

CyPerf Test Drive: Validating Zero Trust Network Access

Testing That Replicates Your Network in Action

Table of Contents

| | |
|---|----|
| Overview - The need for Performance and Security testing for Zero Trust | 3 |
| The Missing Link for Testing Zero Trust Network Access | 3 |
| Lab Introduction | 5 |
| Lab 1: Validating ZTNA Access Policies under traffic load | 8 |
| Lab 2: Security effectiveness | 25 |

Overview - The need for Performance and Security testing for Zero Trust

The zero trust model which assumes trusting nothing and always verifying is driving the need for a complete paradigm shift in IT security where users, applications, and devices must undergo stringent validations before gaining access to an application or requested resource. Every network and application request goes through identity-based authentication. The zero-trust model also enforces a need-to-know approach across networks, devices, users, workloads, and data.

As traditional networks quickly evolve to the distributed cloud and hybrid environments, they are using edge computing and centralized resources where applications reside closer to users and the source of data on local equipment. Legacy networks with a clearly defined network perimeter are easier to defend and validate than distributed clouds, where there is no defined network perimeter, making application and security testing challenging. Traditional network security assumes that once users are inside a network, they are trustworthy. This implicit trust enables lateral movement once inside the network, not only by legitimate users but also by hackers and other malicious traffic that intends to harm.

The Missing Link for Testing Zero Trust Network Access

Achieving full zero trust is a journey rather than a one-time project. Below figure is an example of a methodical approach. First, run a thorough assessment of where your organization stands relative to zero trust and start with the areas which involve securing the most critical and valuable assets of the business. Many technology companies provide tools to assess an organization's zero trust maturity level. A good starting point of the journey is to use the available tools to help identify any gaps your organization has in adopting zero trust.

It is crucial to your success on the road to zero trust to get agreement from all stakeholders. Aside from a very well-built project plan, practitioners should also provide a clear return on investment (ROI) and quantifiable improvements. Every change introduces unknowns and unfortunately the reluctance to the unknown is in many instances, the thing that hinders progress and innovation.

"If you can't measure it, you can't improve it." – this familiar quote by Peter Drucker is paramount when it comes to adopting new technologies and for zero trust it is even more important since having the ability to prove and measure improvements will ultimately be the decisive factor.



[Source: Illumio, <https://www.illumio.com/blog/operationalizing-zero-trust-step-1>]

Figure 1. An incremental six-step approach to zero trust

One of the missing links to accelerating and adopting zero trust is that organizations scarcely perform testing and validation. Incomplete or lack of testing leaves doubts, uncertainty, and even worse — blind spots when it comes to changing an organization's network security posture.

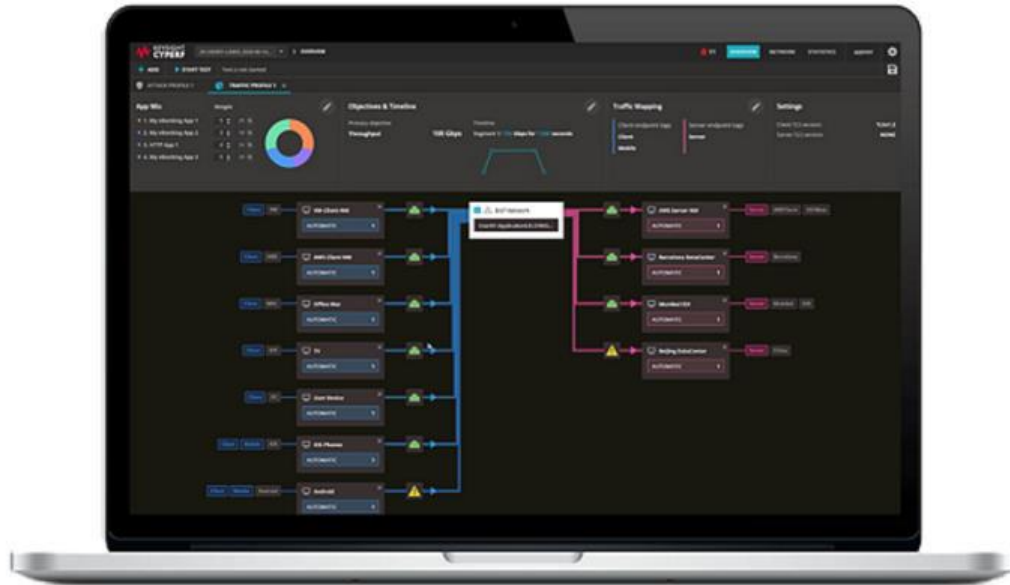
Testing the infrastructure in a distributed zero trust environment is critical for the enterprises and network equipment manufacturers (NEMs) who provide the infrastructure.

Digital business transformation and edge computing are bringing major unknowns to the performance, scalability, and threat protection of new, emerging network, and security architectures.

An enterprise faces the following challenges as it transitions to zero trust network access architectures:

- Accessing quantifiable data and key performance indicators like throughput, concurrent users, latencies, and quality of experience before and after implementing zero trust controls
- Demonstrating to customers and stakeholders the immediate value of zero trust implementations through live-traffic, and custom proof of concepts (POCs)
- Characterizing the security efficacy of zero trust strategies especially with lateral network movement and/or inside sourced attacks
- Assessing the functionality, performance, and scale of the zero trust authentication engines on a continuous basis to monitor for any undesirable deviations

Your perimeter-less, zero trust network architecture requires a new testing paradigm.



