IxNETWORK - NGPF

Quick  
Reference  
 Guide

Notices

© Keysight Technologies 2021

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Keysight Technologies, as governed by United States and international copyright laws.

Warranty

**THE MATERIAL CONTAINED IN THIS DOCUMENT IS PROVIDED “AS IS,” AND IS SUBJECT TO BEING CHANGED, WITHOUT NOTICE, IN FUTURE EDITIONS. FURTHER, TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, KEYSIGHT DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED WITH REGARD TO THIS MANUAL AND ANY INFORMATION CONTAINED HEREIN, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. KEYSIGHT SHALL NOT BE LIABLE FOR ERRORS OR FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, USE, OR PERFORMANCE OF THIS DOCUMENT OR ANY INFORMATION CONTAINED HEREIN. SHOULD KEYSIGHT AND THE USER HAVE A SEPARATE WRITTEN AGREEMENT WITH WARRANTY TERMS COVERING THE MATERIAL IN THIS DOCUMENT THAT CONFLICT WITH THESE TERMS, THE WARRANTY TERMS IN THE SEPARATE AGREEMENT WILL CONTROL.**

Statement of Compliance.

This product has been designed and tested in accordance with accepted industry standards, and has been supplied in a safe condition. The documentation contains information and warnings that must be followed by the user to ensure safe operation and to maintain the product in a safe condition.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as “Commercial computer software” as defined in DFAR 252.227-7014 (June 1995), or as a “commercial item” as defined in FAR 2.101(a) or as “Restricted computer software” as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause. Use, duplication or disclosure of Software is subject to Keysight Technologies’ standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Safety Notices

The following general safety precautions must be observed during all phases of operation of this instrument. Failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the instrument. Keysight Technologies Inc. assumes no liability for the customer’s failure to comply with these requirements.



A CAUTION notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a CAUTION notice until the indicated conditions are fully understood and met.

  
  
A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

Table of Contents

[**1** Overview 1](#_Toc87904042)

[**2** Configure BGP through GUI: 2](#_Toc87904043)

[Add Chassis and Lock Ports: 2](#_Toc87904044)

[Add Topology: 3](#_Toc87904045)

[Emulate a Protocol: 5](#_Toc87904046)

[Device Group Multiplier: 6](#_Toc87904047)

[Edit Protocol Grid: 6](#_Toc87904048)

[Configure BGP: 6](#_Toc87904049)

[Add Network Group: 7](#_Toc87904050)

[Start Protocols: 8](#_Toc87904051)

[Configure Traffic: 8](#_Toc87904052)

[Add Endpoints to Traffic: 9](#_Toc87904053)

[Edit Packet and Setup Flow Groups: 10](#_Toc87904054)

[Setup Frame Size and Rate: 11](#_Toc87904055)

[Setup Flow Tracking and Protocol Behavior: 12](#_Toc87904056)

[Validate Traffic: 13](#_Toc87904057)

[Apply Traffic, Start Traffic, and Statistics View: 14](#_Toc87904058)

[**3** Configure BGP through Automation (HLPyAPI): 15](#_Toc87904059)

[Initialize Environment: 15](#_Toc87904060)

[Add Chassis and Lock Ports: 15](#_Toc87904061)

[Create Topology: 16](#_Toc87904062)

[Create Device Group: 16](#_Toc87904063)

[Create Ethernet Stack: 17](#_Toc87904064)

[Create Ipv4 Stack: 18](#_Toc87904065)

[Create BGP: 19](#_Toc87904066)

[Create Network Group: 20](#_Toc87904067)

[Start Protocols: 22](#_Toc87904068)

[Enable Filter and Apply Changes ion the Fly: 22](#_Toc87904069)

[Retrieve Learned Info: 23](#_Toc87904070)

[Configure Traffic: 23](#_Toc87904071)

[Start Traffic and Stop Traffic: 24](#_Toc87904072)

[**4** Other Utilities: 25](#_Toc87904073)

[IxNetwork API Documentation Browser: 25](#_Toc87904074)

[Script Gen: 26](#_Toc87904075)

[F1 Option: 27](#_Toc87904076)

[**5** To Know More on NGPF: 28](#_Toc87904077)

[**6** Support: 29](#_Toc87904078)

[Service and Support 30](#_Toc87904079)

# Overview

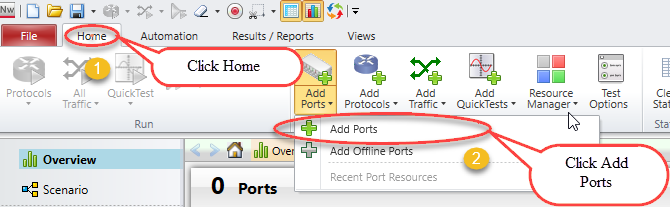
* NextGen Protocol Framework (NGPF) is Ixia’s new protocol framework.
* Upgraded from the classic protocol framework.
* Built to meet and stay ahead of customer requirements in flexibility and scalability.
* Designed to provide consistent and visual workflow across all protocols.
* Designed to more closely simulate dynamic customer environments.
* Industry leading access, routing and SDN protocol coverage.
* Realistic subscriber emulation of mixed single and dual-stack subscribers.
* Flexibility of scaling the number of emulated devices by using the multiplier feature.
* Granular session control by using configuration grids.
* System level statistics dashboards with on-demand drill-downs.
* Comparable feature set with IxN2X.

# Configure BGP through GUI:

This section provides a walk-through of a scenario, which configures BGP emulation manually to get the user introduced to most of the basic features of NGPF.

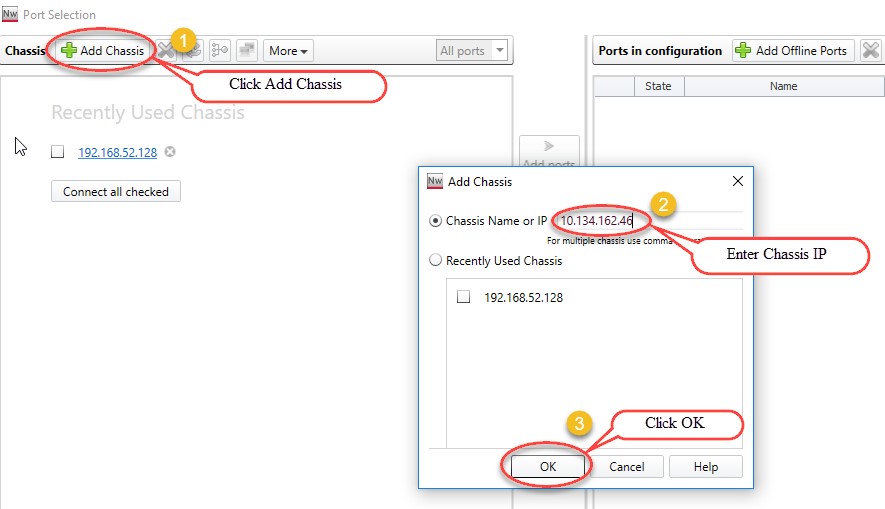
## Add Chassis and Lock Ports:

* The Port Selection window allows you to manage the ports.

