**Environment Setup**

The test environment setup readiness checking involves -

1. OS Set up (Checking and ensuring the readiness of the Linux machines on which the SQL Server, front-end and the back-end applications run)

2. Checking and verifying the system network is active and ready for the testing which involves -

The IP-address, subnet-mask and default gateway (router IP) of the PC network interface card are configured and is connected to the local network on which the router is configured.

3. Verifying all the server are running and the ports are up using the ‘netstat’ command or ‘telnet’ command. The servers and configured ports are -

Tomcat Webserver: 8081(http), 8443(https),

Incoming tcp interface (POSapp server): 3001

Incoming tcp interface (ATMapp server): 4001

Incoming tcp interface (ISO adapter server): 4003

Outgoing tcp interface (ISO adpater client): client port is dynamic, in the inteface configuration it needs to provide the remote server port or dummy server port to which connection should establish.

Outgoing tcp interface (Naigotta client): client port is dynamic

Outgoing tcp interface (MDS client): client port is dynamic

Outgoing tcp interface (Base24 server): client port is dynamic

The Core processer app's TCP-server port: 7878(UI client command on this port)

The Core processer app's API-server port: 9001(Not used currently)

The SQL Server port: 1443(back-end and front-end both connects)

The following ports must be configured in the firewall to perform port-forwarding, as they are supposed to allow connection request from the internet –

* the webserver port on which UI is hosted :to access UI via remote webbrowsers(clients)
* Server ports of the incoming-side tcp interface modules(POS,ATM,ISO) :to access via respective client drivers remotely

4. Verifying that the SQL Server service is running and is accessible through the MSMS (Management Studio client).

5. Verifying the Webserver service is running and the UI is available at the endpoint url (UI is hosted in Apache Tomcat)

6. Starting the back-end application (EFT Switch) and verifying that the application server is listening on the configured TCP server port(7878) to which, the iso-translator interface modules connects, and UI connects while command execution.

7. Verify the datatbase connection with the applications (front and back end apps.):

UI - Check the tomcat catalina.log to verify if sqlserver is started, ( or stderr.log for sqlserver disconnected error)

Back-end - Check the terminal console log

8. Verify that the JMeter test-plans are ready to perform the load test. The plans for ISO message , POS message and ATM message must be ready and loaded in the JMeter binary folder.

9. Verify that the java jdk-1.8 is installed and ensure that -

* System environment variable is added as JAVA\_HOME with value is set to java-root-folder
* Java-root-folder-binary path is added in the path-variable

10. Verify that the java version command is accessible through the general command line. Then ensure that the JDK-1.8(JRE-1.8) is added in the eclipse project class path.

11. Verify that the java package file for the ISO message simulation is imported in the eclipse project and check if the message is working by running the class files of the each outgoing interfaces(Naigota, MDS and B24). For each outgoing interface, the message format for various incoming interfaces (pos,atm,isoadapter) must be tested.

12. Verify that the following tools are installed and available to use -

- Test Management: JIRA, TestRail or Zephyr

- Bug Tracking: JIRA

- Automation: GitHub, Selenium, TestNG.

- Manual Testing: Eclipse IDE, JDk-1.8, Postman

- Performance Testing: J Meter

- Network analysis: Wireshark

- Others: MSOffice, Teams