[Keysight CyPerf](#) is the industry's first instantly scalable network test solution for zero trust. It recreates realistic workloads across various physical and cloud environments to deliver deep insights into end-user experience, security posture, and performance bottlenecks of distributed and 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.

Lab Introduction

Overview

During this lab, you will get access to a cloud-based setup with distributed traffic agents that generate realistic application and malicious traffic to validate a Zero Trust Network Access enforcement device from the following key aspects:

- validate the least privilege access policies
- assess the network performance and scale
- characterize the security efficacy

The main two components of the Lab environment are:

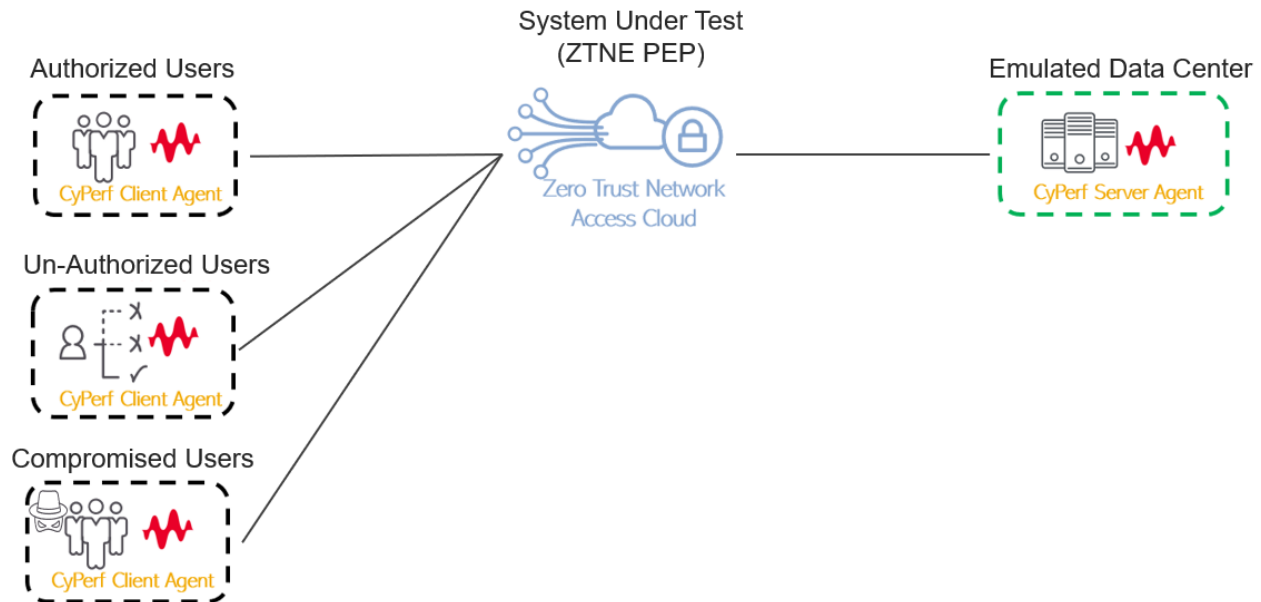
1. The **test tool**: Keysight's CyPerf emulating legitimate and malicious traffic clients as well as servers (all deployed as cloud instances)
2. The **device under test (DUT)**: Zero Trust Network Access (ZTNA) Policy Enforcement Point (PEP)

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

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
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 ZTNA PEP (i.e., DUT).

Setup

Below is the high-level diagram of the setup used in this lab:



The setup for all the labs consists of the following:

1. Traffic Agents:
 - **Clients:** Three CyPerf traffic agents acting as clients deployed in three different locations to model a distributed deployment. Each CyPerf traffic agent will emulate the role of a different user type.
 - **Servers:** One CyPerf traffic agent acting as an application server and protected by the ZTNA PEP (that is, DUT).

Important

CyPerf traffic agents (clients and servers) can be virtually deployed in any Region/Zone, across a variety of public clouds (e.g., 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, please refer to the [product datasheet](#).

2. System/Device Under Test (DUT):

Zero Trust Network Access (ZTNA) Policy Enforcement Point (PEP): The configuration and the DUT will mostly consist of least privilege access policies that will:

- Allow users that have been authenticated to access resources or applications hosted in the Data Center
- Restrict access to certain applications and resources to all remote users, even for remote authenticated users
- Inspect traffic to detect and block security attacks on a continuous basis for all type of users (authenticated or not)

Resources and Prerequisites

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

Resources for these labs can be found at the following location:

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

This includes:

- Configuration files: each lab will start from a configuration file that is pre-loaded but it is also available at above location for users that would like to use it in their own test environment.
- Step-by-step lab guiding document (this document)
- Intro video: a quick video that guides users on how to spin up and manage the test drive environment

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: Validating ZTNA Access Policies under traffic load

Description

In this first test we will validate the **ZTNA PEP access policies under traffic load** by using two CyPerf client agents emulating the following roles:

- **Authorized users:** these emulated users are valid remote users, with valid credentials that are generating web requests and are authorized to access a certain web application hosted on the emulated server agents.
 - These users will test that the DUT was configured appropriately and is functioning accordingly for a legitimate, authorized user group

- **Un-Authorized users:** these emulated users are also valid remote users, with valid credentials as well, however they are generating web requests for another web application that is restricted and not authorized to be accessed by these remote users
 - These users will test that the DUT was configured appropriately and is functioning accordingly for an un-authorized application and user group

Along with the above user groups, CyPerf test agents will also emulate the following two applications:

- **Authorized app:** this will be the authorized web application and will consist of web transactions between the corresponding client test agent and the server test agent
- **Restricted app:** this will be the authorized web application and will consist of web transactions between the corresponding client test agent and the server test agent

Most of the traffic in today's networks is encrypted, therefore the above two applications are also using TLS encryption which will put even more stress on the DUT as it needs to decrypt, inspect, and re-encrypt the traffic that is passed through to properly apply the least privilege policies it has been configured with.

Objectives

This first objective of this test is to validate that the ZTNA PEP access policies are configured and applied correctly for different user groups under traffic load.

In addition, CyPerf provides a very large set of Key Performance Indicators (KPIs) that will assess the DUT performance capabilities that might impact user's Quality of Experience (QoE), such as:

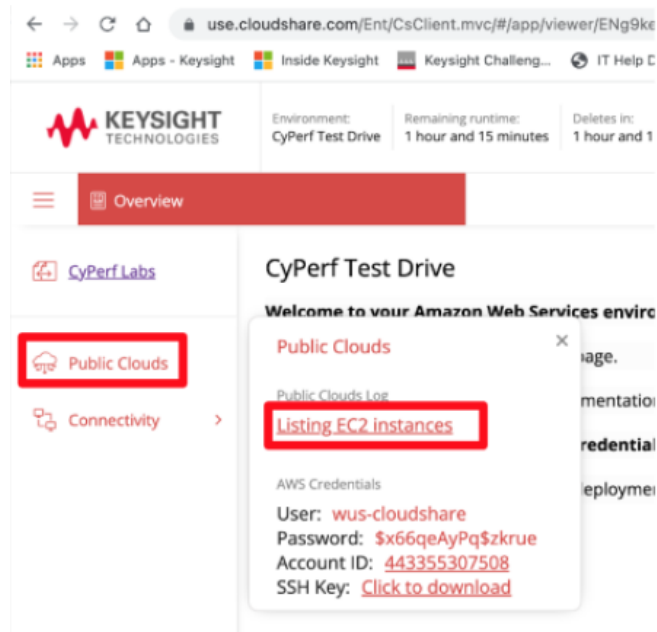
- Throughput
- Concurrent Simulated Users (or Concurrent Connections),
- Simulated Users per Second (or Connections per Second)
- Latencies (connection latency, first response latency, object transfer latency)
- Application iterations successful/failed
- Authentications successful/failed
- TCP statistics (connections initiated vs failures, TCP retries/aborts etc)

As a time saving approach, import a configuration file that was created before. Nevertheless, the most important parameters are examined as per the below instructions.

Step-by-Step Instructions

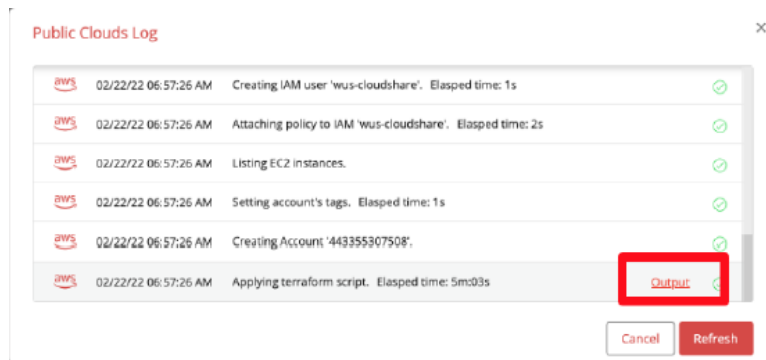
1. Connect to the CyPerf Controller using the IP address or DNS name:

To perform the above step, in the CloudShare web page, select, **Public Clouds > Listing EC2 instance:**