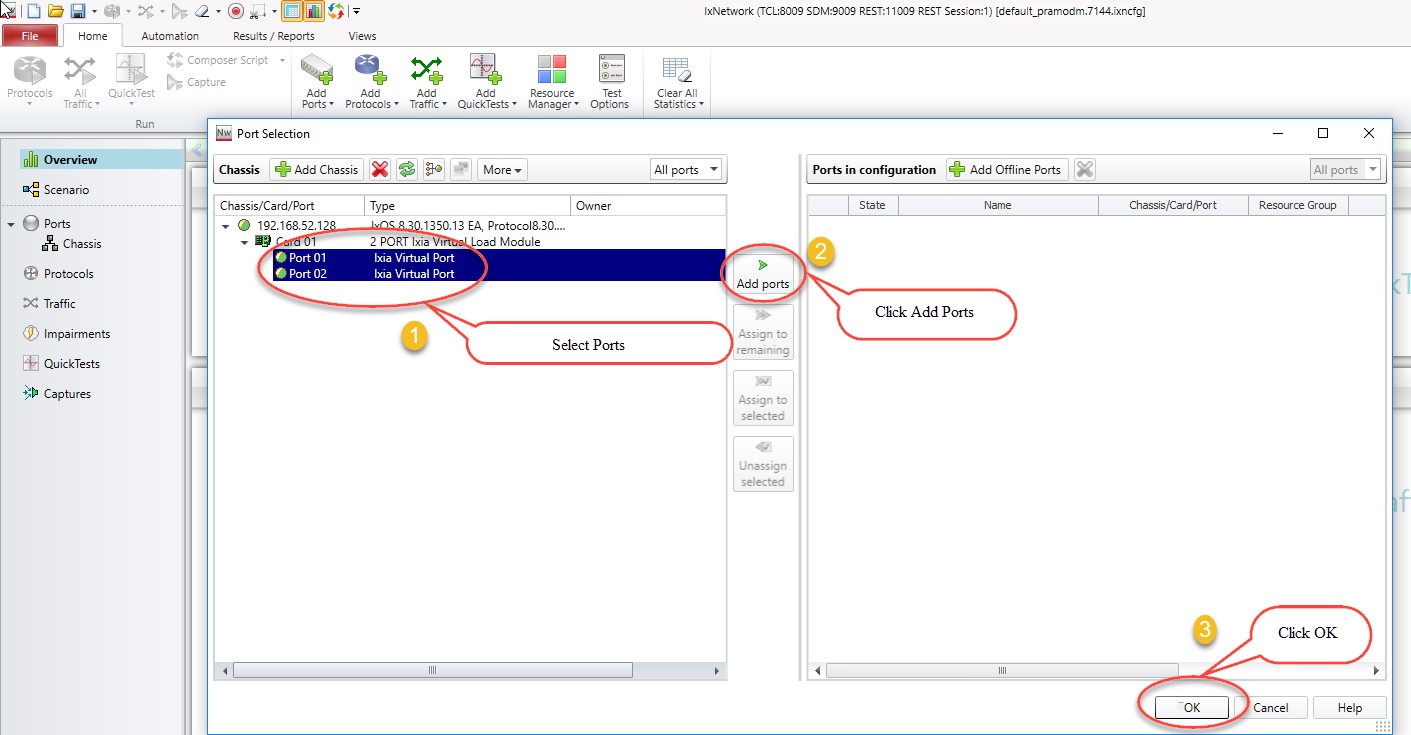


Selecting Ports

* Select chassis by entering chassis IP or select chassis from the list of recently used chassis and then click Connect all checked to add them to the configuration.



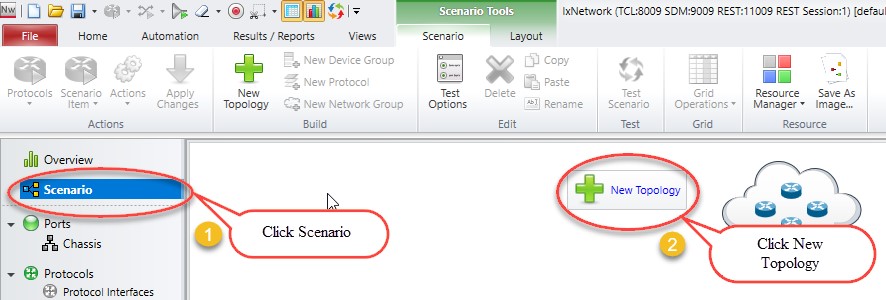
Adding chassis IP



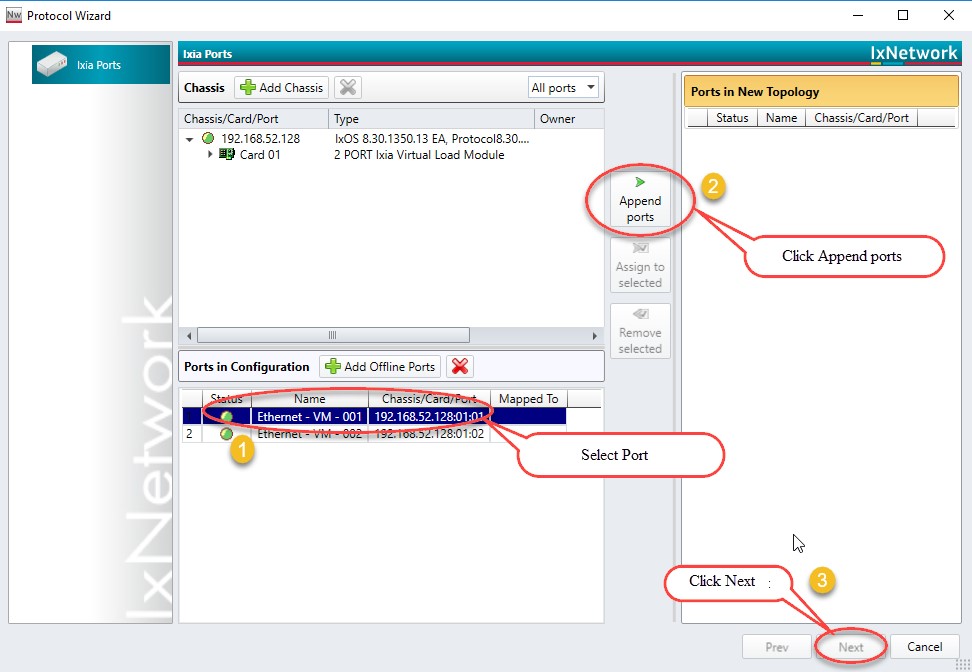
Port selection

## Add Topology:

An IxNetwork instance supports one Scenario, which can contain multiple topologies. Each Topology is a collection of one or more test ports. Each Topology is bound to a virtual port and that virtual port, in turn, binds the physical port.



