Running Test Automation and Using the TenAsys Lab Guide

This document is meant to provide a step-by-step guide on running our test automation.

Table of Contents

[Test Automation Architecture Diagram 4](#_Toc86082098)

[Using a Lab Machine 4](#_Toc86082099)

[Using your Local Machine 4](#_Toc86082100)

[Choosing a test and/or Ifw host machine 5](#_Toc86082101)

[Using the IfW/eVM Hosts page 5](#_Toc86082102)

[Reserving the machine 5](#_Toc86082103)

[Using the Configure Host page 5](#_Toc86082104)

[Choosing dRTOS Host Machines 6](#_Toc86082105)

[Using the dRTOS Hosts page 6](#_Toc86082106)

[Reserving the machine 6](#_Toc86082107)

[Selecting the tests 6](#_Toc86082108)

[Specifying the dRTOS Hosts 7](#_Toc86082109)

[Saving your changes 7](#_Toc86082110)

[Choosing how you would like the tests to be executed 7](#_Toc86082111)

[Pick and go 7](#_Toc86082112)

[Executing the tests manually, like a boss. 8](#_Toc86082113)

[Disabling the Chef Client 8](#_Toc86082114)

[Running Chef from PowerShell 8](#_Toc86082115)

[cc 8](#_Toc86082116)

[chef-client 9](#_Toc86082117)

[Monitoring status 9](#_Toc86082118)

[Locally 9](#_Toc86082119)

[Remotely 9](#_Toc86082120)

[#qa-automation 9](#_Toc86082121)

[Viewing the results 9](#_Toc86082122)

[#qa-automation-results 9](#_Toc86082123)

[The Lab Results page 10](#_Toc86082124)

[TestRail 10](#_Toc86082125)

[You are done... unless you are not! 10](#_Toc86082126)

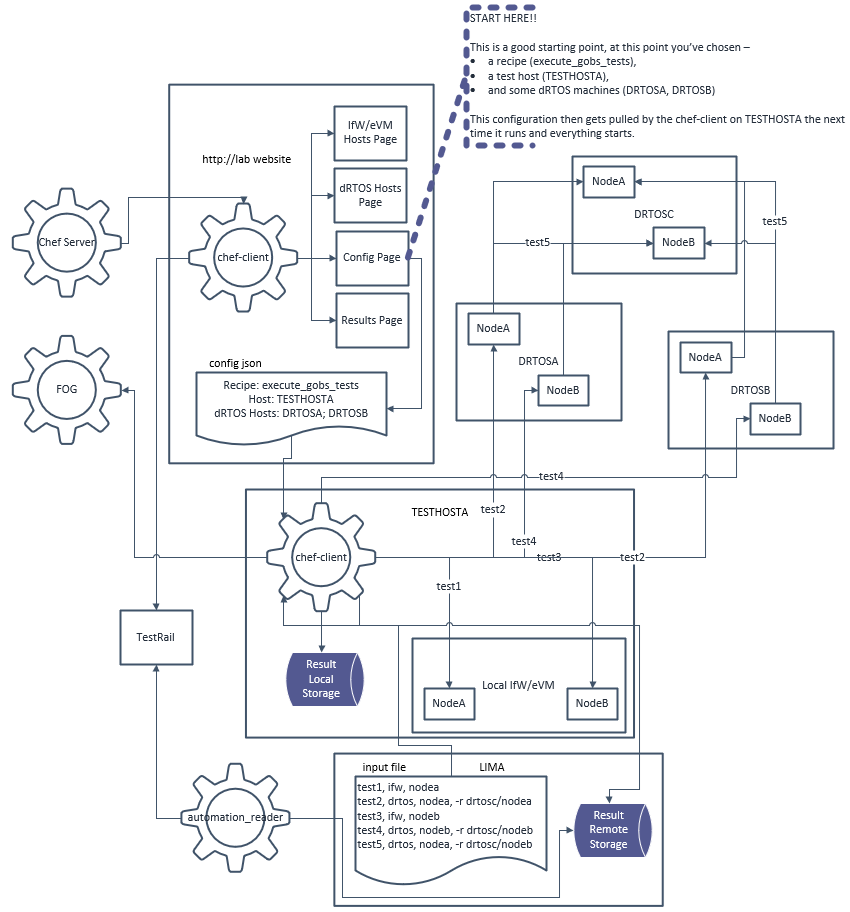
[Modifying the run behavior via attributes 10](#_Toc86082127)

[Overriding the input file 11](#_Toc86082128)

[Overriding the executable 11](#_Toc86082129)

[Pro-Tips 11](#_Toc86082130)

## Test Automation Architecture Diagram



## Using a Lab Machine

This is the most common method for executing test automation – using a lab machine that is already configured for Chef.

