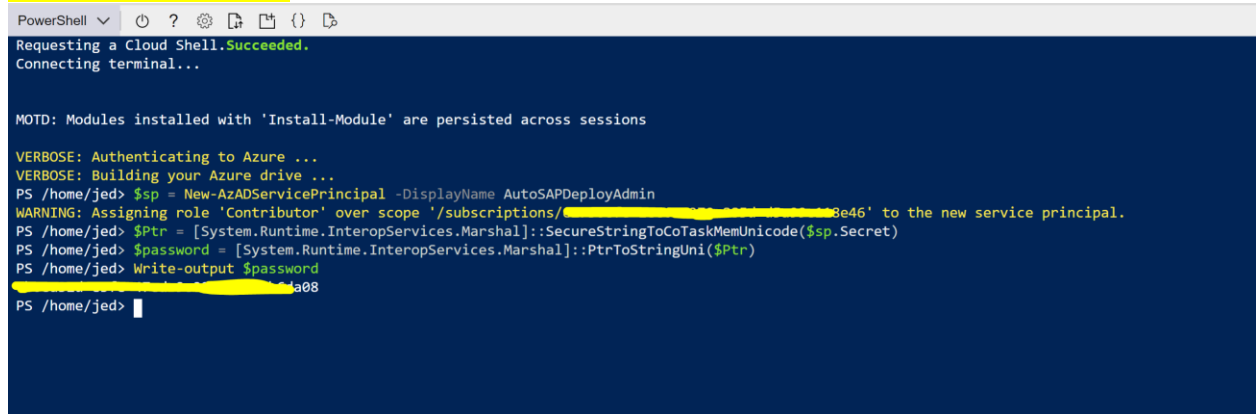
Openhack SAP deployment steps

1. Create a service principle for the deployment in your subscription:

Powershell command: (note: "AutoSAPDeployAdmin" will be the principle name to be used.

You should change it to somename unique like "AutoSAPDeployAdmin_johnsmith")

```
$sp = New-AzADServicePrincipal -DisplayName AutoSAPDeployAdmin
$Ptr =
[System.Runtime.InteropServices.Marshal]::SecureStringToCoTaskMemUnicode($sp.Secret)
$password = [System.Runtime.InteropServices.Marshal]::PtrToStringUni($Ptr)
Write-output $password
```

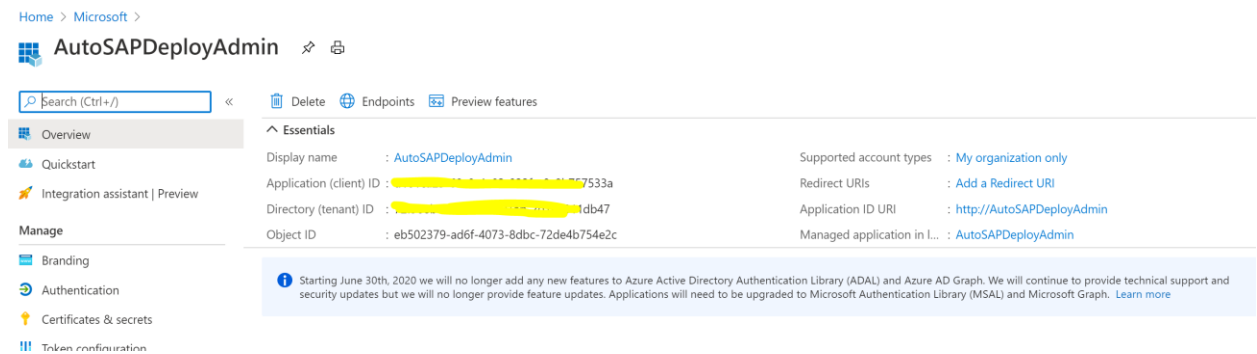


```
PowerShell
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

MOTD: Modules installed with 'Install-Module' are persisted across sessions

VERBOSE: Authenticating to Azure ...
VERBOSE: Building your Azure drive ...
PS /home/jed> $sp = New-AzADServicePrincipal -DisplayName AutoSAPDeployAdmin
WARNING: Assigning role 'Contributor' over scope '/subscriptions/...' to the new service principal.
PS /home/jed> $Ptr = [System.Runtime.InteropServices.Marshal]::SecureStringToCoTaskMemUnicode($sp.Secret)
PS /home/jed> $password = [System.Runtime.InteropServices.Marshal]::PtrToStringUni($Ptr)
PS /home/jed> Write-output $password
AutoSAPDeployAdmin
PS /home/jed>
```

Take down the \$password value. Also take down the following info from Azure portal → Azure Active Directory→App registrations and select the principle "AutoSAPDeployAdmin". Note down the **Application (client) ID** and **Directory (tenant) ID** field.



