Optimize 25 Testing at Scale: Tooling



Prepared by

**Zachery Olson**

Solution Architect

Contributors

**List of Contributors**

Revision and Signoff Sheet

Change Record

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author | Version | Change Reference |
| 2/22/2023 | zacolson | 1.0 | Initial draft for review/discussion |
|  |  |  |  |
|  |  |  |  |

Reviewers

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Version Approved | Position | Date |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

Table of Contents

[1 Introduction 5](#_Toc128573904)

[1.1 JMeter 5](#_Toc128573905)

[1.1.1 Pre-Requisites 5](#_Toc128573906)

[1.1.2 JMeter Terms 5](#_Toc128573907)

[1.2 Azure Load Testing 6](#_Toc128573908)

[1.2.1 Azure Load Testing Terms 6](#_Toc128573909)

[1.3 Resources 7](#_Toc128573910)

1. Introduction

There are a variety of tools or services that you can leverage to achieve your at-scale testing goals. Your decision on the tools used to build and execute your load tests may be a matter of preference. Our guidance, and the samples provided through this guidance will be based on Apache JMeter.

Apache JMeter is not a requirement for load testing your application. However, Azure Load Testing uses JMeter scripts to execute tests. Although the guidance will be based on JMeter, the concepts can be applied to your preferred tool.

* 1. JMeter

Apache JMeter is an open-source Java based application designed to test functional behavior and measure performance.

* + 1. Pre-Requisites

Java 8 or higher. It is recommended to install the latest version for security and performance reasons.

If you plan to record HTTPS traffic, then Java SE Development Kit (JDK) is required. JMeter uses the **keytool** utility in JDK to create certificates and record HTTPS traffic.

* + 1. JMeter Terms

The terms and definitions below describe common elements used in JMeter:

**Test Plan** – Contains the series steps that JMeter will execute during a test run. A test plan will consist of at least one Thread Group, Samplers, Controllers, Listeners, and other configuration elements.

**Thread Group** – Controls the number of threads (users), the ramp-up period, and the number of test iterations to execute. All controllers and samplers are executed in the context of a thread group.

**Sampler** – A child element of a thread group that performs the work of JMeter. For example, to send an HTTP request, this element is used to configure and execute the request.

**Post-processor** – A test element that is applied to a particular sampler. A post-processor allows a user to extract values from a server response.

**Extractor** – A type of post-processor that allows the user to extract values from a server response. There are several types of extractors that can be used against a sampler:

* Boundary Extractor
* CSS Selector Extractor
* Regular Expression Extractor
* XPath (and XPath2) Extractor
* JSON JMESPath Extractor
* JSON Extractor

**Assertion** – Used to perform checks on samplers and validate that the server response matches the expected response. Assertions run after post-processors.

**Recording Controller** – Logical container or placeholder to group and store samplers in a test. A recording controller has no impact on a test run.

**HTTP(S) Test Script Recorder** – A proxy server set up by JMeter to intercept and record your actions while you browse your web application. JMeter will place the recorded samples in the Target Controller you choose. If you choose the default option "Use Recording Controller", they will be stored in the first Recording Controller found in the test object tree (so be sure to add a Recording Controller before you start recording).

Refer to the [resources](#Resources) section for more information about JMeter.

* 1. Azure Load Testing

Azure Load Testing is a fully managed load-testing service in Azure that enables you to generate high-scale load. The service simulates traffic for your applications, regardless of where they're hosted. Developers, testers, and quality assurance (QA) engineers can use it to optimize application performance, scalability, or capacity.

A test in Azure Load Testing can be created by using a URL. For more advanced scenarios, a load test can be created by reusing an existing Apache JMeter test script.

* + 1. Azure Load Testing Terms

**Load testing resource** – The Azure load testing resource is the top-level resource for your load-testing activities. This resource provides a centralized place to view and manage load tests, test results, and related artifacts.

**Test** – A test represents a load test for your application. A test is attached to an Azure load testing resource. A test can be created in one of two ways:

* Existing JMeter script
* URL based quick test

Both types of tests are based on a JMeter test plan, and the related components. Azure Load Testing supports all communication protocols that JMeter supports.

**Test Engine** – A test engine is the computing infrastructure that runs the Apache JMeter test script. Multiple instances of a test engine in a test run the JMeter script in parallel. You can scale out your load test by configuring multiple test engine instances.

**Test run** – Represents one execution of a load test. It collects the logs associated with running the Apache JMeter script, the load test YAML configuration, the list of app components to monitor, and the results of the test.

**App Component** – When you run a load test for an Azure-hosted application, you can monitor resource metrics for the different Azure application components (server-side metrics). While the load test runs, and after completion of the test, you can monitor and analyze the resource metrics in the Azure Load Testing dashboard.

**Metrics** – Measurement of performance or aspects of a test execution. There are two types of metrics:

* *Client-side metrics* – measured by test engines. These metrics include the number of virtual users, response time, number of failed requests, or the number of requests per second.
* *Server-side metrics* – Information provided about your Azure-hosted applications. Azure Load Testing integrates with Azure Monitor, including Application Insights and Container insights, to capture details from the Azure services. Depending on the type of service, different metrics are available.

Refer to the [resources](#Resources) section for more information about Azure Load Testing.

* 1. Resources

JMeter Overview - <https://jmeter.apache.org/usermanual/get-started.html>

JMeter Download - <https://jmeter.apache.org/download_jmeter.cgi>

Keytool - <https://cwiki.apache.org/confluence/display/JMETER/TestRecording210>

JMeter Component Reference - [Apache JMeter - User's Manual: Component Reference](https://jmeter.apache.org/usermanual/component_reference.html)

Azure Load Testing - <https://learn.microsoft.com/en-us/azure/load-testing/overview-what-is-azure-load-testing>

Azure Load Testing Concepts - <https://learn.microsoft.com/en-us/azure/load-testing/concept-load-testing-concepts>