

LISA Quick Start Guide

Table of Contents

Overview	2
Test Environment Provisioning	2
Cloning LIS test code from GitHub repositories.....	2
Collecting additional OSS tools	3
Hyper-V server provisioning	3
Linux VM Creation and Provisioning.....	3
Create a snapshot of the VM	5
Convert the private key to a .ppk format	5
Editing the .xml test suite file	5
Run a test suite	6

Overview

Linux Integrated Services Automation, or LISA, is a PowerShell based framework to automate the running of test scripts on Hyper-V hosted Linux VMs. LISA test scripts can run on either the Linux test VM, or on the Hyper-V host. Host side scripts are written in PowerShell. Guest side scripts, scripts that run on the Linux VM, are typically written in Bash.

The LISA PowerShell scripts, test setup scripts, test cleanup scripts, and test case scripts are stored on a public GitHub repository. In addition to the files provided on the GitHub repository, LISA use some OSS tools. For example, LISA uses SSH to communicate with the Linux test VM. The LISA PowerShell scripts explicitly use the Putty SSH utilities pscp.exe and plink.exe.

To run test cases using the LISA framework, you will need to create and provision a test environment. A test environment includes a minimum of one Hyper-V server, and one or more Linux test VMs with the LIS drivers and daemons installed, and the LISA test code from the lis-test GitHub repository. The Hyper-V server, and the Linux test VMs, need to be provisioned to work with LISA.

Test cases are grouped into test suites, and a test suite is run against a target Linux VM. The definitions of test cases, and test suites, are contained in a XML file. When you run a test suite, you invoke lisa.ps1 and specify the name of the .xml file. In addition to test case and test suite definitions, the XML file defines the VM, to run the test on. There is additional information in the XML file such as the name of a local SMTP server to send the status email once the test run completes.

To setup a test environment to use LISA to run LIS test cases, the following needs to be done:

- Provision the Test environment.
 - Provision a Hyper-V server.
 - Provision a Linux VM.
- Clone the lis-test repository from GitHub.
- Install additional OSS tools.
- Edit the test suite XML file.
- Run LISA to run a test suite against a Linux VM.

Test Environment Provisioning

A LISA test environment consists of at least one Hyper-V server, one or more Linux VMs, the LISA framework and test case scripts. Both the Hyper-V server and the Linux VMs need to be provisioned so they work correctly with LISA. Provisioning the test environment includes cloning the LISA framework and test cases, configuring the Windows Server, and creating and configuring the Linux test VMs.

Cloning LIS test code from GitHub repositories

To get a copy of the LIS test code and the LISA test framework, you will need to clone the lis-test repository from GitHub. The following steps assume you have installed a Git client on your Windows server. If you do not have a Git client installed on your Windows Server, you can download a client from github.com:

<https://github.com/msysgit/msysgit/releases/download/Git-1.9.4-preview20140929/Git-1.9.4-preview20140929.exe>

Note: You will want to ensure the directory hosting the git utilities are part of your path.

Once you have a working git client installed, on your Windows Server, start PowerShell as Administrator, and issue the following commands

```
cd ~  
git clone https://github.com/LIS/lis-test
```

This will clone the lis-test repository into a directory named lis-test. For testing on Server 2012 R2 and Server 2012, you will need to change to the .\lis-test\WS2012R2\lisa directory.

Collecting additional OSS tools

Once the Hyper-V server is provisioned, and the GitHub repository is cloned, you will need additional OSS tools. LISA uses Putty. Putty is a SSH client for Windows. The Putty utilities are expected to reside under the lis-test\WS2012R2\lisa\bin directory. The minimum Putty tools to include are following.

pscp.exe	Used for secure copy of files to or from Linux VMs.
plink.exe	Used to execute commands on Linux VMs.
puttygen.exe	Used to create SSH keys, and to convert SSH private keys to the .ppk format.

Hyper-V server provisioning

To provision a Windows server to support the LISA test framework, perform the following tasks:

- Install the Hyper-V role.
- Configure Hyper-V role as follows:
 - Create vSwitches. You will need the following to cover most network test scenarios
 - One external vSwitch, using a network with a DHCP server.
 - One internal vSwitch.
 - Two private vSwitches.
 - Optionally, you may want to specify new VHD Directory and VM Directory rather than using the default values.

Linux VM Creation and Provisioning

For LISA to run test cases on a Linux VM, the Linux VM needs to be created and provisioned. When creating the Linux VM with Hyper-V Manager, the following is the recommended configuration:

- 1024 MB of memory
- 1 CPU
- The boot disk. The default 127GB dynamically expanding VHD/VHDX works well.
- A DVD drive.
- 1 Network Adapter, connected to the external vSwitch.