Home > Microsoft > AutoSAPDeployAdmin

Search (Ctrl+/) Delete Endpoints Preview features

Overview

Quickstart

Integration assistant | Preview

Manage

Branding

Authentication

Certificates & secrets

Token configuration

Essentials

Display name	: AutoSAPDeployAdmin	Supported account types	: My organization only
Application (client) ID	: 7533a...	Redirect URIs	: Add a Redirect URI
Directory (tenant) ID	: 1db47...	Application ID URI	: http://AutoSAPDeployAdmin
Object ID	: eb502379-ad6f-4073-8dbc-72de4b754e2c	Managed application in l...	: AutoSAPDeployAdmin

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

Finally take down your **subscription ID**:

Subscriptions

Microsoft

+ Add

View list of subscriptions for which you have role-based access control (RBAC) permissions to manage Azure resources. To view subscriptions for which you have permissions, see [View subscriptions for which you have permissions](#). Showing subscriptions in Microsoft directory. Don't see a subscription? [Switch directories](#)

My role ⓘ

Status ⓘ

8 selected

3 selected

Apply

Showing 1 of 16 subscriptions ☒ Show only subscriptions selected in the [global subscriptions filter](#) ⓘ

Search

Subscription name ↑↓	Subscription ID ↑↓	My role ↑↓
 jedwu	378-005-1-15-00-118e46	Account admin

- Provision an ubuntu linux server through Azure portal (18.04 LTS, SKU: Standard DS1 v2) with named user "azureuser". Login to the server as the named user "**azureuser**"

```
% mkdir TST200/
```

```
% cd TST200/
```

```
% wget "https://csasapdemo.blob.core.windows.net/saptools/OPHK.tar.gz?sp=r&st=2021-03-31T20:48:14Z&se=2022-04-01T04:48:14Z&spr=https&sv=2020-02-10&sr=b&sig=C2eseMQIGPMZjiYJPTcMguw8ZrXdZOVVHgVfkw90Y%3D" -O ophk.tar.gz
```

```
% gzip -d ophk.tar.gz
```

```
% tar xf ophk.tar
```

```
% ./local_setup_env.sh
```

- Edit the "main.inputs" parameter file within this folder for the following settings:
 - [required] Subscription and principle information → replace all the "xxxxx" with the data taken down from step 1.

```
azure_login:
```

```
subscription_id: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx"
```

```
client_id: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx"
```

```
client_secret: "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
```

```
tenant_id: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx"
```

- [required] Resource prefix setting: please change the "teamxx" to represent your team number. Eg. "team00"

```
resource_prefix: "teamxx"
```

- [required] Resource group information: you can change the "saprg_ophk_teamxx" to the name representing your team.

```
resource_group:
```

```
name: "saprg_ophk_teamxx"
```

```
state: "new"
```

```
region: "westus2"
```

- [optional] SAP system/app build information: you can change the SAP system name to a three-letter name.

`SAP_system_name: "nwp"`

Now save the main.inputs

4. Generate runnable terraform scripts

`% python3 gen_terraform_script.py`

5. Run terraform script to build the Azure infrastructure

`% ./Run_Terraform_Build.sh`

This script typically run 15-20 min and will provision VNET/subnet/PPG/AVset/VM/.. within a resource group named "saprg_ophk_teamxx" (which is configurable within "main.inputs")

Note: Please make sure that you are running this script under "azureuser" account as we observed some permission issue while using root.

Example of a successful run:

```

azureuser@vmsapauto: ~/testdir/TST200
-rwxr-xr-x 1 azureuser azureuser 719 Oct 16 15:54 setup_terraform.sh
drwxrwxr-x 3 azureuser azureuser 4096 Oct 21 01:02 terraform_info
drwxrwxr-x 3 azureuser azureuser 4096 Oct 21 01:01 terraform_run
azureuser@vmsapauto:~/testdir/TST200$ ./Run_Terraform_Build.sh

Initializing the backend...

Initializing provider plugins...
- Using previously-installed hashicorp/azurerm v2.32.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
azurerm_resource_group.saprg: Creating...
azurerm_resource_group.saprg: Creation complete after 0s [id=/subscriptions/e2736efe-1
psaprgopenhack]
azurerm_virtual_network.vnet_siteB: Creating...
azurerm_virtual_network.vnet_siteA: Creating...
azurerm_virtual_network.vnet_siteA: Creation complete after 4s [id=/subscriptions/e273
ups/sapsaprgopenhack/providers/Microsoft.Network/virtualNetworks/vnet-siteA]
azurerm_storage_account.diagacctdiagsitea: Creating...
azurerm_managed_disk.vm-nfs01sapmntdisk01: Creating...
azurerm_subnet.subnet_sapdb_siteA: Creating...
azurerm_managed_disk.vm-db01dbdatadisk01: Creating...
azurerm_managed_disk.vm-db01dbdatadisk02: Creating...
azurerm_managed_disk.vm-app01usrsapdisk01: Creating...
azurerm_managed_disk.vm-db01dbsharedisk01: Creating...
azurerm_public_ip.piplinuxbox: Creating...
azurerm_managed_disk.vm-db01usrsapdisk01: Creating...
azurerm_virtual_network.vnet_siteB: Creation complete after 5s [id=/subscriptions/e273
ups/sapsaprgopenhack/providers/Microsoft.Network/virtualNetworks/vnet-siteB]
azurerm_subnet.subnet_sapapp_siteA: Creating...
azurerm_public_ip.piplinuxbox: Creation complete after 2s [id=/subscriptions/e2736efe-

