

# CyPerf Test Drive

Testing That Replicates Your Network in Action

The focus of these labs is to provide the users with hands-on experience on how to use CyPerf for a few test scenarios. These scenarios assess the performance and security efficacy of a cloud-deployed network.

## Table of Contents

Overview — Quantifying Cloud Excellence Lab .....	3
Lab 1: Concurrent Connections with HTTP Traffic .....	7
Lab 2: Maximum Throughput with Bi-directional HTTP Traffic .....	10
Lab 3: Jumbo Frames with HTTP Traffic.....	14
Lab 4: Application Traffic Mix .....	17
Lab 5: Security Attacks with HTTP traffic .....	24

# Overview — Quantifying Cloud Excellence Lab

Cloud excellence involves proficiently using, managing, and optimizing cloud resources to deliver value to businesses and end-users. As reliance on cloud services grows, measuring and quantifying excellence becomes crucial for optimal outcomes.

Keysight CyPerf provides organizations with measurable metrics to demonstrate their cloud deployments' effectiveness, security, and resilience.

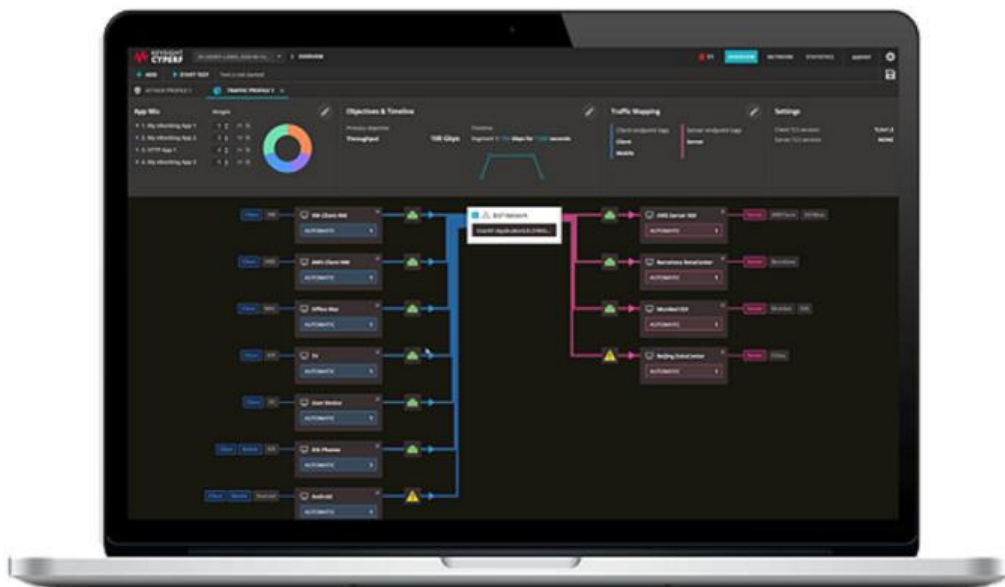
The Test drive gives you:

**Hands-on Experience:** Engage with Keysight CyPerf's intuitive interface and features in a live environment.

**Expert Guidance:** Our team of cloud professionals has put together step-by-step guidance on how to use the test drive

**Strategic Insights:** Post-test drive, receive a detailed report highlighting how CyPerf can enhance your cloud operations

Take advantage of this unique opportunity to harness the power of Keysight CyPerf and steer your cloud operations toward unbridled excellence. Join our test drive and pave the way for an adequate and exemplary cloud infrastructure.



[Keysight CyPerf](#) is the industry's first cloud-native software test solution that recreates every aspect of a realistic workload across a variety of physical and cloud environments to deliver unprecedented insights into end user experience, security posture, and performance bottlenecks of distributed, hybrid networks.

CyPerf delivers new heights in realism that comes from simultaneously generating both legitimate traffic mixes and malicious activities across a complex network of proxies,

software-defined wide area networking (SD-WAN), Secure Access Service Edge (SASE), VPN tunnels, Transport Layer Security (TLS) inspection, elastic load balancers, and web applications firewalls (WAF). Combined with the unique ability to interleave applications and attacks to model user behavior and security breaches, CyPerf enables a holistic approach in replicating distributed customer deployment environments faster and with more fidelity than other solutions.

A cloud-based setup with distributed, lightweight traffic agents that generate realistic application and malicious traffic to assess the performance and security efficacy of cloud-deployed network. The following elements are used in all of the labs described in this test drive.

The main two components of the Lab environment are as follows:

1. The **test tool**: Keysight's CyPerf emulating the malicious and legitimate traffic clients as well as traffic servers (all deployed as cloud instances)
2. The **device under test (DUT)**: you may include any cloud-based network element for testing e.g. web application firewalls WAFs, load balancers, servers, etc.

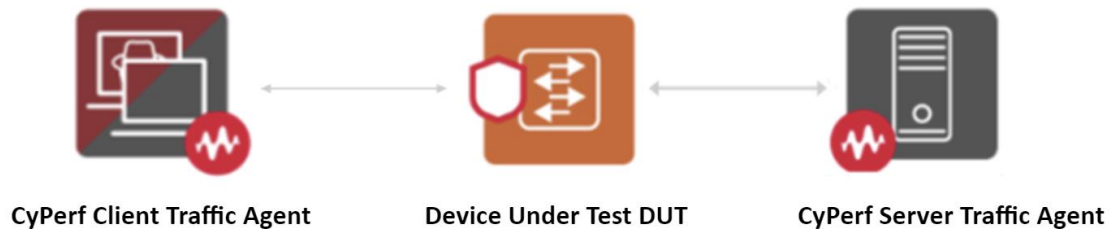


The main components of Keysight's CyPerf (test tool) are as follows:

1. **Test Controller**: web-based UI for configuring and running tests, viewing real-time statistics, and reviewing results.
  - The CyPerf Controller is deployed in the cloud and publicly available to users executing this lab as per <tba>.
2. **Traffic Agents**: software agents generating test traffic.
  - The CyPerf traffic agents (clients and servers) are deployed in the cloud to generate legitimate and malicious traffic going through the cloud-deployed DUT.

## Setup

The following is the high-level diagram of the setup that is used in this lab:



The setup for all the labs consists of the following:

### 1. Traffic Agents:

- **Client:** One CyPerf traffic agent acting as a client deployed in a location.
- **Server:** One CyPerf traffic agent acting as server behind the cloud-based DUT e.g. WAF.

## Important

CyPerf traffic agents (clients and servers) can be virtually deployed in any Region/Zone, across a variety of public clouds (for example, Microsoft Azure, Amazon Web Services, Google Cloud Platform) as well as on on-prem machines to emulate a large-scale distributed network to test the performance and security efficacy of such infrastructures. For more details, see the [product datasheet](#).

## Running tests and labs Setup

The labs in this test drive range from a very basic concurrent connections test to more complex tests such as application traffic and attacks. In every lab we add complexity and observe different KPIs you would need to observe in an end-to-end network.

## Caution

In this Lab, because of cost related considerations, we have forcefully kept one single server agent in the group. In addition, because of the same considerations, although the cloud instances type used for the CyPerf traffic can generate up to 10Gbps, we will limit the maximum throughput per test to 5Gbps.