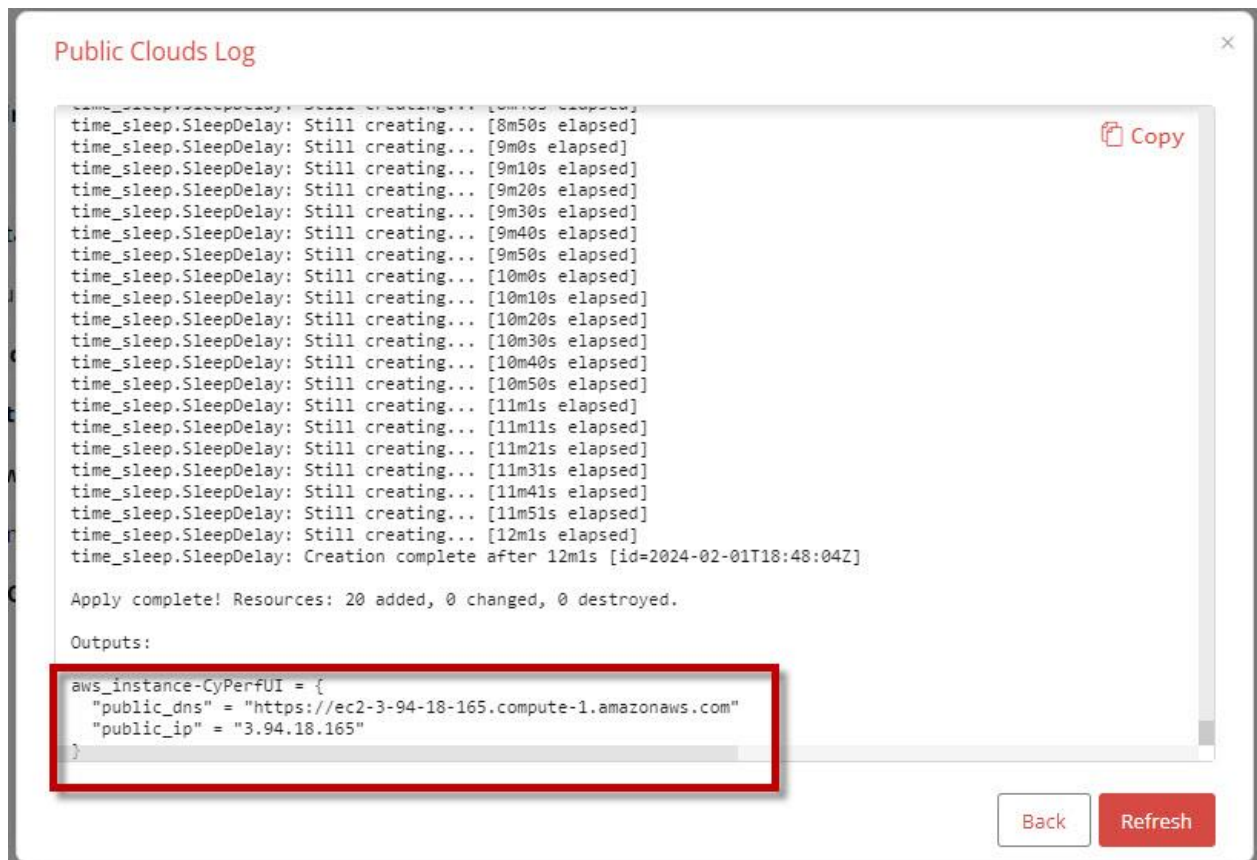


Note: If the **Listing EC2 instance** link is not available, ensure that you started the cloud deployment as per the introduction video (stream it from [here](#)).

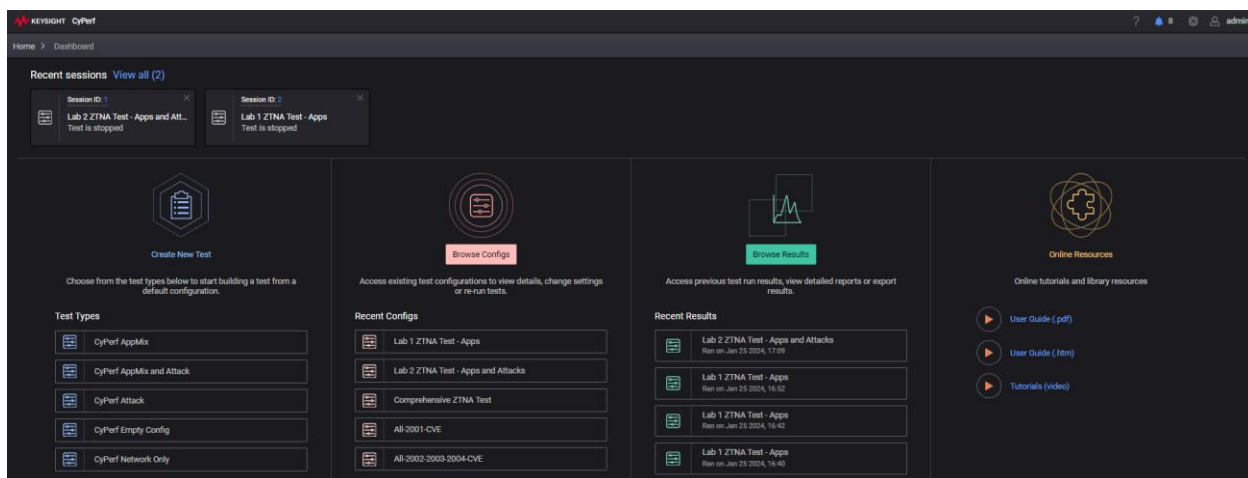
2. Select the **Output** link next to the *Applying terraform script* line as depicted in the following image:



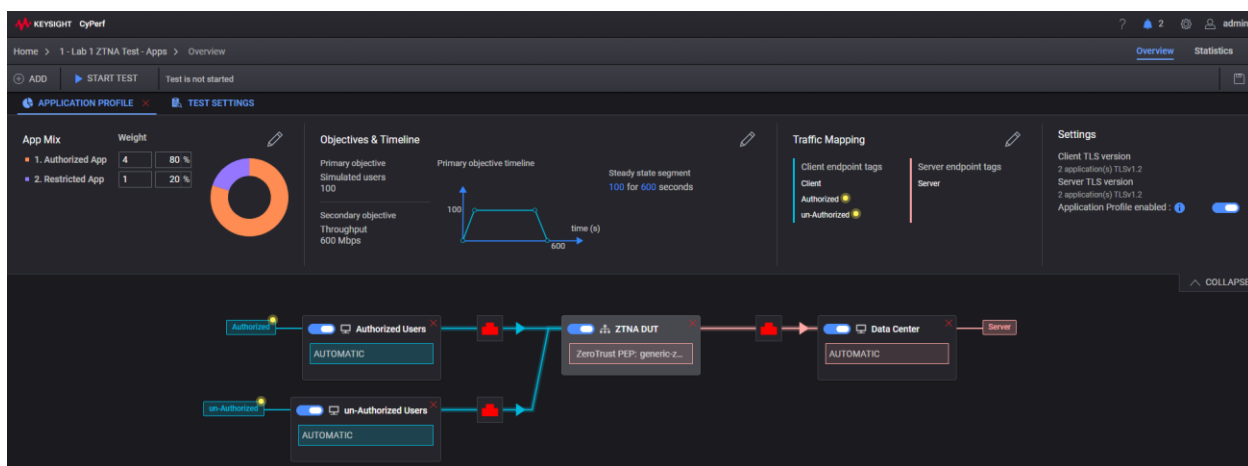
3. Search the IP address of the CyPerf Controller (that is, *aws_instance-CyPerfUI*) in the Public Clouds Log (scroll till the end) as depicted in the following image. This is the IP address that will be used to connected to the webUI of the CyPerf controller.



4. Connect to the CyPerf Controller: open a Web Browser and type the IP address of the CyPerf Controller (from above step).
5. Accept the TLS certificate of the Controller Web UI, because this is a self-signed certificate.
6. Use the following credentials to sign in:
 - UserName: admin
 - Password: CyPerf&Keysight#1
7. On the controller UI landing page (after login), in the **Recent Configs** area of the **Browse Configs** section click on the config "**Lab 1: ZTNA Test - Apps**".



The new config file loads into a new CyPerf Session:

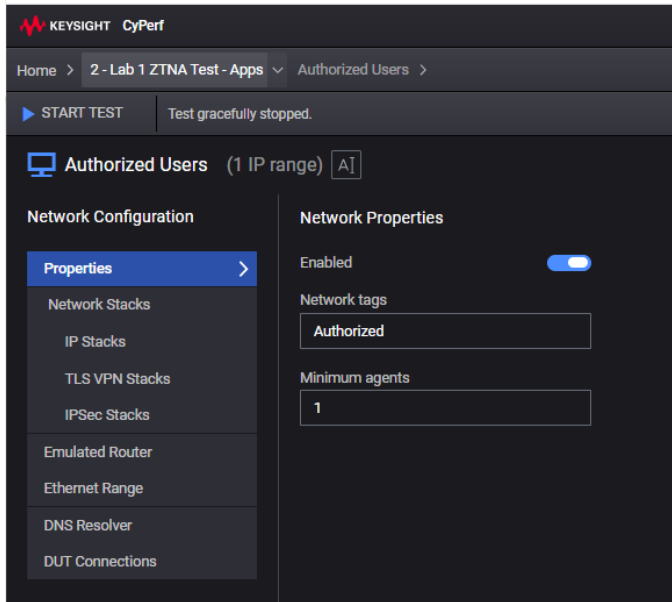


This is the CyPerf test overview page, where all the important test parameters are available in a single pane of glass.


The bottom half of the webpage is an interactive representation of the logical topology used in this test. This topology reflects the setup described in the Lab Introduction section.

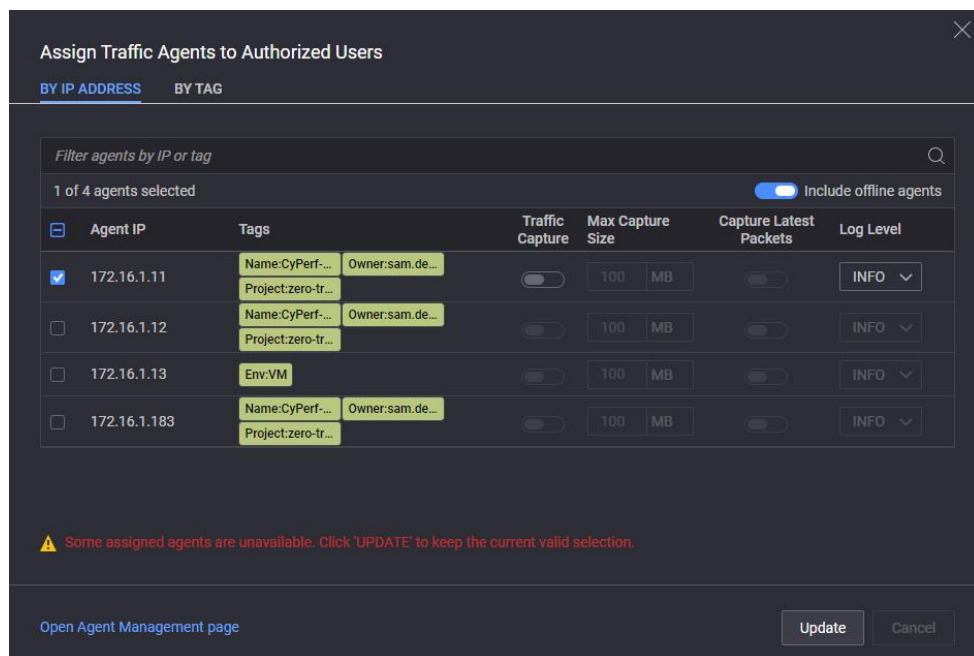
For example, if you select the **Authorized Users** network segment, all the network related parameters like configured IP addresses (in this case it is set on Automatic, which means that the already assigned IP addresses are used), DNS Resolvers, and so on are visible.