## Using your Local Machine

We can install Chef on our dev boxes which has some benefits:

1) Always available

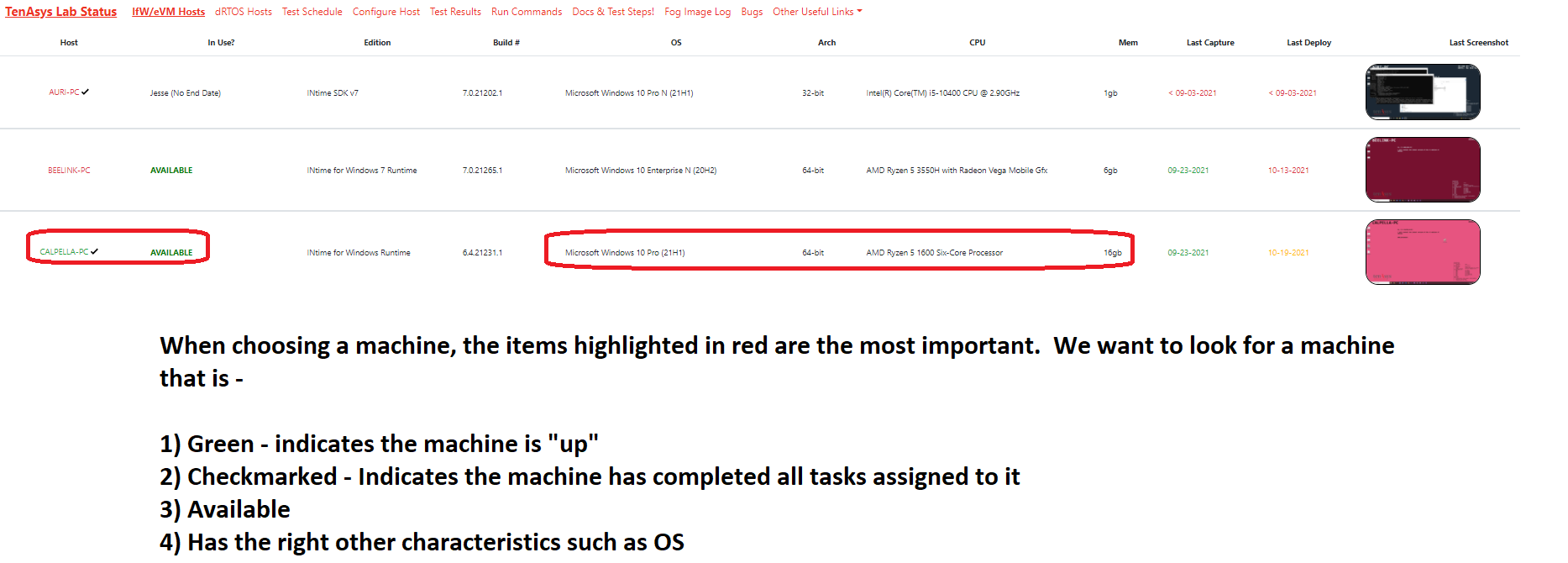
2) Can automate daily tasks such as installing INtime on your dev machine

NOTE – Chef would not be a scheduled task on your dev box. It would only run when manually executed.

More instructions on this method to come, for now, this document assumes you are using a lab machine already configured for Chef.

## Choosing a test and/or Ifw host machine

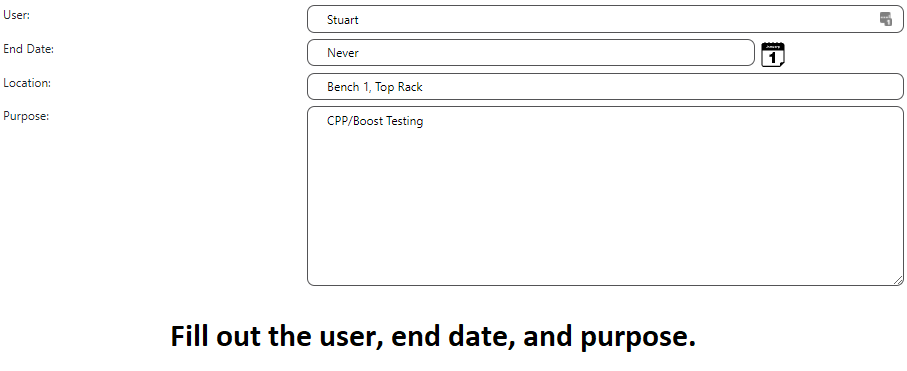
#### Using the [IfW/eVM Hosts](http://lab/) page



### Reserving the machine

Once you find the machine that meets your needs, head over to the “Configure Host” page to reserve it. It’s also good practice to alert everyone in the #lab channel as well.