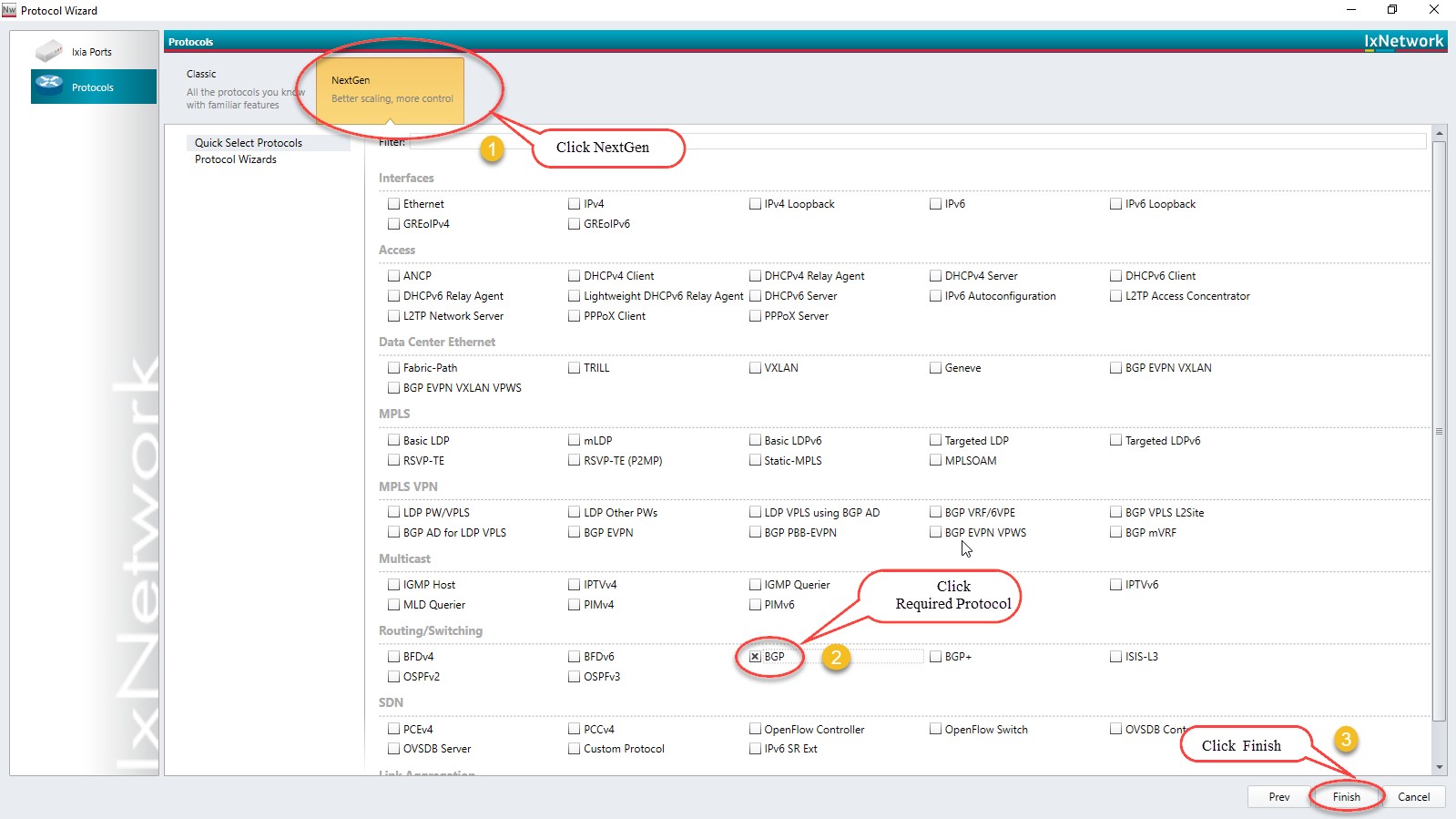
Adding New Topology



Topology with selected ports

## Emulate a Protocol:

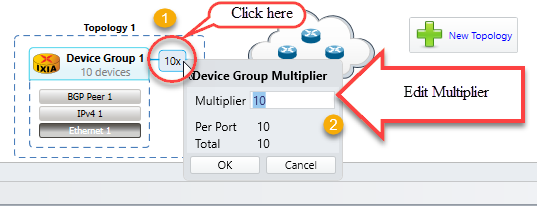
* The Protocols page in the Protocol Wizard allows you to select the protocols in the Topology.
  + The Protocols page lists the available Classic and NextGen protocols under the respective tabs.
  + Click NextGen, and then select the required protocol for the test.
* Presents all supported protocols in Next Generation Protocol Framework in a single window.



Selected Protocol BGP

## Device Group Multiplier:

* A Device Group has similar Devices per test port. A Device can be a router, host, switch, and so on. It can run multiple protocols and protocol stacks.
* A Device Group count is the number of instances in the group. A configuration can be scaled by modeling a group of n Devices per test port by changing the multiplier.

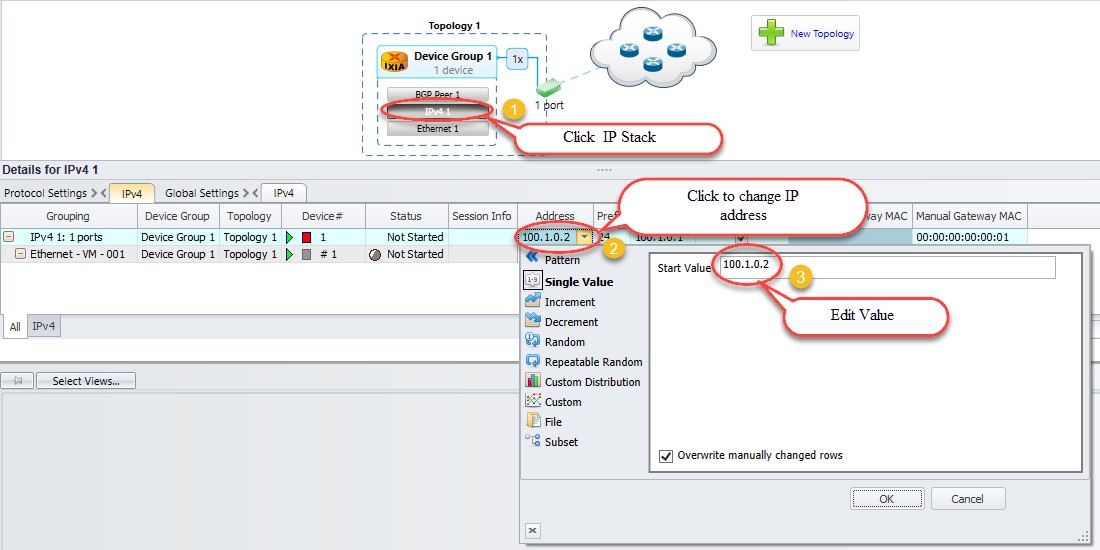


Emulate number of devices by using device group multiplier

## Edit Protocol Grid:

The protocol stack shown in the Scenario view is interactive. Click any protocol stack and edit the values according to the requirement.