You can select the back button in the browser or select the **Lab 1 ZTNA Test - Apps** button in the breadcrumb to navigate back to the test overview page:



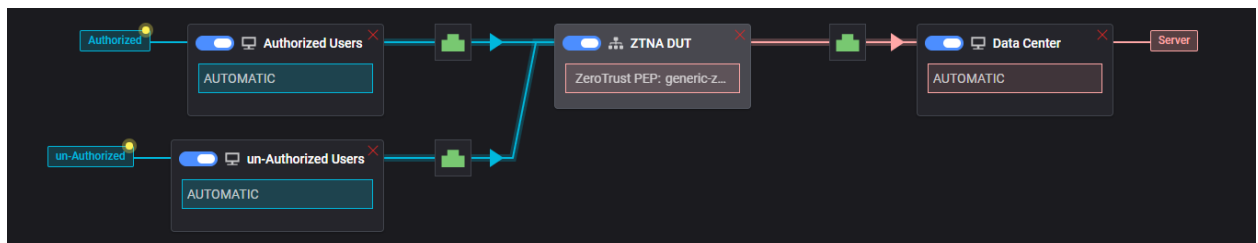
8. For each of the emulated Network Segments (i.e., **Authorized Users**, **un-Authorized Users** and **Data Center**) we need to assigned a CyPerf test agent that was pre-deployed as a VM instance in the cloud:

- a. Click the red port () associated to the **Authorized Users** Network Segment:
 - i. In the new Dialog select the **172.16.1.11** IP address (which corresponds to one pre-deployed CyPerf test agent) and select update

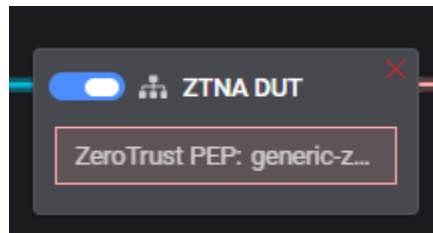


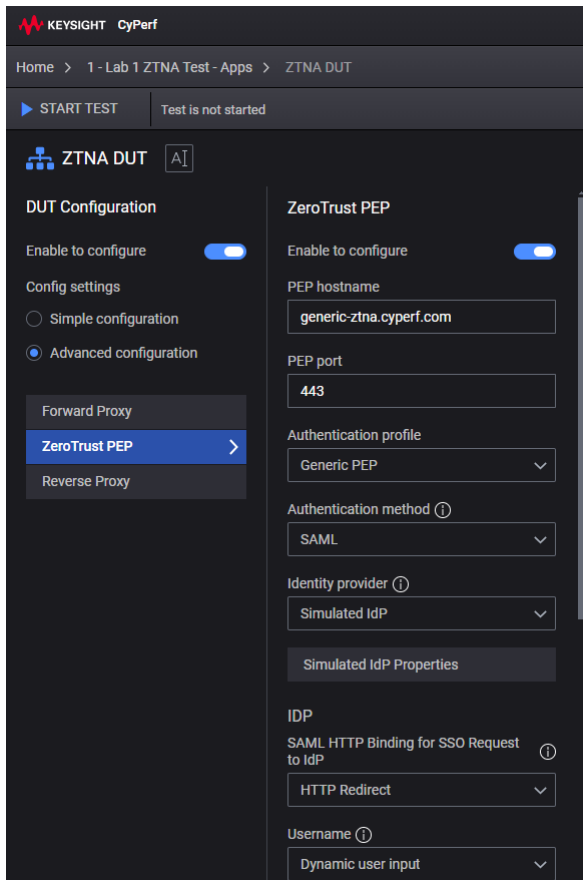
- ii. The green port () associated to the **Authorized Users** Network Segments indicates that an CyPerf agent was successfully assigned and is ready to generate traffic based on the test config
- b. Repeat above steps for the **un-Authorized Users** Network Segment and assign the CyPerf agent with IP address: 172.16.1.12
- c. Finally, repeat above steps for the **Data Center** Network Segment and assign the CyPerf agent with IP address: 172.16.1.183

Now all Network Segments need to have a green port associated. If there is a single red port, please re-visits steps a-c above and make sure all Network Segments have a green port before proceeding.