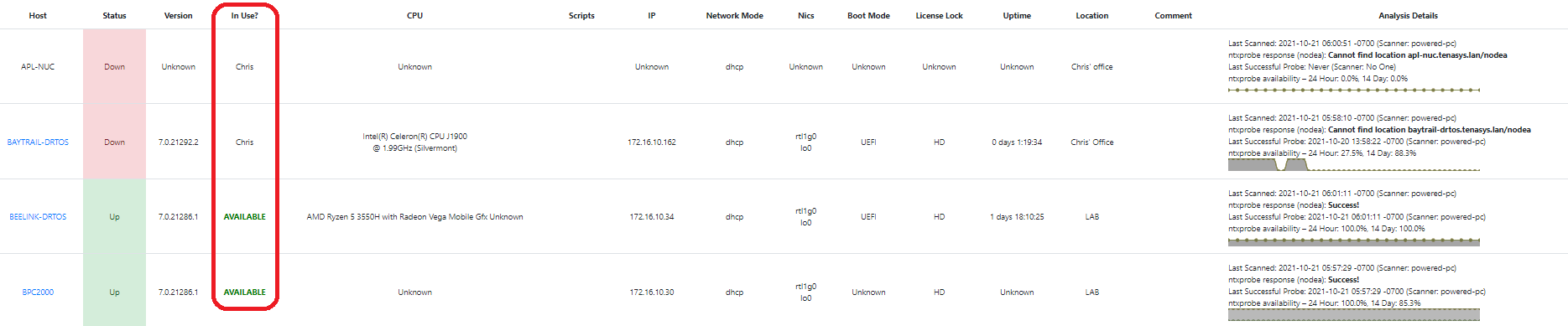
#### Using the [Configure Host](http://lab/config.html) page



## Choosing dRTOS Host Machines

#### Using the [dRTOS Hosts](http://lab/drtos.html) page

Look for an available machine that meets your needs. Review the CPU, and any other information that is useful for your purpose. An example is it might be important whether the machine is configured for DHCP or static network configuration.



*Each lab machine row can be expanded to show additional information such as network interfaces, license details, file versions, and INtime installation details.*

### Reserving the machine

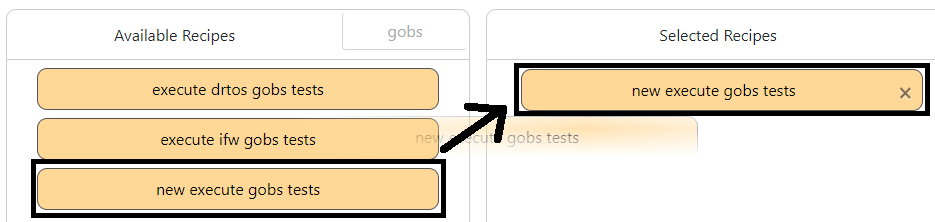
Click the “Edit dRTOS Hosts” button and type your name in the appropriate “USER” spot

Click Save

It’s also good practice to alert everyone in the #lab channel.

## Selecting the tests

On the “[Configure Host](http://lab/config.html)” Page, the available tests are listed in the “Available Recipes” section. There’s a search in the upper-right to quickly find what you want. For example, searching for “gobs” will filter out non-relevant items. To select a test, drag it to the available section. Multiple items can be selected, and they will run from top to bottom.

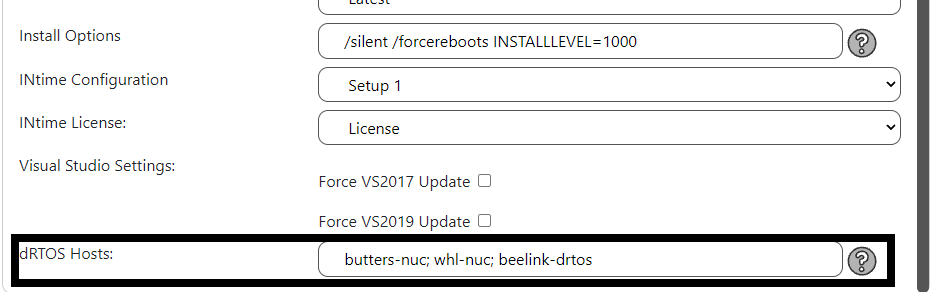


*The underlying technology we use is called “Chef” and we use their terminology of “recipes.” Think of them simply as scripts written in Ruby.*

Recipes handle their own dependencies. For example, we do not need to drag over an “Install INtime” recipe before a test recipe. Only the test recipe is needed. It will install INtime and prepare the machine as necessary prior to beginning test. You would only use the “Install INtime” recipe directly if you wished to have INtime installed but nothing else.