**Fig 2.5**



Configuring interfaces

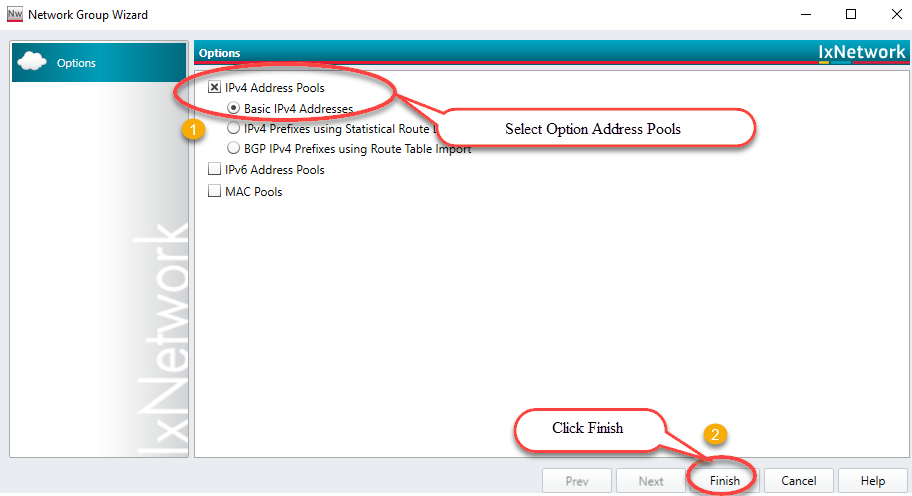
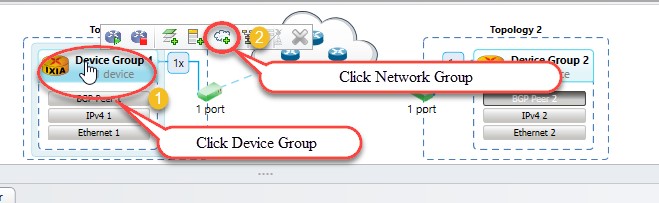
## Configure BGP:

* Configure Interface IP Address to 20.20.20.2 and Gateway Address to 20.20.20.1 in device group 1 IP stack by using method 2.5. Configure Interface IP Address to 20.20.20.1 and Gateway Address to 20.20.20.2 in device group 2 IP stack using by method 2.5.
* Similarly, configure Local IP to 20.20.20.2 and DUT IP to 20.20.20.1 in BGP Stack in device group 1 by clicking the BGP stack. Configure Local IP to 20.20.20.1 and DUT IP to 20.20.20.2 in BGP Stack in device group 2 by clicking the BGP stack.

## Add Network Group:

A Network Group represents a set of L3 networks (sub-netted or switched) with optional information explaining the reachability to each of these networks.

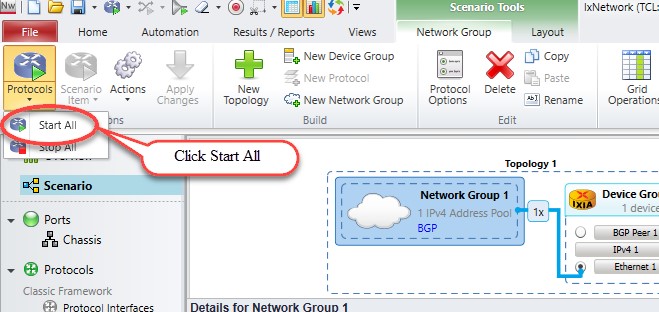
All Devices connected to a Network Group must belong to one of the networks modeled by that Network Group.



Route Profile addition by using network groups

## Start Protocols:

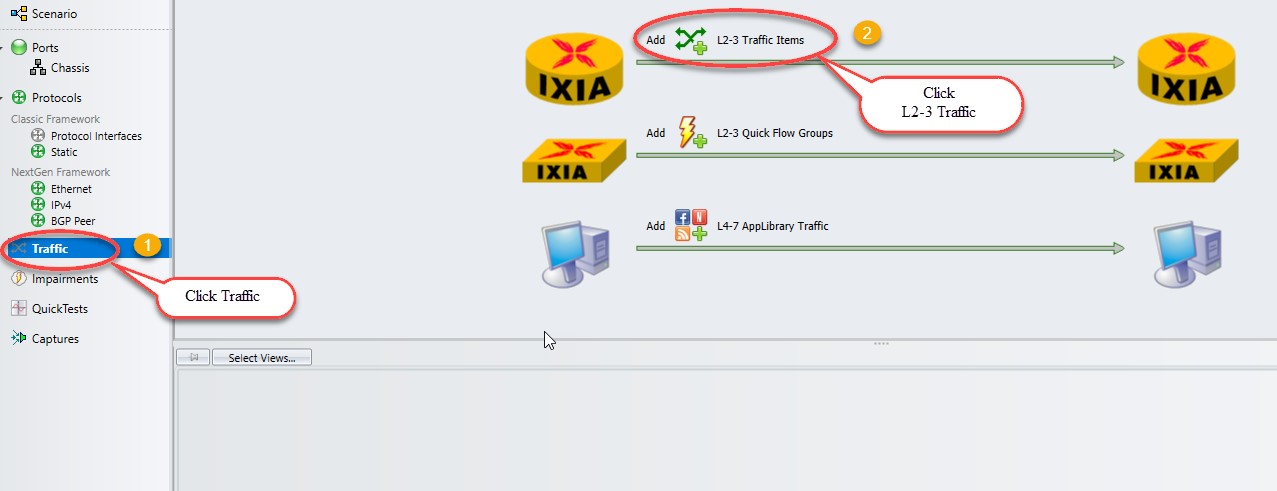
Click Start All to start all the protocols configured in the test.



Brings up all protocol stacks

## Configure Traffic:

* Traffic Wizard helps to integrate the options for traffic configuration in the control plane and data plane of IxNetwork, thereby facilitating the quick setup of large-scale testing.



Configures L2-3 traffic item

## Add Endpoints to Traffic:

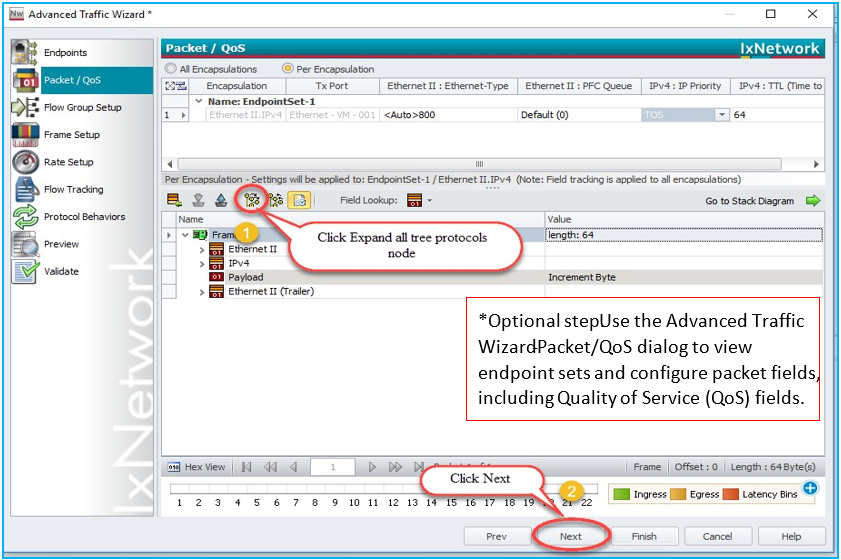
* The Endpoints dialog box is the first dialog box in a series that form the Advanced Traffic Wizard. To access the Endpoints dialog box, click the Endpoints tab in the left pane of the Advanced Wizard window.
* The Endpoints dialog box shows the options to select the traffic endpoints.