CyPerf delivers elastically scaling traffic agents that can spawn and tear down dynamically during a test to validate auto-scale policies and enables customers to fine-tune the balance between user experience and security. For more examples and templates for deploying environments with multiple server agents in autoscaling groups, see the CyPerf's public GitHub repo at: <https://github.com/Keysight/cyperf>

## Resources and Prerequisites

To run this lab, users only need access to a common web browser. Everything will be run from a web interface.

All the resources for these labs can be found at the following location:

<https://github.com/Keysight/cyperf/tree/main/CyPerfTestDrive>

This includes the following:

- Configuration files: each lab will start from a configuration file that is available at [\[redacted\]](#).
- Lab's document (this document).
- Intro video: a quick video that guides users on how to spin up and manage the test drive environment.
- Cloud Formation templates
  - Using the Cloud Formation templates found at the preceding location, users can deploy a similar setup with the one from this lab in their own cloud account.
- Terraform script deploys the same environment as the preceding Cloud Formation templates, through a single, aggregated Terraform script.

Looking for more resources? We offer a broad range of additional resources like deployment templates (for major public clouds), associated instructions and REST API wrappers at the following GitHub repository: <https://github.com/Keysight/cyperf>

# Lab 1: Concurrent Connections with HTTP Traffic

## Description

In this lab, we will observe the effect of having several concurrent connections open and transmitting data in our setup.

This test will set up 64,000 connections. These connections will be left open through the duration of the test. Every group of connections will transmit some data every few seconds. The test will rotate through all groups of connections such that all the connections are transmitting some data traffic every few seconds.

This lab targets the CPU and memory resources of the system since we will open 64,000 connections and they will remain open while sending and receiving data traffic, thus also testing the bandwidth of the network. It takes time and resources to open such a large number of connections. If there were any DUT or other devices in the network, e.g. any Layer 4 aware devices such as firewalls / other devices in the network, this lab will test resources for such devices as well, e.g. filtering rules, etc. This lab uses HTTP POST and GET to transmit and receive data between the client and server agents.

## Config

Load the config “Lab 1: HTTP Concurrent Connections with HTTP Traffic”.

Observe:

- The ‘Objectives & Timeline’ tab has been configured to try and achieve 64,000 connections over a duration of 5 minutes.
- The ‘App Mix’ tab has been configured with HTTP GET and POST commands traffic.

Click on ‘Start’ and run the test.

## Result KPIs

While running the test and after the test has completed, please observe the following important KPIs:

- Application connection rate

This metric shows the number of connections initiated/succeeded/failed per second. Here we observe if our setup is able to achieve and sustain 64,000 connections. Opening and keeping all these connections alive while transmitting data through them will stress the memory and CPU resources of our system.

- Application successes/failures

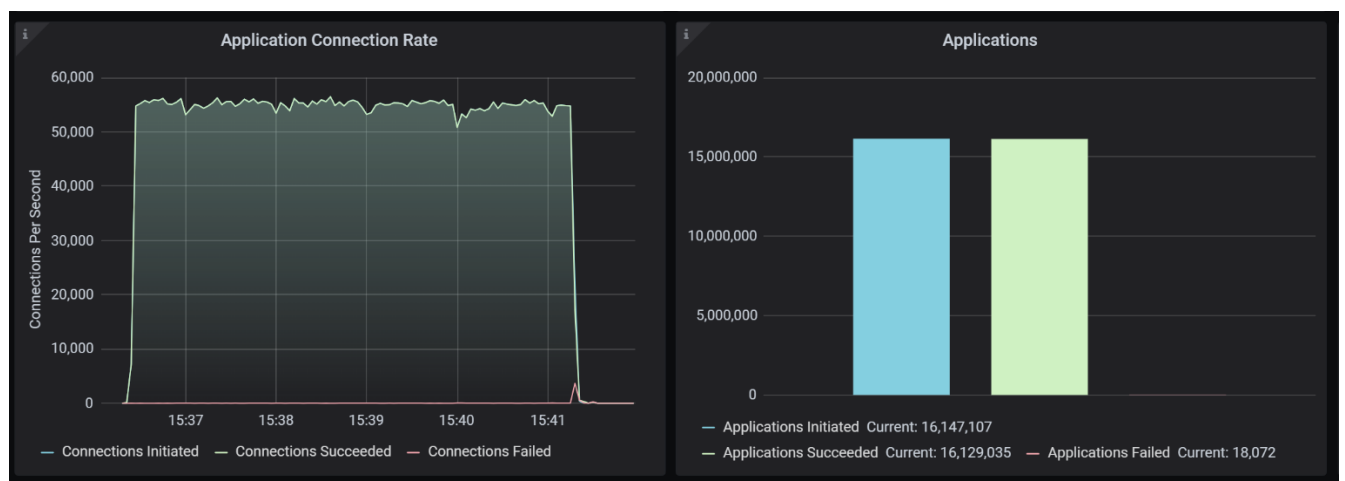
This metric shows the number of applications initiated/succeeded/failed.

- TX/RX Throughput for L23 and Apps

We compare the TX vs RX throughput and observe if there are any drops while the connections per second ramped up and also while they were kept alive.

- Agent resource metrics

This metric shows us if the AWS test agents are operating at full capacity and if so, one needs to increase the limits of the underlying testing infrastructure.

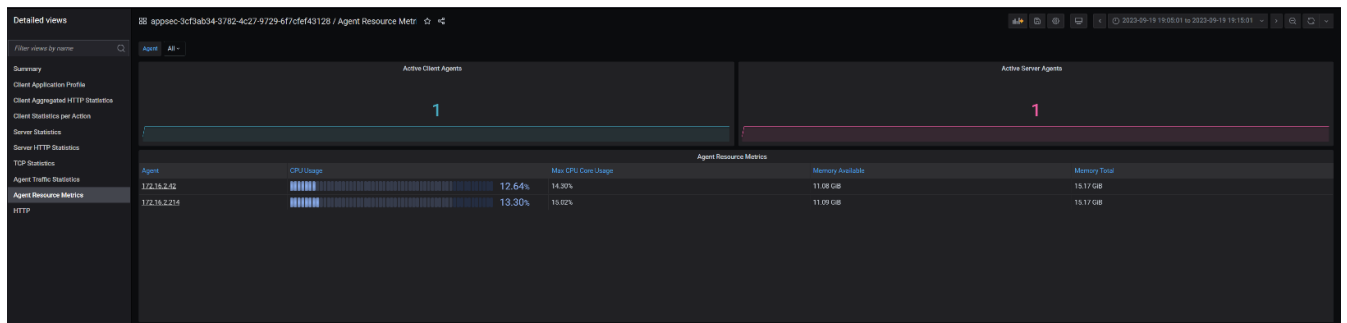


Application connection rate & application successes/failures





Throughput TX/RX



TX/RX test agent resources

## Conclusion

We observe the effect of opening and sustaining 64,000 connections on the CPU and memory resources of our system. This helps us analyze if we need to increase the capacity of our system to be able to handle the load of the network. This test also helps us understand if any of the middle devices in our network need a capacity upgrade.