```

```

azureuser@vmsapauto: ~/testdir/TST200
dbdr-dbdata-disk02]
azurerm_virtual_machine.vm-winbox: Still creating... [9m0s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [9m10s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [9m20s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [9m30s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [9m40s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [9m50s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [10m0s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [10m10s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [10m20s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [10m30s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [10m40s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [10m50s elapsed]
azurerm_virtual_machine.vm-winbox: Still creating... [11m0s elapsed]
azurerm_virtual_machine.vm-winbox: Creation complete after 11m1s [id=/subscriptions/sapsaprg_openhack/providers/Microsoft.Compute/virtualMachines/csa-winbox]

Apply complete! Resources: 75 added, 0 changed, 0 destroyed.

Initializing the backend...

Initializing provider plugins...
- Using previously-installed hashicorp/azurerm v2.32.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
data.azurerm_resource_group.saprg: Refreshing state...
data.azurerm_network_interface.nic-linuxbox: Refreshing state...
data.azurerm_network_interface.nic-db01: Refreshing state...
data.azurerm_network_interface.nic-winbox: Refreshing state...
data.azurerm_public_ip.pipwinbox: Refreshing state...
8700K ..... 96% 302M 0s
8750K ..... 97% 114M 0s
8800K ..... 98% 53.0M 0s
8850K ..... 98% 49.8M 0s
8900K ..... 99% 22.1M 0s
8950K ..... 99% 7.16M 0s
9000K ..... 100% 37.8M=0.2s
2020-10-21 01:19:15 (55.2 MB/s) - 'downloadazcopy-v10-linux' saved [9242772/9242772]
{
  "app01nic_ip": {"sensitive": false, "type": "string", "value": "10.20.30.69"},
  "app02nic_ip": {"sensitive": false, "type": "string", "value": "10.20.30.71"},
  "asc01nic_ip": {"sensitive": false, "type": "string", "value": "10.20.30.70"},
  "db01nic_ip": {"sensitive": false, "type": "string", "value": "10.20.30.132"},
  "dbdrnic_ip": {"sensitive": false, "type": "string", "value": "10.20.30.197"},
  "nfs01nic_ip": {"sensitive": false, "type": "string", "value": "10.20.30.68"},
  "piplinuxbox": {"sensitive": false, "type": "string", "value": "52.183.76.33"},
  "pipwinbox": {"sensitive": false, "type": "string", "value": "52.183.59.138"},
  "winboxnic_ip": {"sensitive": false, "type": "string", "value": "10.20.30.196"}
}

```

At the end of the execution find out the public IP of the window jumpbox and RDP to the window jumpbox (get the public IP show in the redline field from “pipwinbox”): login credential:

azureuser/Welcome!2345

6. Logon to the window jumpbox. Download the following tools and SAP packages: Note, you might want to install and switch to the google chrome browser to download these as MS edge with window defender will block the download.

Putty.exe: <https://csasapdemo.blob.core.windows.net/saptools/putty.exe?sp=r&st=2021-03-05T18:25:36Z&se=2022-03-06T02:25:36Z&spr=https&sv=2020-02-10&sr=b&sig=TBecQMZgVKmD7a%2FCgEJTRdotZcltmXX%2FUtZRDBwIjyc%3D>

SAP GUI 7.60: <https://csasapdemo.blob.core.windows.net/saptools/SAPGUI.ZIP?sp=r&st=2021-03-02T22:27:06Z&se=2022-03-03T06:27:06Z&spr=https&sv=2020-02-10&sr=b&sig=gX3hoi5oNQKwZ3mtRg%2B%2BL01X9Jf%2BYWw7qIU3VjZn6eM%3D>

HANA studio 2.0:

https://csasapdemo.blob.core.windows.net/saptools/SAP_HANA_STUDIO.zip?sp=r&st=2021-03-02T22:26:19Z&se=2022-03-03T06:26:19Z&spr=https&sv=2020-02-10&sr=b&sig=434PUw8cckZyKijmitjhWzfJ8%2B4H601o2cyU2fPGq6Y%3D

7. From the window jumpbox, logon to the linux jumpbox:

Putty session to “teamxx-linux-jumpbox” with the credential “azureuser/Welcome!2345” – please replace “xx” with your team number chosen in step 3.

```
% cd ~azureuser/Current_Deployment
```

```
% cd ansible
```

```
% ./SAP_Ansible_Deploy.sh
```

This script will config and install a complete SAP system and last several hours (typically 2-7 hours) session of server configuration and SAP installations (you can leave the session running in window jumpbox and reconnect late time). For a S4Hana instance fresh install it could take much longer.

While the ansible script is running, you can continue with step 5 on install SAPGUI and SAP HANA studio on Window jumpbox.

8. Login to SAP system or HANA DB through SAPGUI or hana studio...