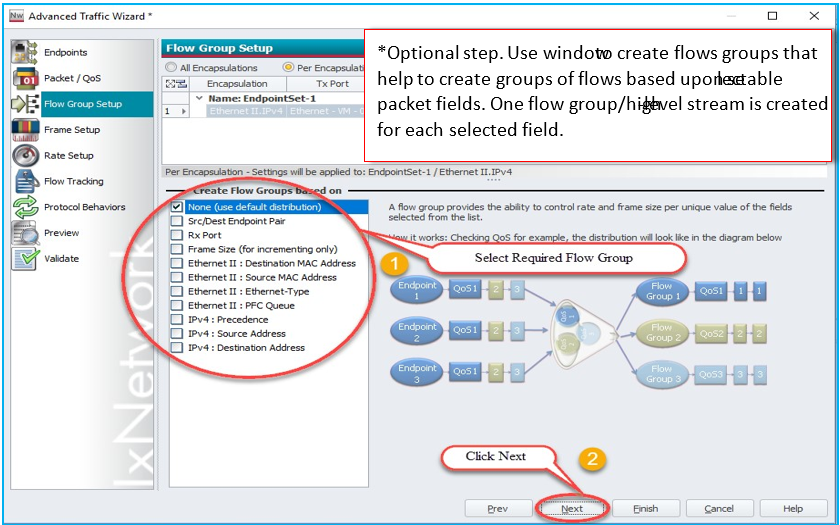


Configures source and destinations endpoints set

## Edit Packet and Setup Flow Groups:

* Editing the packet and setting up flow groups is optional.

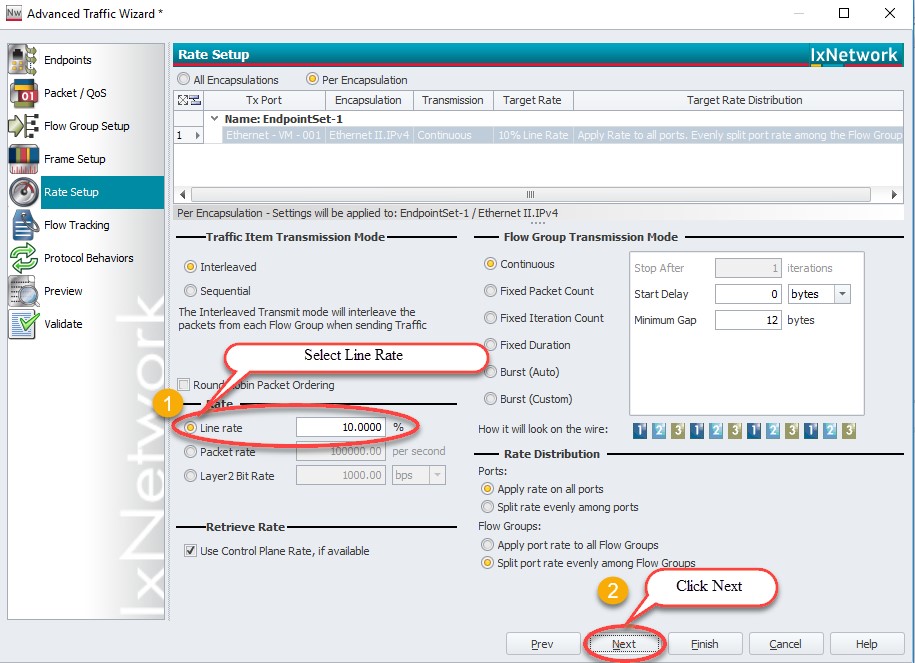
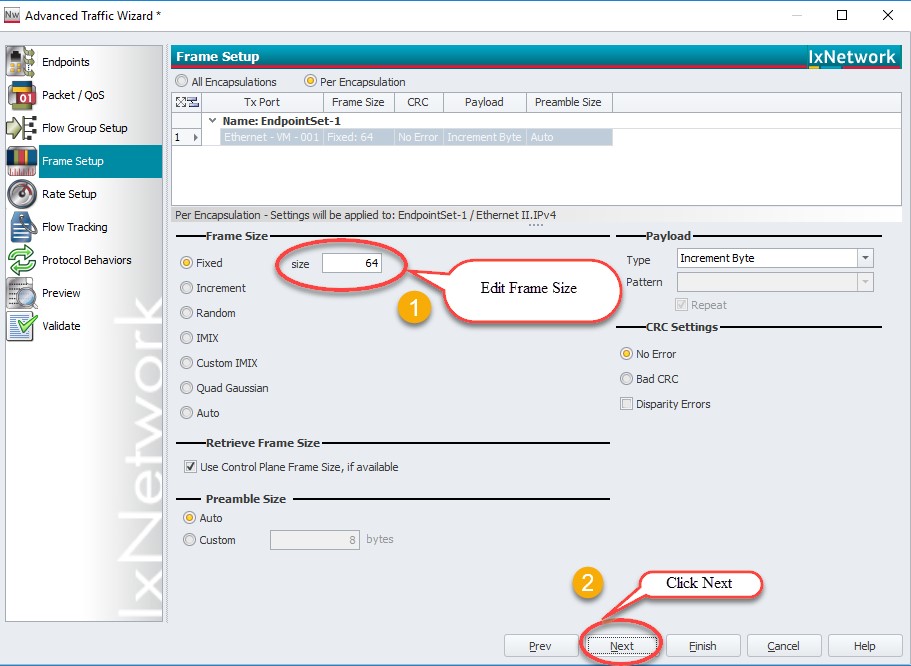




Customizing the packet and creating flow groups

## Setup Frame Size and Rate:

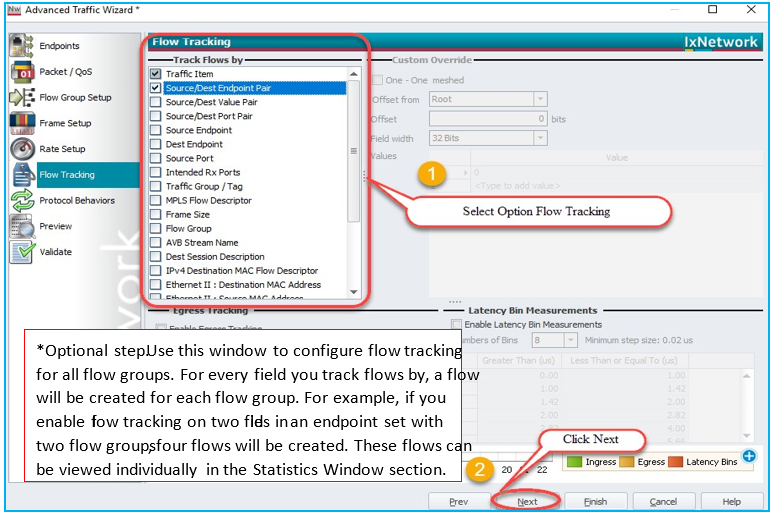
* Setting up the frame Size and Line rate is optional.