# Lab 2: Maximum Throughput with Bi-directional HTTP Traffic

## Description

In this lab, we will test the maximum throughput capacity of the network.

## Config

Load the config “Lab 2: Maximum Throughput with Bi-directional HTTP Traffic”.

Observe:

- The ‘Objectives & Timeline’ tab has been configured to try and achieve 1 Gbps of throughput over a duration of 300 seconds.
- The ‘App Mix’ tab has been configured with HTTP POST and GET commands to transmit traffic from the TX agent to the RX agent and from the RX agent to the TX agent.

Click on ‘Start’ and run the test.

## Result KPIs

While running the test and after the test has completed, please observe the following important KPIs:

- Application connection rate  
This metric shows the number of connections initiated/succeeded/failed per second. Here we observe if our setup is able to achieve and sustain connections. Every connection that is opened and kept alive will stress the memory and CPU resources of our system.
- Application successes/failures  
This metric shows the number of applications initiated/succeeded/failed.
- TX/RX Throughput for L23 and Apps  
We compare the TX vs RX throughput and observe if there are any drops while the test is running. This metric helps us characterize the performance of our network.
- Instantaneous latency

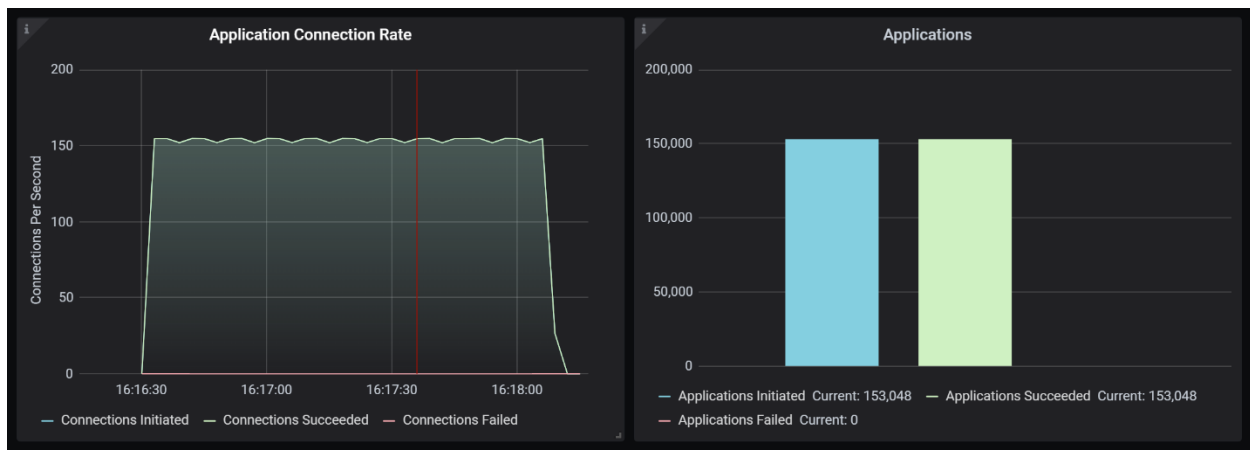
This graph has several sub graphs showing the average connection latency and min/avg/max of the following parameters: time to first byte, time to last byte.

- TCP client and server stats

This metric shows us all TCP handshake and other TCP stats. Watch for retransmissions

- HTTP client and server stats

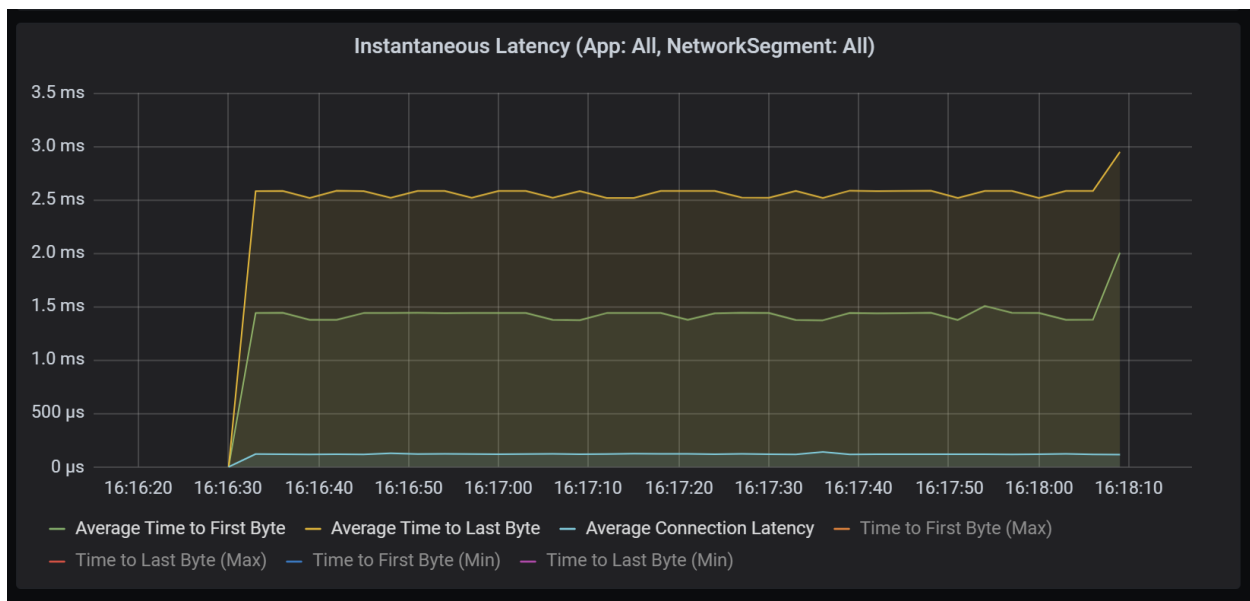
In this metric, we observe HTTP bytes sent/received and other HTTP actions and stats.



Application connection rate & application successes/failures



Throughput TX/RX



Instantaneous latency

TCP Client Statistics (NetworkSegment: All)									
Application Profile - Network Segment	SYN Sent	SYN Received	SYN Failed	Connections Established	Connection Initiation Failed	FIN Sent	FIN Received		
App Client Application Profile 1-IP Network 1	15,312	0	0	15,312	0	15,312	0		
TCP Client Retransmissions Statistics (NetworkSegment: All)									
Application Profile - Network Segment	SYN Retransmitted	SYN Retransmissions Aborted	Data Retransmitted	Data Retransmitted Aborted	SYN-ACK Retransmitted	SYN-ACK Retransmission A			
App Client Application Profile 1-IP Network 1	0	0	0	0	0				
TCP Server Statistics (Port: All)									
Port	SYN ACK Sent	SYN Received	SYN ACK Failed	Connections Established	Connection Initiation Failed	FIN-ACK Sent	FIN Received	FIN Sent	FIN-ACK Receive
80	15,312	15,312	0	15,312	0	15312	15,312	0	

TCP client/server stats

Client HTTP statistics (All)					
Profile - Network Segment	Application & Attack	Requests Completed	Premature Responses Received	Payload Bytes Sent	Payload Bytes Received
App Client Application Profile 1-IP Network 1	HTTP App 1	306,096	0	6,121,920,000	6,125,287,056

HTTP client stats

Server HTTP Statistics (All)				
Application & Attack	Payload Bytes Received	Payload Bytes Sent	Requests Received	Responses Sent
HTTP	6,121,920,000	6,125,287,056	306,096	306,096

HTTP server stats

## Conclusion

We observe the TX vs RX throughput and watch for drops or any performance degradation. This helps us characterize the performance of our network. If there are other devices in the network, observe the drops, if any, at each of these hops.