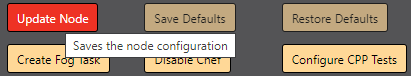
## Specifying the dRTOS Hosts

On the “[Configure Host](http://lab/config.html)” Page, specify one or more dRTOS hosts separated by a semi-colon (;).



## Saving your changes

After everything is configured as you would like, click the “Update Node” button to save your changes.



This brings you to a final screen that allows further customization. Often, no changes will be required, and the “Save Changes” button can be clicked.

## Choosing how you would like the tests to be executed

### Pick and go

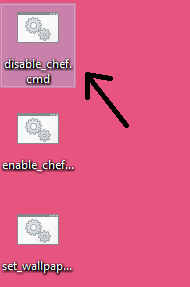
This is the default method for executing tests. Unless deactivated, a scheduled task runs on each lab machine that attempts to run scripts (Chef recipes) every 2 minutes. At this time, if the configuration has been changed via the website, we will perform actions based on the changed. The advantage of this approach is that much of our automation requires several restarts. This is handled nicely when automation is running via this method but must be repeatedly restarted when executing tests manually.

### Executing the tests manually, like a boss.

This method is open ended. You can disable Chef and just use the machine or disable Chef and then manually execute specific Chef commands.

#### Disabling the Chef Client

Double-click on the “Disable Chef Client” batch script located on the desktop of each lab machine.



## Running Chef from PowerShell

##### cc

When the schedule task runs Chef, it effectively runs “cc.” This is the command to use to leverage the <http://lab/config.html> page. It does several things:

1) Pulls down the latest configuration from the website

2) Pulls down all the latest scripts needed to run Chef

3) Updates your PowerShell profile

4) Sets the wallpaper

5) Finally, it then will do what you asked for it to do here...

To run CC:

1) Open PowerShell as an Administrator

2) Run “cc”

##### chef-client

Often, when wanting specific results quickly, it is faster to directly execute Chef than rely on the scheduled task.

To run Chef manually:

1) Disable Chef

2) Run “chef-client -o <recipe> -j <point to JSON config file>”

e.g., “chef-client -o tenasys-test::new\_execute\_gobs\_tests -j $node\_json”



*Hovering your mouse over a recipe will show you the full name.*

## Monitoring status

### Locally

When running Chef locally, we can watch the progress in the PowerShell window. Green text means something has happened.

### Remotely

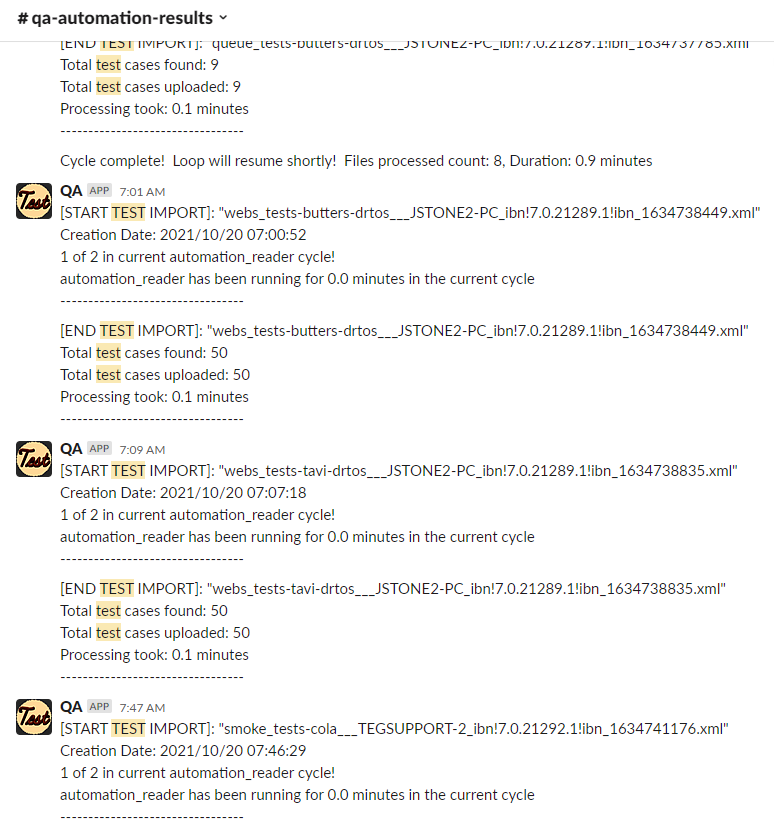
#### #qa-automation

This is a good channel to keep an eye on if you have made changes and just want to monitor what is happening. All lab machines are sent to this channel and provide status.