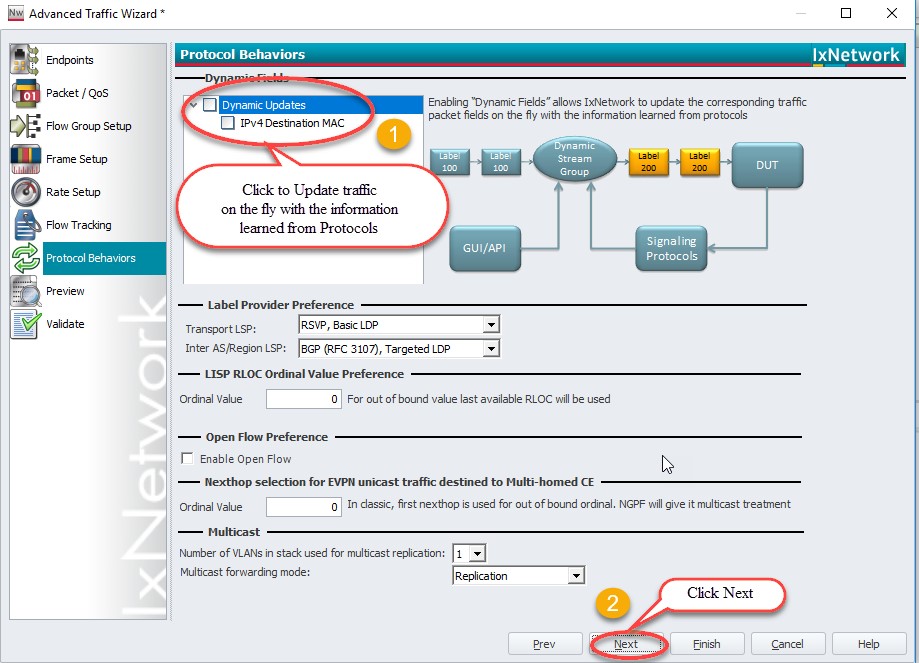


Setting up the frame size and line rate of the traffic

## Setup Flow Tracking and Protocol Behavior:

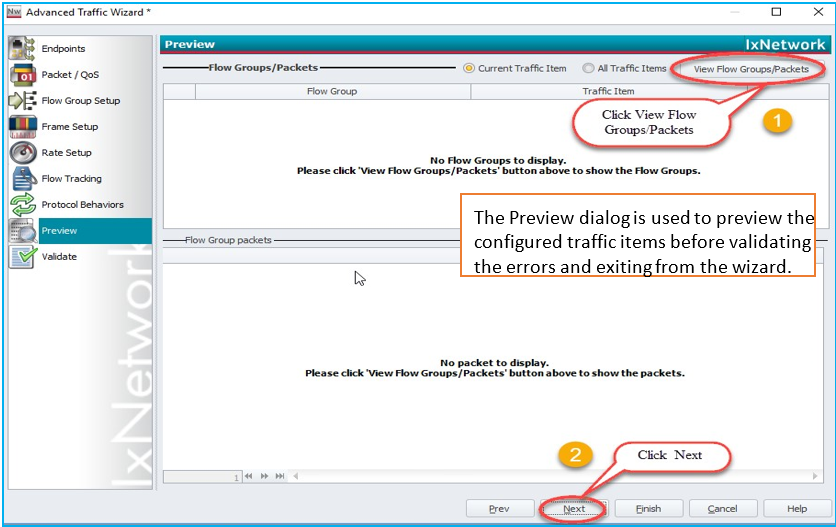
* Setting up the flow tracking and Protocol Behavior is optional.

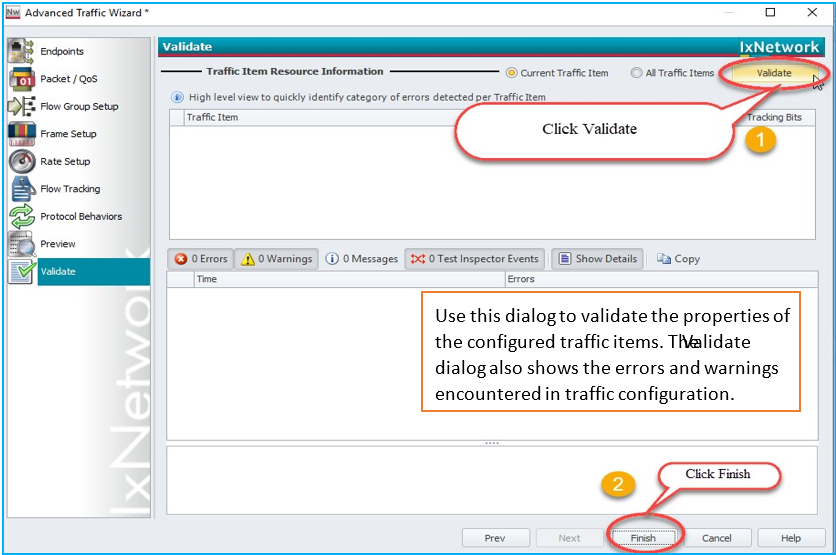




Setting up the flow tracks and traffic update option

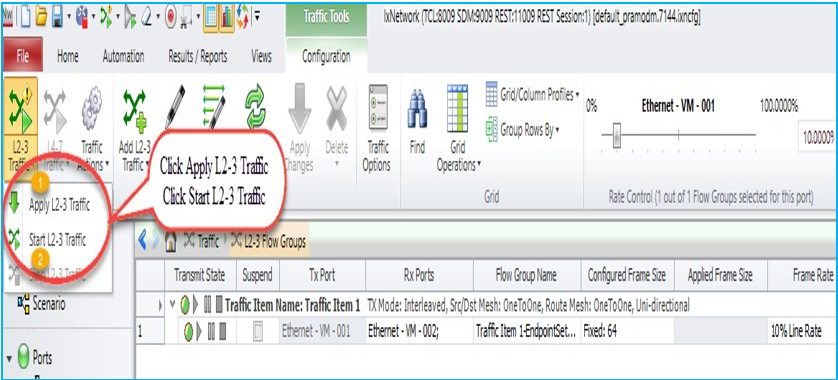
## Validate Traffic:



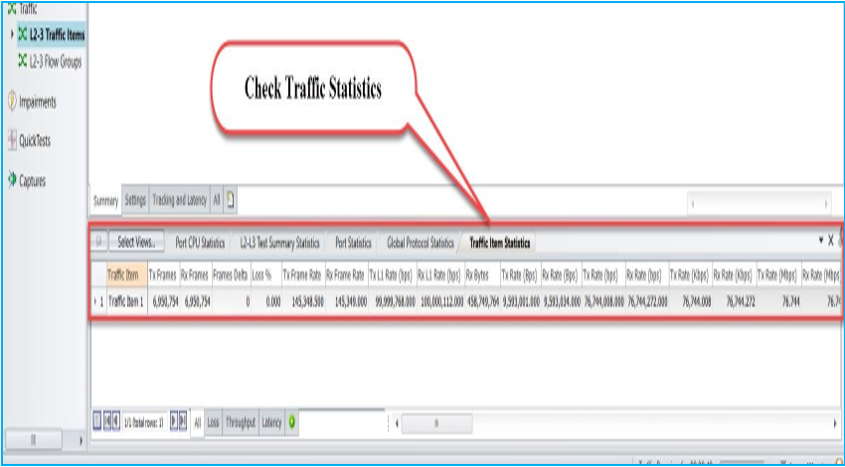


Viewing the flow groups and validating the traffic

## Apply Traffic, Start Traffic, and Statistics View:



Applying and Starting L2-3 Traffic items



Check for traffic item statistics

# Configure BGP through Automation (HLPyAPI):

The REST API is the recommended method for creating new test scripts for IxNetwork.  Please refer to the IxNetwork REST API quick reference guide for more details. The legacy API library continue to be supported for existing test scripts. This section will walk through to reproduce the same BGP emulation scenario through High Level Python API’s to get the user introduced with most of the HLPyAPI’s used in NGPF framework.