# Lab 3: Jumbo Frames with HTTP Traffic

## Description

In this lab, we will observe the effect of transmitting jumbo frames in our setup.

Data will be sent over jumbo frames of 3000-3200 bytes through the network.

If you have a DUT or other devices in the setup, set the MTU size to allow packet sizes of up to 4000 bytes on every device/hop through the network.

This lab will test the resiliency of every hop to successfully allow and transmit larger than usual packet sizes.

## Configuration

Load the config “Lab 3: Jumbo Frames”.

Observe:

- The ‘Objectives & Timeline’ tab has been configured to try and achieve 1 Gbps throughput over duration of 300 seconds.
- The ‘App Mix’ tab has been configured with HTTP POST command traffic with 40000 bytes of synthetic traffic. This config will send traffic with frame sizes of ~ 3000 bytes.

Click on ‘Start’ and run the test.

## Result KPIs

While running the test and after the test has completed, please observe the following important KPIs:

- L23 TX/RX Throughput  
We compare the TX vs RX throughput and observe if there are any drops. We also compare the throughput performance of 3000-byte frame traffic vs the previous 1500-byte frames traffic.
- L23 packets sent/received per second  
We compare the number of packets sent vs received by the TX and RX agents. Here we can observe if there are any packet drops.
- L23 average packet sent/received size

This metric shows that jumbo frames were transmitted by the TX test agent, that these jumbo frames traversed the network, and were received by the RX test agent.

- Application connection rate

This metric shows the number of connections initiated/succeeded/failed per second.

- Application successes/failures

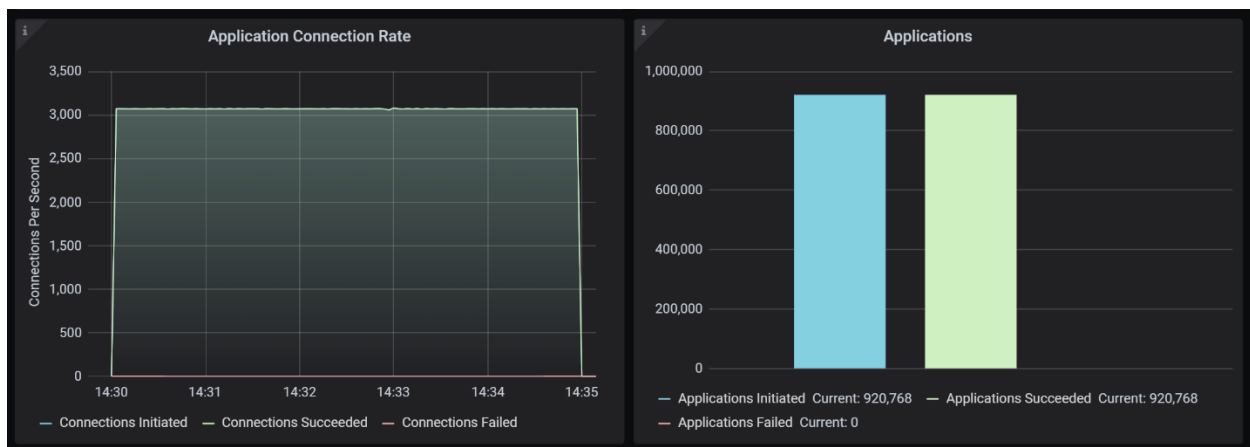
This metric shows the number of applications initiated/succeeded/failed.



Throughput TX/RX



Packets per second & packet sizes



Application connection rate & application successes/failures

## Conclusion

We observe if our test setup can allow and transmit jumbo frames through every hop in the network. The TX vs RX throughput is compared and we analyze if there are any drops in performance or if performance has improved as compared to standard 1500 bytes packet sizes.



# Lab 4: Application Traffic Mix

## Description
















































For this lab, we will use a realistic traffic profile resembling the typical applications traffic flows in an enterprise network.

Many Network Equipment Manufacturers publish the performance figures of their devices in ideal scenarios with either generic, large packet HTTP traffic, or with application mixes that are geared, so that those devices render better performance.

Datasheets are a good starting point, but because each environment is unique, it is paramount to test with an application traffic profile resembling the closest production environment. This ensures that the test results are as relevant as possible to make informed decisions.

For this Lab, the emulated application traffic includes applications like Salesforce, SMTP, Jira, Skype, DocuSign, Office365 Outlook, ADP Internet Explorer 8, AWS Console Firefox, Gmail Chrome.

The complete list of the application traffic and the associated weights is emphasized next:

Applications			 Edit	
<input type="text" value="Search applications by name..."/>				
Active	Name	Weight		
<input checked="" type="checkbox"/>	 Jira Chrome 1	<input type="text" value="1"/>	  	
<input checked="" type="checkbox"/>	 Salesforce Firefox 2	<input type="text" value="1"/>	  	
<input checked="" type="checkbox"/>	 Skype Firefox 3	<input type="text" value="1"/>	  	
<input checked="" type="checkbox"/>	 DocuSign Internet Explorer 4	<input type="text" value="1"/>	  	
<input checked="" type="checkbox"/>	 SMTP 5	<input type="text" value="1"/>	  	
<input checked="" type="checkbox"/>	 Office365 Outlook Microsoft ...	<input type="text" value="1"/>	  	
<input checked="" type="checkbox"/>	 ADP Internet Explorer 8	<input type="text" value="1"/>	  	
<input checked="" type="checkbox"/>	 AWS Console Firefox 9	<input type="text" value="1"/>	  	
<input checked="" type="checkbox"/>	 Gmail Chrome 10	<input type="text" value="1"/>	  	

## Config

Load the config “Lab 4: Application Traffic Mix”.

Observe:

- The ‘Application Profile’ tab has been configured with an multiple realistic application traffics.
- The ‘Objectives & Timeline’ for the application profile tab has been configured to try and achieve 1 Gbps throughput over duration of 300 seconds.
- Click on ‘Start’ and run the test.

## Result KPIs

While running the test and after the test has completed, please observe the following important KPIs:

- L23 TX/RX Throughput

We compare the TX vs RX throughput and observe if there are any drops. We also compare the throughput performance of 3000-byte frame traffic vs the previous 1500-byte frames traffic.

- Application connection rate

This metric shows the number of connections initiated/succeeded/failed per second.

- Application successes/failures

This metric shows the number of applications initiated/succeeded/failed.

- Instantaneous latency for application traffic

This graph has several sub graphs showing the average connection latency and min/avg/max of the following parameters: time to first byte, time to last byte.

- Detailed application stats per application

These results show the number of connections and applications initiated/succeeded/failed, bytes sent/received.