Next, install a Linux distribution which includes the LIS drivers. In addition to installing a Linux distribution which contains the Linux Integrated Services (LIS) and the LIS daemons, the Linux VM needs to be provisioned as follows:

- Install a SSH daemon
 - Configure the SSH daemon to start on boot.
e.g. `systemctl enable sshd.service`

- Allow root to login via SSH
 - Edit /etc/ssh/sshd_config and set PermitRootLogin to yes
- Use ssh-keygen to create SSH keys. Note: when creating SSH keys for use with LISA, do not password protect them.


```
sudo -s
cd /root
ssh-keygen
```
- Configure the public key by appending it to the .ssh/authorized_keys file.


```
e.g. cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```
- Copy the private to the Hyper-V host that will be running LISA and convert the private key to a Putty Private Key (.ppk). Store the .ppk file under lis-test\WS2012R\lisa\ssh\

Note: See section “Convert the Private Key to .ppk Format” for more detail.
- Install the dos2unix package
- Install the at package (atd daemon).
- Configure the at daemon to start at boot.


```
e.g. systemctl enable atd.service
```
- Verify that your Linux distribution installed the LIS daemons. For example, Ubuntu does not install them by default. You have to manually add the packages. A simple way to check is to look for the KVP daemon. Issue the command:


```
ps aux | grep kvp
```

There should be a 'hv_kvp_daemon' process.

Ubuntu is currently the only supported Linux distribution that does not install the LIS daemons by default. To install the daemons on Ubuntu:

For Ubuntu 14.04

```
apt-get install linux-tools-3.13.0-24-generic linux-cloud-tools-3.13.0-24-generic hv-kvp-daemon-init
```

For Ubuntu 14.10

```
apt-get install linux-tools-3.16.0-23-generic linux-cloud-tools-3.16.0-23-generic linux-cloud-tools-common
```

The above VM provisioning is the bare minimum to allow the VM to work with LISA. Various test cases require additional packages to be installed. You may want to install these packages at this time. The list of packages for the three most common Linux distributions follow. Note that some of the packages will be installed depending on your install options.

RHEL	SLES	Ubuntu
at bridge-utils btrfs-progs crash dos2unix dosfstools e2fsprogs e2fsprogs-libs util-linux gpm	at bridge-utils btrfsprogs crash dos2unix dosfstools e2fsprogs util-linux gpm kdump	At bridge-utils btrfs-tools dos2unix dosfstools e2fsprogs e2fslibs util-linux libgpm2 libaio-dev

iscsi-initiator-utils kdump-utils libaio-utils ntp ntpd parted reiserfs-utils wget xfsprogs	libaio-devel ntp parted reiserfs wget xfsprogs kernel-devel kernel-headers	ntp ntpd parted reiserfsprogs wget xfsprogs build-essential
---	---	---

Create a snapshot of the VM

Once the VM is provisioned, the last step is to shut down the VM and create a Snapshot. While a snapshot is not required, it is highly recommended. Before starting a test run, LISA will attempt to reset the VM to a snapshot named ICABase. This is to ensure the VM is starting from a known good state.

Convert the private key to a .ppk format

The Putty utilities require the private key be in the .ppk format. Copy the private key from the Linux VM, to the lis-test\WS2012R2\lisa\ssh directory. Then run puttygen.exe to convert the private key to a .ppk format.

The puttygen.exe utility is used to convert a private key to the .ppk format. The steps include the following:

- Start puttygen.exe.
- Click the “Conversions” menu item, then select the “Import Key” option.
- In the File dialog box, navigate to the private key you copied from your Linux VM.
- Select the private key, then click the Open button.
- In the puttygen window, click the “Save private key” button.

Note: It is recommend you keep the same file name and only add the .ppk extension.

Editing the .xml test suite file

A LISA test run is driven by a XML file. The XML file defines test cases, test suites, the VMs to run the test suite on, along with some additional setting/information. In the lis-test\WS2012R2\lisa\xml directory are XML files for each LIS feature area. For example, the KVP tests are defined in the KVP_Tests.xml file. Each XML file has four sections:

Global

The <global> section defines settings that are not specific to any test or VM. For example, where to store log files, the name of an SMTP server to use when sending test results.

Test Suites

The <testSuites> section defines one or more test suite. Each test suite lists the test cases which are part of the test suite.

Test Cases

The <testCases> section defines all the individual test cases. Each test case definition lists the test case script, and test parameters, among other things. Some test parameters will need to be updated to reflect your test environment.

VMs

The <VMs> section defines one or more Linux VMs to run the tests. Each VM definition specifies the name of the test suite to run on this VM.

To run the test suite defined in a .xml file, you will need to edit the .xml file and make changes that are appropriate for your environment. The KVP test suite is the simplest since it only requires the Linux VM with the LIS drivers and daemons, the hv_kvp_daemon in particular. To run the KVP test suite in your environment, edit the KvpTests.xml file and make the following two changes:

1. In the <VM> definition at the bottom of the file, update the <vmName> to match the name of your test VM.
2. In the <VM> definition at the bottom of the file, update the <sshKey> to the name of the .ppk file to match the name of the SSH key file you used when provisioning your VM.

Run a test suite

To run a test suite, you invoke the lisa.ps1 script and specify the .xml file to use. The lisa.ps1 script assumes the current directory is the lis-test\WS2012R2\lisa directory. The following is a sample command line that would invoke the KVP tests on the VMs specified in the XML file:

```
.\lisa.ps1 run xml\KvpTests.xml -dbg 3
```