## Initialize Environment:

Import the Required Packages and Check for the sanity of the System.

*import sys, os*

*import time, re*

*sys.path.append('C:/Program Files (x86)/Ixia/hltapi/8.40.1123.18/TclScripts/lib/hltapi/library/common/ixiangpf/python/')  
from ixiatcl import IxiaTcl*

*from ixiahlt import IxiaHlt*

*from ixiangpf import IxiaNgpf*

*from ixiaerror import IxiaError*

*ixiangpf = IxiaNgpf(ixiahlt)*

## Add Chassis and Lock Ports:



## Create Topology:

**Ixiangpf.topology\_config:** Adds Topology to the specified port handle and returns the topology handle and Device Group handle which can be used to configure Device Groups.

*topologyConfig1 = ixiangpf.topology\_config (*

*topology\_name = """BGP\_1 Topology""",*

*port\_handle = ports[0],*

*)*

*topology\_1\_handle = topologyConfig1['topology\_handle']*

*deviceGroup1 = ixiangpf.topology\_config (*

*topology\_handle = topology\_1\_handle,*

*device\_group\_name = """BGP\_1 Device Group""",*

*device\_group\_multiplier = "1",*

*device\_group\_enabled = "1",*

*)*

*deviceGroup\_1\_handle = deviceGroup1['device\_group\_handle']*

*topologyConfig2 = ixiangpf.topology\_config (*

*topology\_name = """BGP\_2 Topology""",*

*port\_handle = ports[1],*

*)*

*topology\_2\_handle = topologyConfig2['topology\_handle']*

*deviceGroup2 = ixiangpf.topology\_config (*

*topology\_handle = topology\_2\_handle,*

*device\_group\_name = """BGP\_2 Device Group""",*

*device\_group\_multiplier = "1",*

*device\_group\_enabled = "1",*

*)*

*deviceGroup\_2\_handle = deviceGroup2['device\_group\_handle']*

## Create Device Group:

Device groups added to respective topologies

## Create Ethernet Stack:

**Ixiangpf.interface\_config:** Configures the protocol stack with the Specified Options by using the Device Group Handle and returns the particular protocol stack handle

*interfaceConfig1 = ixiangpf.interface\_config (*

*protocol\_name = """Ethernet 1""",*

*protocol\_handle = deviceGroup\_1\_handle,*

*mtu = "1500",*

*src\_mac\_addr = "18.03.73.c7.6c.b1",*

*src\_mac\_addr\_step = "00.00.00.00.00.00",*

*)*

*ethernet\_1\_handle = interfaceConfig1['ethernet\_handle']*

*interfaceConfig2 = ixiangpf.interface\_config (*

*protocol\_name = """Ethernet 2""",*

*protocol\_handle = deviceGroup\_2\_handle,*

*mtu = "1500",*

*src\_mac\_addr = "18.03.73.c7.6c.01",*

*src\_mac\_addr\_step = "00.00.00.00.00.00",*

*)*

*ethernet\_2\_handle = interfaceConfig2['ethernet\_handle']*

Ethernet stacks added to device groups

## Create Ipv4 Stack:

*ipv4config1 = ixiangpf.interface\_config (*

*protocol\_name = """IPv4 1""",*

*protocol\_handle = ethernet\_1\_handle,*

*ipv4\_resolve\_gateway = "1",*

*ipv4\_manual\_gateway\_mac = "00.00.00.00.00.01",*

*gateway = "20.20.20.1",*

*gateway\_step = "0.0.0.0",*

*intf\_ip\_addr = "20.20.20.2",*

*intf\_ip\_addr\_step = "0.0.0.0",*

*netmask = "255.255.255.0",*

*)*

*ipv4\_1\_handle = ipv4config1['ipv4\_handle']*

*ipv4config2 = ixiangpf.interface\_config (*

*protocol\_name = """IPv4 2""",*

*protocol\_handle = ethernet\_2\_handle,*

*ipv4\_resolve\_gateway = "1",*

*ipv4\_manual\_gateway\_mac = "00.00.00.00.00.01",*

*gateway = "20.20.20.2",*

*gateway\_step = "0.0.0.0",*

*intf\_ip\_addr = "20.20.20.1",*

*intf\_ip\_addr\_step = "0.0.0.0",*

*netmask = "255.255.255.0",*

*)*

*ipv4\_2\_handle = ipv4config2['ipv4\_handle']*

*Creating Ipv4 Stack on topology-1 device group 1:*

*# To create a new IPv4 stack use Ethernet Object*

*ipv4Obj1 = ethObj1.Ipv4.add()*

IPv4 stacks added to ethernet stacks

## Create BGP:

**Ixiangpf.emulation\_bgp\_config:** Configures BGP stack with the specified options by using IPv4 handle and returns the BGP Stack handle

*bgpConfig1 = ixiangpf.emulation\_bgp\_config (*

*mode = "enable",*

*active = "1",*

*handle = ipv4\_1\_handle,*

*remote\_ip\_addr = "20.20.20.1",*

*)*

*bgpIpv4Peer\_1\_handle = bgpConfig1['bgp\_handle']*

*bgpConfig2 = ixiangpf.emulation\_bgp\_config (*

*mode = "enable",*

*active = "1",*

*handle = ipv4\_2\_handle,*

*remote\_ip\_addr = "20.20.20.2",*

*)*

*bgpIpv4Peer\_2\_handle = bgpConfig2['bgp\_handle']*

BGP stacks added to ipv4 stacks

## Create Network Group:

**Ixiangpf.multivalue\_config:** Configures multivalue with specified options by using Device Group handle and topology handle and returns the multivalue handle

*multiValueConfig1 = ixiangpf.multivalue\_config (*

*pattern = "counter",*

*counter\_start = "200.1.0.0",*

*counter\_step = "0.1.0.0",*

*nest\_step = "0.0.0.1,0.1.0.0",*

*nest\_owner = '%s,%s' % (deviceGroup\_1\_handle, topology\_1\_handle),*

*nest\_enabled = "0,1",*

*)*

*multivalue\_4\_handle = multiValueConfig1['multivalue\_handle']*

**Ixiangpf.network\_group\_config:** Configures Network Group with specified options by using Device Group handle, topology handle, ethernet handle and returns the Network Group handle