- Detailed application stats per application per action for client and server agents

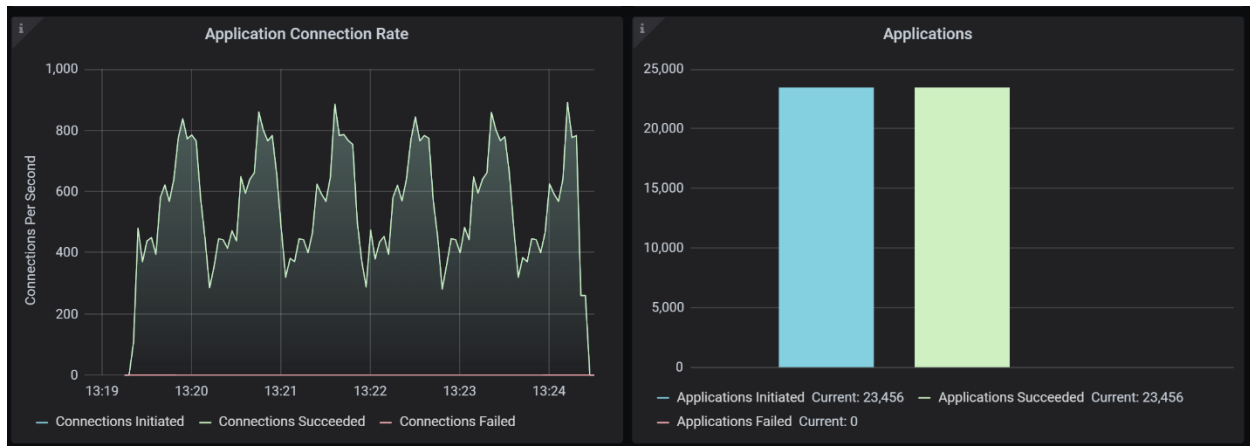
These detailed results show the stats for every action performed by a particular application that was run during the test. This data is available for the client and server sides.

- Application stats drill down

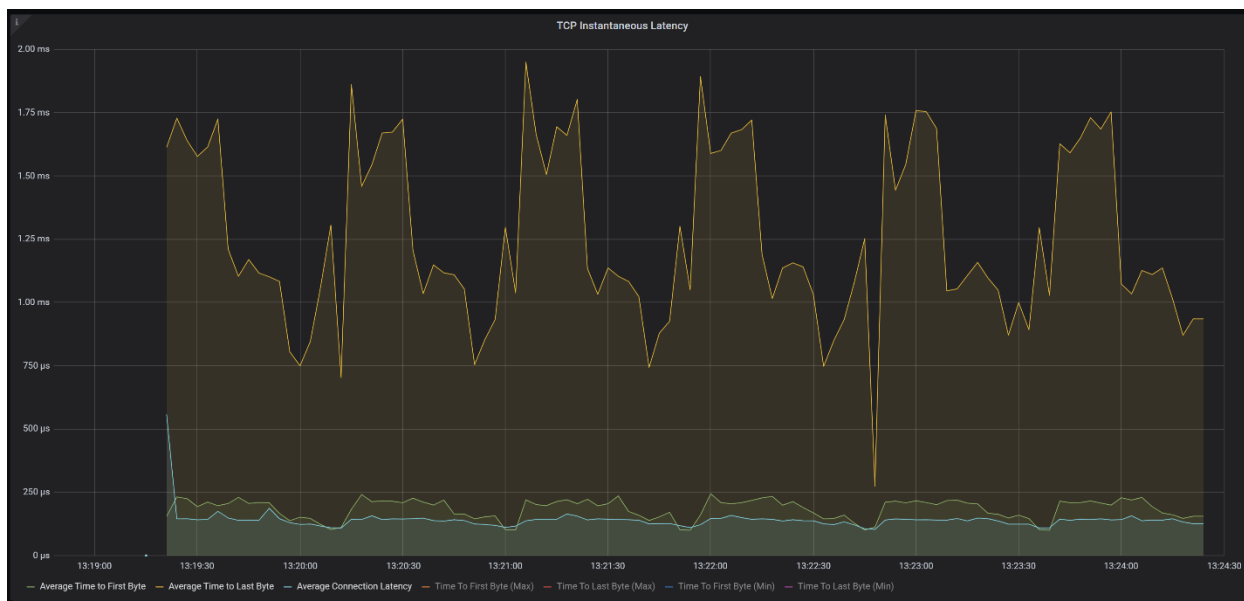
These detailed results show the drilled-down stats for every application that was run as part of the test.



Throughput TX/RX



Application connection rate & application successes/failures



Instantaneous latency

Application Statistics (App: All, NetworkSegment: All)								
Application Profile - Network Segment	Application	Bytes Sent	Bytes Received	Connections Initiated	Connections Succeeded	Connections Failed	Connections Aborted	Applications
App Client Traffic Profile-IP Network 1	Skype Firefox 3	757,337,118	3,535,920,960	34,944	34,944	0	0	
App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	1,206,674,929	3,174,951,984	3,648	3,648	0	0	
App Client Traffic Profile-IP Network 1	SMTP 5	3,768,730,608	6,593,776	8,336	8,336	0	0	
App Client Traffic Profile-IP Network 1	Office365 Outlook Microsoft Edge 7	1,468,956,840	2,880,698,688	21,312	21,312	0	0	
App Client Traffic Profile-IP Network 1	Jira Chrome 1	1,811,669,498	2,479,246,992	29,328	29,328	0	0	
App Client Traffic Profile-IP Network 1	Gmail Chrome 10	149,256,434	4,031,847,552	1,152	1,152	0	0	
App Client Traffic Profile-IP Network 1	DocuSign Internet Explorer 4	106,173,520	3,844,187,488	16,064	16,064	0	0	
App Client Traffic Profile-IP Network 1	AWS Console Firefox 9	1,188,425,109	3,033,463,288	37,560	37,560	0	0	
App Client Traffic Profile-IP Network 1	ADP Internet Explorer 8	117,049,360	3,944,399,576	19,344	19,344	0	0	

Detailed application stats per application

Summary	Action Statistics (AppAttack: All)					
Client Application Profile	Profile - Network Segment	Application & Attack	Action	Started	Succeeded	Failed
Client Aggregated HTTP Statistics	App Client Traffic Profile-IP Network 1	Skype Firefox 3	Voice Call	1,248	1,248	0
Client Statistics per Action	App Client Traffic Profile-IP Network 1	Skype Firefox 3	Video Call	1,248	1,248	0
Server Statistics	App Client Traffic Profile-IP Network 1	Skype Firefox 3	Logout	1,248	1,248	0
Server HTTP Statistics	App Client Traffic Profile-IP Network 1	Skype Firefox 3	Login	1,248	1,248	0
TCP Statistics	App Client Traffic Profile-IP Network 1	Skype Firefox 3	End Voice Call	1,248	1,248	0
Agent Traffic Statistics	App Client Traffic Profile-IP Network 1	Skype Firefox 3	End Video Call	1,248	1,248	0
Agent Resource Metrics	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Update Call Log	912	912	0
ADP	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Select an Opportunity	912	912	0
AWS Console	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Select Top Deal	912	912	0
DocuSign	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Select Opportunities Tab	912	912	0
Gmail	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Select Notes Tab	912	912	0
Jira	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Select Dashboards Tab	912	912	0
Office365 Outlook	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Select Calendar Tab	912	912	0
SMTP	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Open Adoption Dashboard	912	912	0
Salesforce	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Logout	912	912	0
Skype	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Login	912	912	0
	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Load Login Page	912	912	0
	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Edit a Note	912	912	0
	App Client Traffic Profile-IP Network 1	Salesforce Firefox 2	Edit Amount	912	912	0