## Viewing the results

### #qa-automation-results

This is a good channel to keep an eye on if you are expecting results. automation\_reader, the service that processes test results and publishes them to TestRail posts to this channel both at the beginning and conclusion of each test result file.



### The [Lab Results](http://lab/results.html) page

This is the preferred way to view the results. It is significantly faster than the search capabilities of TestRail and provides high-level test result information. As needed, we can go to TestRail and see detailed information by selecting a hyperlink.

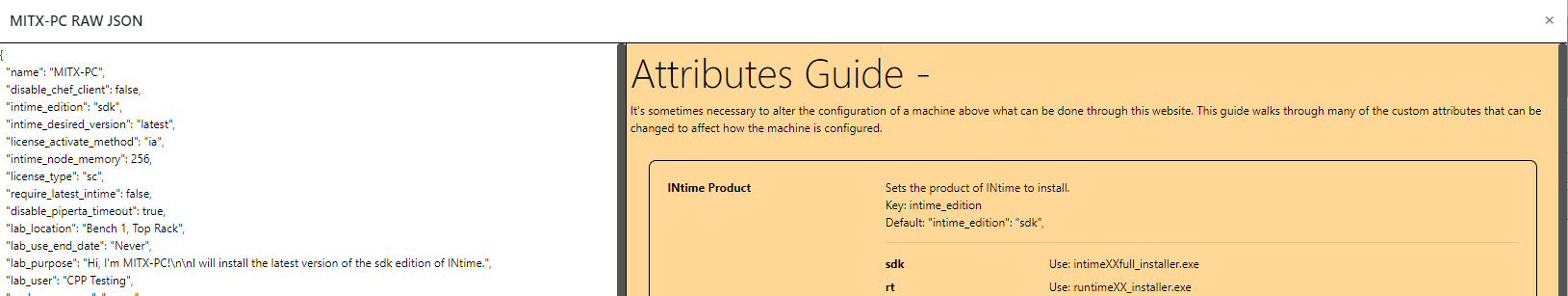
### TestRail

TestRail is our test result repository. It has all analyzed test results (old results do get trimmed) and maintains a history at the test case level for each result.

## You are done... unless you are not!

#### Modifying the run behavior via attributes

When configuring a host via the “Configure Host” page, you have a final page that offers additional customization of your configuration. From this page, you can configure the machine in ways not exposed through the forms on the configuration page. Please read the Attributes Guide document to the right of the “Update Node” page for information on updating attributes.



Many attributes are optional and may not be already included in the machine JSON configuration. In this case, you must add the attribute.

##### Overriding the input file

For our tests that rely on an input file, we will always have a default “global” location that will be used for general testing via the lab. This can be overridden and will commonly be done so for these scenarios:

1) Greater testing around a release (regression input files)

2) Testing to assist developers (specifying custom branches, custom executables)

To change the input file, find or add the attribute and set it accordingly.

e.g., “gobs\_input\_file”: “c:\mycustominput.csv” *NOTE: This will be c:\ on the machine running Chef. Not your local C drive.*

Please read the Attributes Guide document to the right of the “Update Node” page for information on updating attributes.

##### Overriding the executable

All our tests that use an executable can have it overridden by a custom replacement build.

To change the executable, find or add the attribute and set it accordingly.

e.g., “gobs\_executable”: “c:\mycustomrta.rta”

Please read the Attributes Guide document to the right of the “Update Node” page for information on updating attributes.

## Pro-Tips

The test automation framework is only loosely interconnected. For example, changes made to the website will only be prorogated down to the test machine when “cc” is executed in PowerShell (or when the Scheduled Task next runs). Likewise, some information on the website will not be updated until after the next time the test machine finishes execution and publishes additional information.

These are some suggestions on how to circumvent some of these limitations.

* Use Admin PowerShell, not CMD
* The test automation is not foolproof, and it is possible to get the machine into a bad state. For example, I would advise against downgrading the INtime version to an earlier version. Instead, redeploy the machine via FOG first and start fresh.