*networkGroupConfig1 = ixiangpf.network\_group\_config (*

*protocol\_handle = deviceGroup\_1\_handle,*

*protocol\_name = "BGP\_1\_Network\_Group1",*

*multiplier = "1",*

*enable\_device = "1",*

*connected\_to\_handle = ethernet\_1\_handle,*

*type = "ipv4-prefix",*

*ipv4\_prefix\_network\_address = multivalue\_4\_handle,*

*ipv4\_prefix\_length = "24",*

*)*

*networkGroup\_1\_handle = networkGroupConfig1['network\_group\_handle’]*

Adding BGP network group to device group 1

*multiValueConfig2 = ixiangpf.multivalue\_config (*

*pattern = "counter",*

*counter\_start = "201.1.0.0",*

*counter\_step = "0.1.0.0",*

*counter\_direction = "increment",*

*nest\_step = "0.0.0.1,0.1.0.0",*

*nest\_owner = '%s,%s' % (deviceGroup\_2\_handle, topology\_2\_handle),*

*nest\_enabled = "0,1",*

*)*

*multivalue\_10\_handle = multiValueConfig2['multivalue\_handle']*

*networkGroupConfig2 = ixiangpf.network\_group\_config (*

*protocol\_handle = deviceGroup\_2\_handle,*

*protocol\_name = "BGP\_2\_Network\_Group1",*

*multiplier = "1",*

*enable\_device = "1",*

*connected\_to\_handle = ethernet\_2\_handle,*

*type = "ipv4-prefix",*

*ipv4\_prefix\_network\_address = multivalue\_10\_handle,*

*ipv4\_prefix\_length = "24",*

*)*

*networkGroup\_3\_handle = networkGroupConfig2['network\_group\_handle']*

Adding BGP network group to device group 2

## Start Protocols:

**ixiangpf.test\_control**: Start/Stop all the protocols configured in the test session

*testControl = ixiangpf.test\_control (action='start\_all\_protocols')*

*print("Waiting for 45 seconds for the protocols to converge")*

*timer = 30*

*time.sleep(timer)*

Protocol stacks are up

## Enable Filter and Apply Changes ion the Fly:

*bgp\_1\_status = ixiangpf.emulation\_bgp\_config (*

*handle = bgpIpv4Peer\_1\_handle,*

*mode = 'modify',*

*ipv4\_filter\_unicast\_nlri = '1',*

*)*

*applyChanges = ixiangpf.test\_control (*

*handle = ipv4\_1\_handle,*

*action = 'apply\_on\_the\_fly\_changes',*

*)*

Enabling the route filter and applying the changes on the fly

## Retrieve Learned Info:

*bgpLearnedInfo = ixiangpf.emulation\_bgp\_info (*

*handle = bgpIpv4Peer\_1\_handle,*

*mode = 'learned\_info');*

*pprint(bgpLearnedInfo)*

*bgpLearnedInfo = ixiangpf.emulation\_bgp\_info (*

*handle = bgpIpv4Peer\_2\_handle,*

*mode = 'learned\_info');*

*pprint(bgpLearnedInfo)*

Showing details of BGP learned routes

## Configure Traffic:

**ixiangpf.traffic\_config:** Configures the traffic streams on the specified ports with specified options

*trafficConfig = ixiangpf.traffic\_config (*

*mode ='create',*

*traffic\_generator ='ixnetwork\_540',*

*endpointset\_count =1,*

*emulation\_src\_handle =networkGroup\_1\_handle,*

*emulation\_dst\_handle =networkGroup\_3\_handle,*

*track\_by ='sourceDestEndpointPair0 trackingenabled0',*

*rate\_pps =1000,*

*frame\_size =512,*

*)*

L2-3 Traffic configured with the specified options

## Start Traffic and Stop Traffic:

**ixiangpf.traffic\_control:** Start/stop traffic and allows to modify global traffic options

*trafficControl = ixiangpf.traffic\_control (*

*Action ='run',*

*traffic\_generator ='ixnetwork\_540',*

*type =['l23']*

*)*

Running traffic

**ixiangpf.traffic\_stats:** Collect Traffic statistics with the specified options

*protostats = ixiangpf.traffic\_stats (*

*mode = 'all',*

*traffic\_generator = 'ixnetwork\_540',*

*measure\_mode = 'mixed'*

*)*

# Other Utilities:

## IxNetwork API Documentation Browser:

*The main feature of this application is the ability to browse the API meta data in a hierarchical format. Access each level of the hierarchy with a view of siblings, attributes, execs, errors, and children by on clicking on BROWSE.*

IxNetwork API documentation link

## Script Gen:

ScriptGen is a tool that may be used to generate a script that reflects the current configuration of IxNetwork.

It is intended to be used after IxNetwork has been successfully configured. The generated scripts can be used to re-create a configuration as the basis for a new test.



Serialized to the target script file

as a base64 encoded method.

This is a fast method for all

sizes of configuration.

Serialized to the target script file as ixNet commands.

IxNetwork ScriptGen link

## F1 Option:

*Move the mouse pointer over any field in the GUI, and then press F1 to get more information about the field.*

**Fig 4.3** IxNetwork F1 option usability

## 

IxNetwork F1 option usability

# To Know More on NGPF:

* <https://www.youtube.com/watch?v=A0mbZuP94jo>
* <http://openixia.com/sampleScripts//IxNetwork/HighLevelApi/Ngpf/Python>

# Support:

* For more information: <https://support.ixiacom.com/>
* For support assistance, contact: [support.ix@keysight.com](file:///C:\Users\sshimoga\AppData\Local\Microsoft\Windows\INetCache\Content.Outlook\6W27DVNM\support.ix@keysight.com)

### Service and Support

|  |  |  |
| --- | --- | --- |
| **Americas** |  |  |
| Brazil  (11) 4197 3600 | Canada  (877) 894 4414 | Mexico  01800 5064 800 |
| United States  (800) 829 4444 |  |  |

|  |  |  |
| --- | --- | --- |
| **Asia Pacific** |  |  |
| Australia  1 800 629 485 | India  1 800 112 929 | Malaysia  1 800 888 848 |
| China  800 810 0189 | Japan  0120 (421) 345 | Singapore  1 800 375 8100 |
| Hong Kong  800 938 693 | Korea  080 769 0800 | Taiwan  0800 047 866 |
| Other Asian Countries: [www.Keysight.com/find/contactus](http://www.agilent.com/find/contactus) |  |  |

|  |  |  |
| --- | --- | --- |
| **Europe & Middle East** |  |  |
| Belgium  32 (0) 2 404 93 40 | Ireland  1890 924 204 | Spain  34 (91) 631 3300 |
| Denmark  45 45 80 12 15 | Israel  972-3-9288-504/544 | Sweden  0200-88 22 55 |
| Finland  358 (0) 10 855 2100 | Italy  39 02 92 60 8484 | Switzerland  0800 80 53 53 |
| France  0825 010 700\* \*0.125 €/minute | Netherlands  31 (0) 20 547 2111 | United Kingdom  44 (0) 118 927 6201 |
| Germany  49 (0) 7031 464 6333 |  |  |
| **Other Unlisted Countries:** [www.Keysight.com/find/contactus](http://www.agilent.com/find/contactus) |  |  |

© Agilent Technologies, Inc. 2014

Printed in Malaysia

April 2014

For Keysight Internal Reference Only  
Manufacturing Part Number  