Detailed application stats per application per action for client agent

Application & Attack Statistics (All)					
Application & Attack	Bytes Sent	Bytes Received	Connections Received	Strikes Allowed	Strikes Initiated
Skype	3,535,920,960	757,337,118	34,944	0	0
Salesforce	3,174,951,984	1,206,674,929	3,648	0	0
SMTP	6,593,776	3,768,730,608	8,336	0	0
Office365 Outlook	2,880,698,688	1,468,956,840	21,312	0	0
Jira	2,479,246,992	1,811,669,498	29,328	0	0
Gmail	4,031,847,552	149,256,434	1,152	0	0

Detailed application stats per application per action for server agent

Detailed views

Filter views by name

Summary

Client Application Profile

Client Aggregated HTTP Statistics

Client Statistics per Action

Server Statistics

Server HTTP Statistics

TCP Statistics

Agent Traffic Statistics

Agent Resource Metrics

ADP

AWS Console

DocuSign

Gmail

Jira

Office365 Outlook

SMTP

Salesforce

Skype

appsec-86f8e73c-1a31-4a3f-b667-c1a1ef1a0bod / Gmail

ClientProfileNetworkSegmentAllClientApplicationAll

Client Application Statistics

Action	Started	Succeeded	Failed	Skipped
View Message with Attachment	384	384	0	0
View Message	384	384	0	0
View Inbox	384	384	0	0
Sign Out	384	384	0	0
Sign In	384	384	0	0
Send Message with Attachment	384	384	0	0

Server Application Statistics

Action	Succeeded	Failed
View Message with Attachment	384	0
View Message	384	0
View Inbox	384	0
Sign Out	384	0
Sign In	384	0
Send Message with Attachment	384	0

Application stats drill down

## Conclusion

In this test, the true performance of the Device Under Test (DUT) is characterized by using a more realistic application traffic mix, which typically isn't provided in the vendors' datasheets.

This is a very important test to run because it uncovers the true capabilities of various DUTs, and helps customers make informed, data-driven decisions.

# Lab 5: Security Attacks with HTTP traffic

## Description

In this lab, we will observe the effect of transmitting data and simulating attacks resembling attacks in an enterprise network. These attacks are directed from client to server and from server to client. The severity of these attacks varies from critical, high, medium, and low.

For this lab, the emulated attacks list includes encrypted attacks against applications that run by default over SSL, critical strikes which have a CVSS v3.0 score between 9 and 10, and DoS attacks.

## Configuration

Load the config “Lab 5: Security Attacks with HTTP traffic”

Observe:

- The ‘Application Profile’ tab has been configured with an HTTP GET command.
- The ‘Attacks Profile’ tab has been configured with ‘All Encrypted Attacks’ from client to server and ‘Firefox Browser Attacks’ from server to client.
- The ‘Objectives & Timeline’ for the application profile tab has been configured to try and achieve 1 Gbps throughput over duration of 300 seconds.
- The ‘Objectives & Timeline’ for the attacks profile tab has been configured to achieve 1 attack per second, 1 mac concurrent attack, and 1 iteration.

Click on ‘Start’ and run the test.

## Result KPIs

While running the test and after the test has completed, please observe the following important KPIs:

- L23 TX/RX Throughput

We compare the TX vs RX throughput and observe if there are any drops. We also compare the throughput performance of 3000-byte frame traffic vs the previous 1500-byte frames traffic.



- Application successes/failures

This metric shows the number of applications initiated/succeeded/failed.

- Client to server (C2S) and server to client (S2C) attacks initiated/allowed/blocked

This metric shows us the number of attacks that were initiated vs allowed to pass through the network vs blocked, thus displaying the vulnerability of the end-to-end network. If the system has a firewall as a DUT, this test can be used to test the effectiveness of the firewall as well.

- C2S and S2C attack statistics

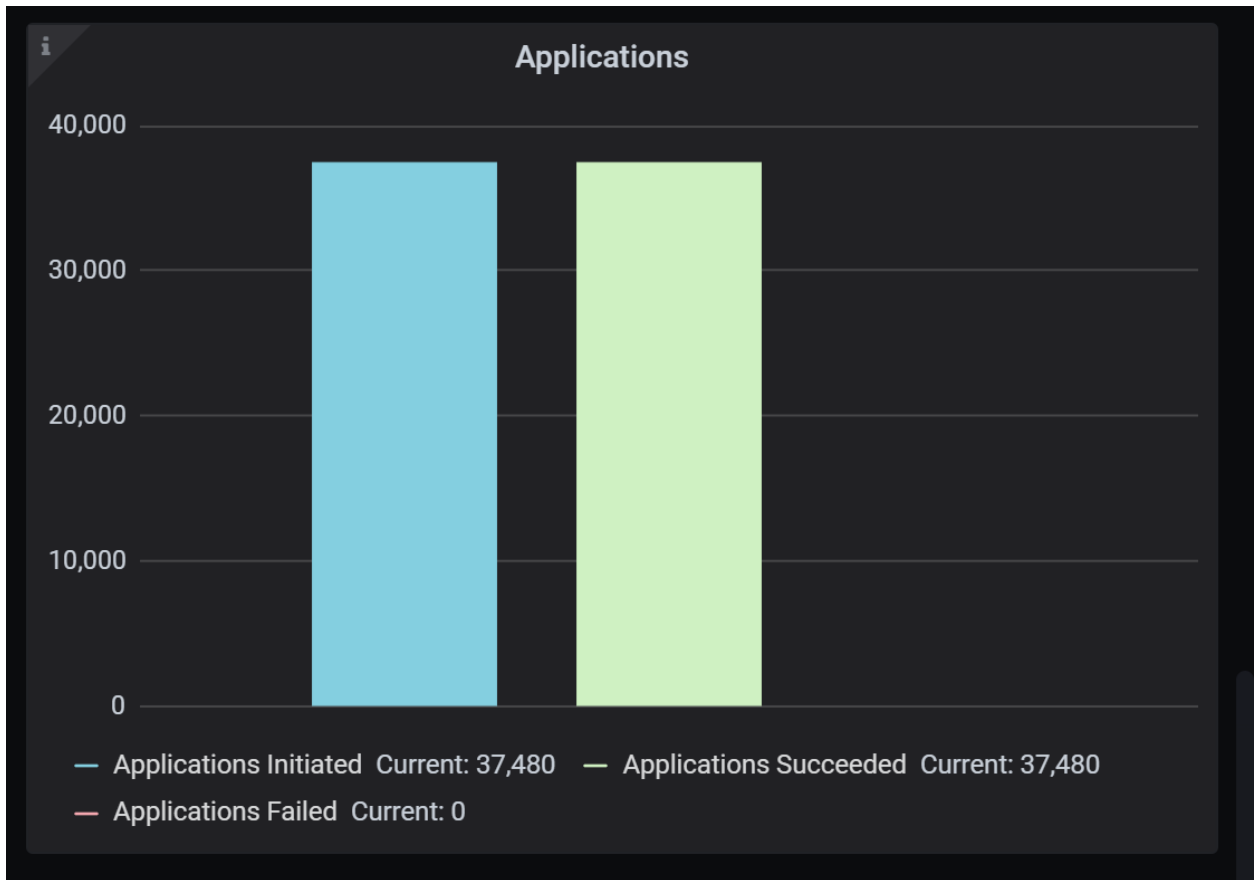
These detailed result pages display additional info on every strike in every attack we have added in the attack profile in the test. This detailed drill-down helps you take a step further in analyzing what strikes were initiated/allowed/blocked, thus helping with troubleshooting the network for vulnerabilities in a detailed manner.

