# Guide for running Load Tests

Prerequisite:

You will need to get the following info to construct the config file used in JMeter script. You can work with OmniChannel Dev team to get the following info.

*"OrgId": "40d48f78-fd28-42db-8c45-983e665c7437",*

*"ACSGlobalCustomerUserId": "8:acs:5f5761ed-8cdc-43f8-b76d-b59df1bfe3e2\_0000000e-3364-a3ff-34fa-ad3a0d00cb14",*

*"ACSGlobalAgentUserId": "8:acs:5f5761ed-8cdc-43f8-b76d-b59df1bfe3e2\_0000000e-3364-a3ec-f4f3-ad3a0d00d0ac",*

*"ACSResourceName": "fe1173d6-8962-455f-a4f0-e2d3807edf78-OCChannels-acs",*

*"ACSAppId": "e09818c0-d76c-4830-b8bf-6d42bfce2e7b",*

*"ACSAppSecret": "XXXXXXXXXXXXXXXXXXX",*

*"ACSAppTenantId": "975f013f-7f24-47e8-a7d3-abc4752bf346",*

*"ConfigDbEndpoint": "oc-prod-public-ger-cfdbaccnt.gremlin.cosmos.azure.com:443",*

*"ConfigDbPrimaryKey": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",*

C2 script

*"widgetid": "7b06f695-80b1-46dd-b8f3-7e616e1d7bce"*

*"orginUrl": "https://oc-cdn-public-ger.azureedge.net"*

*"refererUrl": "https://oc-cdn-public-ger.azureedge.net"*

*"apimurl": "unq40d48f78fd2842db8c45983e665c7-crm16.omnichannelengagementhub.com"*

*"customeContext": "{'QueueToRoute': {'value': 'q1','isDisplayable': false}}"*

*"ACSMessagingToken": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX "*

C1 script

*"apimpath": "unq40d48f78fd2842db8c45983e665c7-crm16.omnichannelengagementhub.com"*

*"occlientappid": "e5a9e51c-7d56-4f4d-96c3-09e314ef4976"*

*"backendappid": "8d2dc28a-7275-409e-a256-85643fe230db"*

*"OrgURL": "https://octestgergeonde.crm16.dynamics.com/"*

Step 1: Modify JMX for ALT

* Modify csv file path to point directly to file name (config will be uploaded in same relative path as jmx) 
* Hard code number of threads based on desired load in thread groups (ALT recommends going with max of 250, you can scale this by adding more engine instances later on) Graphical user interface, text, application

  Description automatically generated
* In User Defined Variables, load any required secrets from environment using beanshell (we will pass these via Key Vault later). Eg; ${\_\_BeanShell( System.getenv("clientsecret") )}. Fill in any other required variablesGraphical user interface, text, application

  Description automatically generated

Step 2: Create Load Test Resource

Go to portal.azure.com -> marketplace

Graphical user interface, text, application

Description automatically generated

Search for “Azure Load Testing” click create

Fill in required details then “Review + Create”

Graphical user interface, application, email

Description automatically generated

Step 3: Create Key Vault

Create a Key Vault and store and required secrets as secrets.

Refer to this documentation [Quickstart - Create an Azure Key Vault with the Azure portal | Microsoft Learn](https://learn.microsoft.com/en-us/azure/key-vault/general/quick-create-portal)

Graphical user interface, application

Description automatically generated with medium confidence

Step 4: Configure ALT access to Key Vault

Refer to this documentation [Use managed identity to access Azure key vault - Azure Load Testing | Microsoft Learn](https://learn.microsoft.com/en-us/azure/load-testing/how-to-use-a-managed-identity?tabs=azure-portal#grant-access-to-your-azure-key-vault)

Enable System Assigned Identity.

Graphical user interface, text, application, email

Description automatically generated

Go to Key vault created in step 3 -> Access Control -> Add -> Role Assignment

Graphical user interface, text, application

Description automatically generated

Pick Key Vault Secrets User. Click Next



Select “Members”

Graphical user interface, text, application

Description automatically generated

Search for the name of your load test resource Select this, click next and then Review + Assign

Step 5: Create and Run Load Test

Under the load test resource you already created, click “create” and select “Upload a Jmeter script”

Graphical user interface, text, application, email

Description automatically generated

Under “Test Plan” select the jmeter and csv files you want to run

Under “Parameters” add secret values, with value being a reference to the secret URI

Under “Load” select engine instances. # of Virtual users will be # Engine stances \* Threads