- Instantaneous latency for application traffic and attacks

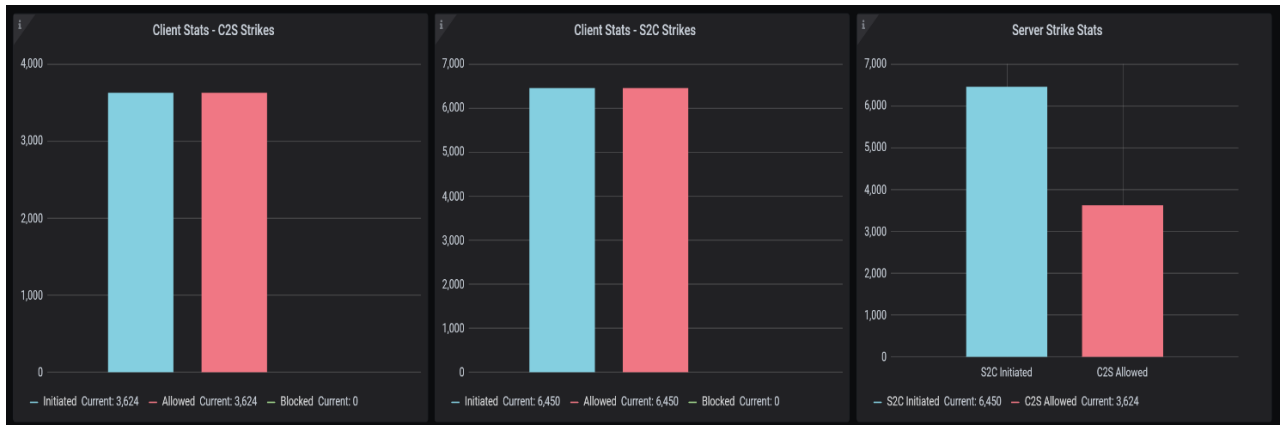
This graph has several sub graphs showing the average connection latency and min/avg/max of the following parameters: time to first byte, time to last byte. This info is displayed via separate graphs for application traffic and attacks.



Throughput TX/RX



Application successes/failures



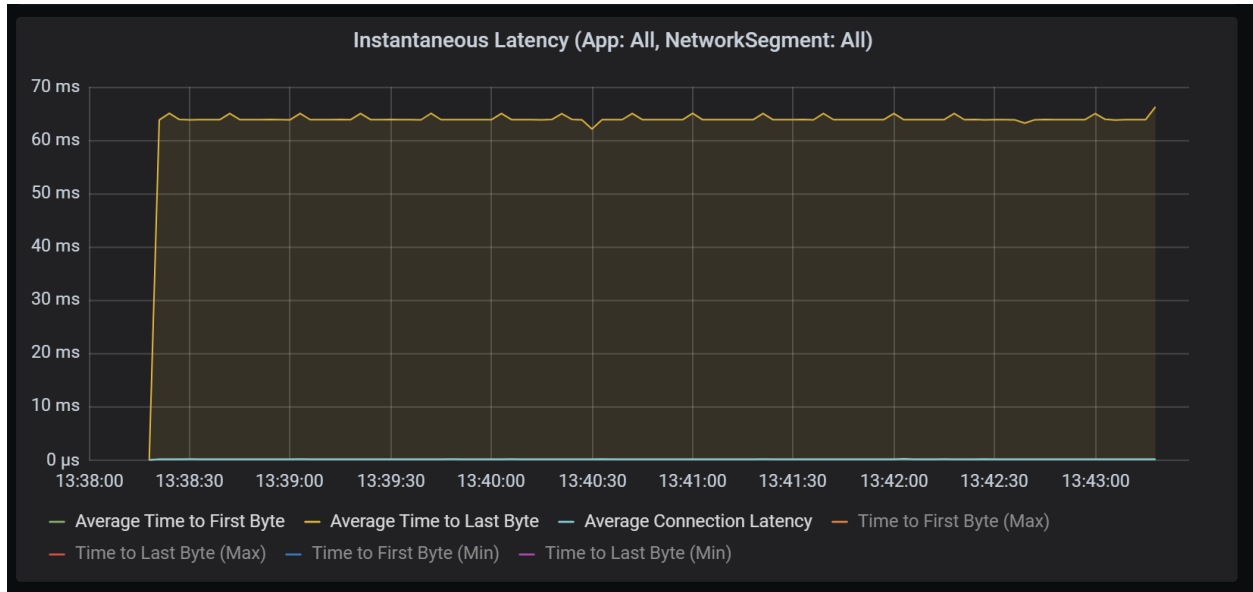
Attacks initiated/allowed/blocked

Filter views by name	ClientProfileNetworkSegment	All	ClientApplication	All					
Summary	C2S Attack Statistics								
Client Application Profile	Action	Reference	Severity	Protocol	Client Started	Client Allowed	Client Blocked	Client Skipped	Server Allow
Client Attack Profile	Strike WS02 Identity Server Stored Cross-Site Scripting	CVE: 2018-8716	Medium	http	151	151	0	0	151
Client Aggregated HTTP Statistics	Strike Trend Micro IWSVA Domain List bdn Parameter Command Inj...	URL: https://succes...	High	http	151	151	0	0	151
Client Statistics per Action	Strike Trend Micro Email Encryption Gateway searchEmail SQL Inje...	CVE: 2018-6230	Medium	http	151	151	0	0	151
Server Statistics	Strike SaltStack Salt API SSH Client Command Injection	CVE: 2020-16846	High	http	151	151	0	0	151
Server HTTP Statistics	Strike Ruckus IoT Controller Web UI OS Command Injection	CVE: 2020-26878	High	http	151	151	0	0	151
TCP Statistics	Strike Ruckus IoT Controller Web UI Authentication Bypass	CVE: 2020-26879	High	http	151	151	0	0	151
Agent Traffic Statistics	Strike Novell Zenworks Configuration Management scheduleQuery ...	CVE: 2015-0782	Medium	http	151	151	0	0	151
Agent Resource Metrics	Strike Novell Zenworks Configuration Management remote code ex...	CVE: 2015-0779	Critical	http	151	151	0	0	151
All Encrypted Attacks	Strike Multiple FS BIG-IP products Directory Traversal	CVE: 2020-5902	Critical	http	151	151	0	0	151
Firefox Browser Attacks	Strike Kubernetes Dashboard Authentication Bypass Information Di...	CVE: 2018-18264	Medium	http	151	151	0	0	151
HTTP	Strike HPE iLO 4.1.00-2.50 Administrator Account Creation	CVE: 2017-12542	Critical	http	151	151	0	0	151
	Strike HPE System Management Homepage gsearch.php.en Cross-	CVE: 2017-12544	Low	http	151	151	0	0	151
	Strike HPE Network Automation RedirectServlet SQL Injection	CVE: 2017-5810	High	http	151	151	0	0	151
	Strike HPE Network Automation PermissionFilter Authentication By...	CVE: 2017-5812	Medium	http	151	151	0	0	151
	Strike HPE Intelligent Management Center accessMgrServlet Insecur...	CVE: 2017-5790	Critical	http	151	151	0	0	151
	Strike Fortinet FortiOS SSL VPN Credentials Disclosure	CVE: 2018-13379	Medium	http	151	151	0	0	151
	Strike Dell EMC VMAX Virtual Appliance Manager Authentication B...	CVE: 2018-1216	Critical	http	151	151	0	0	151
	Strike Dell EMC Storage Manager Server Directory Traversal	CVE: 2017-14384	High	http	151	151	0	0	151
	Strike D-Link DNS-320 ShareCenter Unauthenticated Remote Code ...	CVE: 2019-16057	Critical	http	151	151	0	0	151

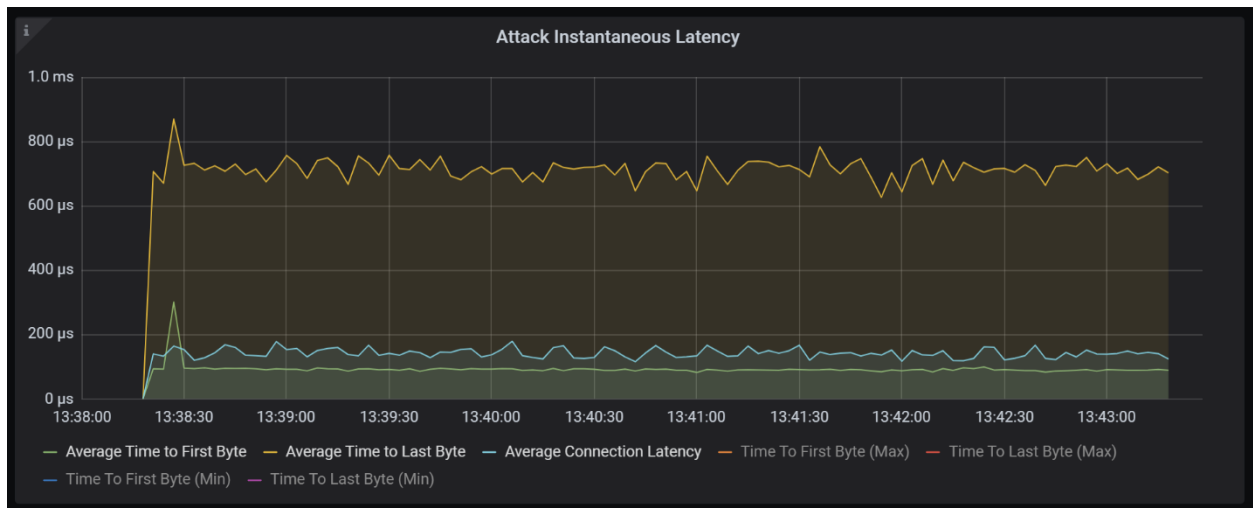
## C2S attack statistics

Summary	S2C Attack Statistics						
Client Application Profile	Action	Reference	Severity	Protocol	Server Initiated	Client Allowed	Client Blocked
Client Attack Profile	Strike Windows SMB Redirect	URL: http://blog.cylance.co...	High	http	150	150	0
Client Aggregated HTTP Statistics	Strike Mozilla SpiderMonkey IonMonkey Type Confusion	CVE: 2019-9813	High	http	150	150	0
Client Statistics per Action	Strike Mozilla Multiple Products Multiple Header Handling	CVE: 2011-3000	Medium	http	150	150	0
Server Statistics	Strike Mozilla Firefox Thunderbird SeaMonkey Javascript IDBKeyRa...	CVE: 2012-0469	Critical	http	150	150	0
Server HTTP Statistics	Strike Mozilla Firefox plugin Array Pointer Heap Corruption	CVE: 2010-2755	Critical	http	150	150	0
TCP Statistics	Strike Mozilla Firefox nsPropertyTable PropertyList Memory Corrupt...	CVE: 2009-3070	Critical	http	150	150	0
Agent Traffic Statistics	Strike Mozilla Firefox createImageBitmap Integer Overflow	CVE: 2017-5428	Medium	http	150	150	0
Agent Resource Metrics	Strike Mozilla Firefox clipPath SVG stroke-width Memory Corruption	CVE: 2007-0776	High	http	150	150	0
All Encrypted Attacks	Strike Mozilla Firefox WebGL Intersect Integer Overflow	CVE: 2017-5459	Critical	http	150	150	0
Firefox Browser Attacks	Strike Mozilla Firefox WebAssembly Table Object Integer Underflow	CVE: 2018-5093	High	http	150	150	0
HTTP	Strike Mozilla Firefox TypeObject Use After Free	CVE: 2014-1512	High	http	150	150	0
	Strike Mozilla Firefox Thunderbird and Seamonkey Table Memory C...	CVE: 2012-1952	High	http	150	150	0
	Strike Mozilla Firefox Spidermonkey IonMonkey Spidermonkey Arr...	CVE: 2019-11707	High	http	150	150	0
	Strike Mozilla Firefox Spidermonkey IonMonkey ObjectGroup Type ...	CVE: 2019-9816	Medium	http	150	150	0
	Strike Mozilla Firefox SVGZoom Memory Corruption	CVE: 2007-2867	High	http	150	150	0
	Strike Mozilla Firefox SVG pathSegList getItem Negative Argument ...	CVE: 2007-2867	High	http	150	150	0
	Strike Mozilla Firefox SVG Animation NotifyTimeChange Use After ...	CVE: 2016-9079	High	http	150	150	0
	Strike Mozilla Firefox ReadableStreamCloseInternal Out of Bounds ...	CVE: 2020-6806	High	http	150	150	0
	Strike Mozilla Firefox QueryInterface() Arbitrary Code Execution (O...	CVE: 2006-0295	High	http	150	150	0

## S2C attack statistics



## Instantaneous latency for application traffic



## Instantaneous latency for attacks

## Conclusion

In this test, we observe the true performance of the Device Under Test (DUT) is characterized by using a more realistic attacks and traffic mix, which typically isn't provided in the vendors' datasheets.

If you introduce a DUT in your setup, you can observe if these attacks are allowed or blocked by middle devices, firewalls, etc.

This is a very important test to run because it uncovers the true capabilities of various DUTs, and helps customers make informed, data-driven decisions.