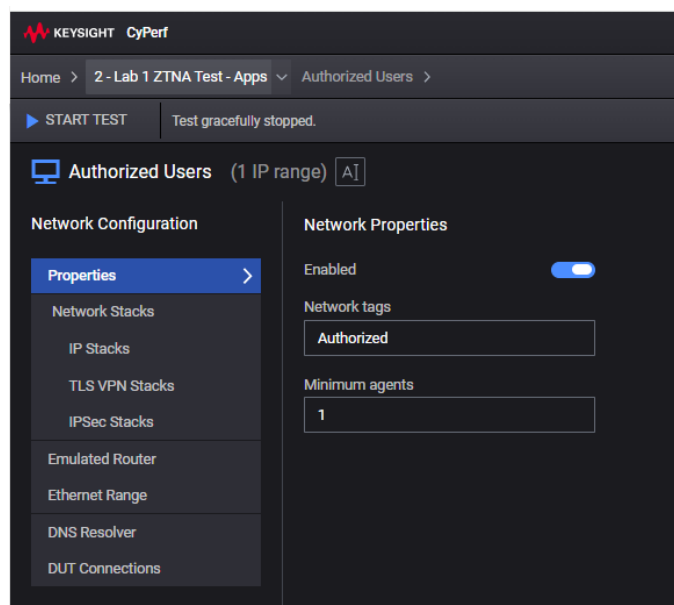
9. Next, to see the configuration parameters that CyPerf emulated users use to connect and generate traffic through the ZTNA PEP, Select the **DUT Network** object in the middle of the topology diagram:



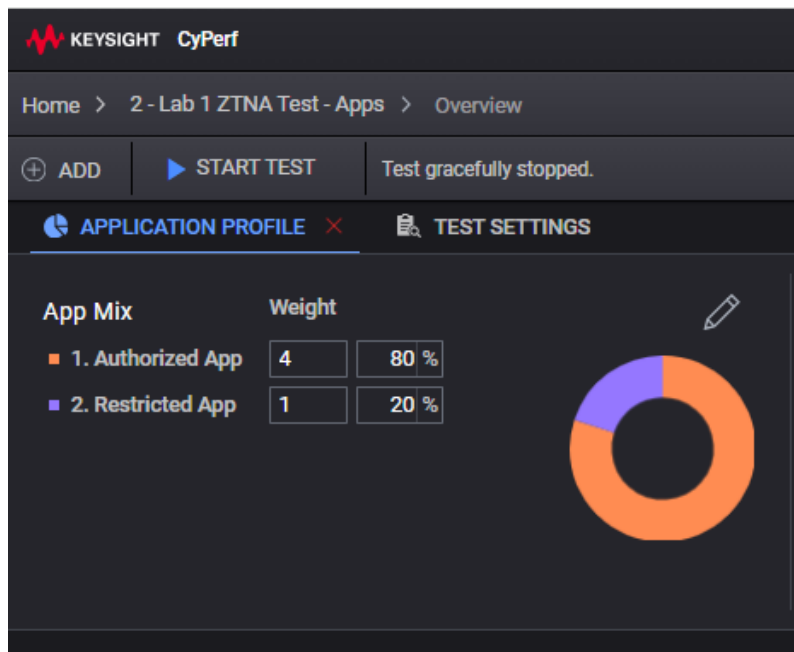


Under the ZeroTrust PEP the most important parameters are being configured so that the CyPerf emulated clients would statefully interact with the ZTNA PEP for the authentication and authorization phase before sending applications requests.

10. Select the **Lab 1 ZTNA Test - Apps** button in the breadcrumb at the top of the page to return to the test overview page:

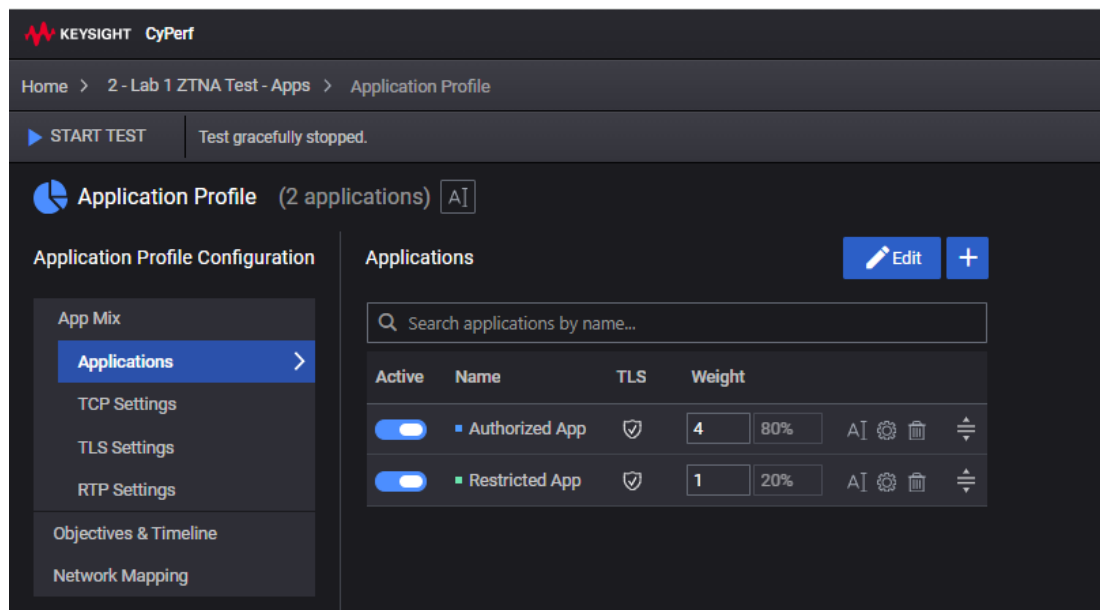


11. **Application Profile:** Select the edit icon () in the Application Profile section:



The new menu provides access to all the L4-7 configurations options, like:

- **Type of traffic** that is being sent from an L7 perspective:



In this test, there will be two different applications that will be emulated:

- **Authorized App:** this application has been configured in the DUT (ZTNA PEP), through its URL to be remotely accessible by authenticated users. The CyPerf server agent is hosting this application and the CyPerf client agent is trying to access it via the DUT (ZTNA PEP).

- **Restricted App:** this application has been configured in the DUT (ZTNA PEP), through its URL to NOT be remotely accessible even by authenticated users. The CyPerf server agent is hosting this application and the CyPerf client agent is trying to access it via the DUT (ZTNA PEP).

Important

CyPerf offers an unmatched flexibility for configuring and parametrizing the emulated applications (and its actions), so that you can create your own unique application traffic mixes matching your production environment as close as possible.

- **Objectives and Timeline:** this is where the test load objective and duration are configured:

KEYSIGHT CyPerf

Home > 2 - Lab 1 ZTNA Test - Apps > Application Profile

▶ START TEST Test gracefully stopped.

Application Profile (2 applications) AI

Application Profile Configuration

- App Mix
- Applications
- TCP Settings
- TLS Settings
- RTP Settings
- Objectives & Timeline**
- Network Mapping

Objectives & Timeline

- Primary Objective**
- Secondary Objective
- Advanced Settings

Primary Objective

Objective type
Simulated users

Steady state segment

| Value | Duration |
|-------|----------|
| 100 | 600 |

Timeline

Max simulated users per second ①
10

Max pending simulated users (use a fixed value or a percentage of the total users)
1%

Timeline representation

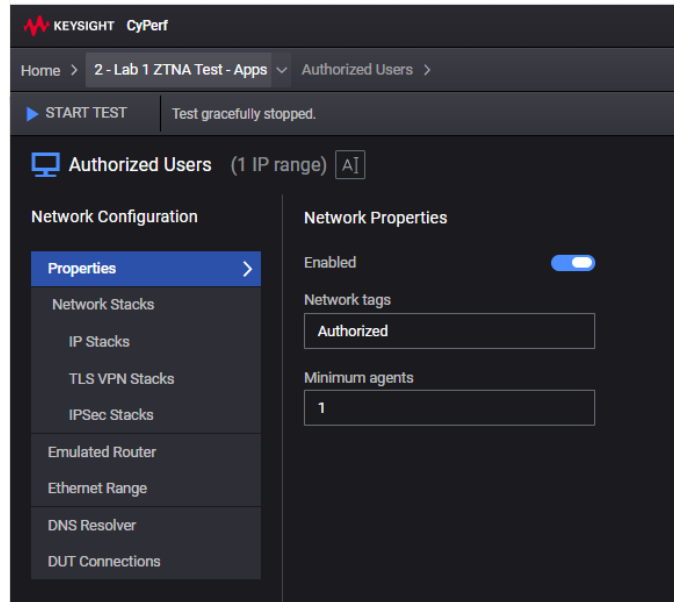
The graph shows a blue line representing the number of simulated users over time. The y-axis is labeled '100' and the x-axis is labeled 'time (s)' with a mark at '600'. The line starts at the origin (0,0), rises to a value of 100, remains constant at 100 for a duration of 600 seconds, and then falls back to 0.

In this test we will use as a primary objective 100 Simulated Users and will run the test for 600 seconds.:

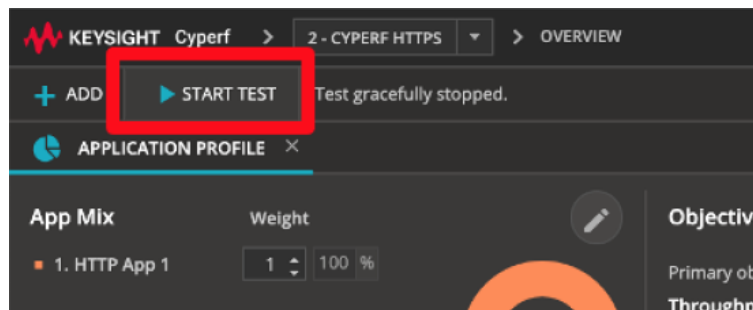
- 80% of these users will emulate **authorized** application requests.
- 20% of these users will emulate **un-authorized** application requests.

In addition, we have also configured an “**Secondary Objective**” where we are “instructing” the 100 Simulated Users to generate 600Mbps of throughput.

12. Select the **Lab 1 ZTNA Test - Apps** button in the breadcrumb at the top of the page to return to the test overview page:



13. Select Start Test:



The test traffic agents are configured and in a few moments the traffic starts. The view automatically switches to the STATISTICS dashboard.

Important

The reason why the same 600Mbps is kept as the traffic load is due to cost reasons (as the traffic is distributed and crossing through a number of network and security controls).

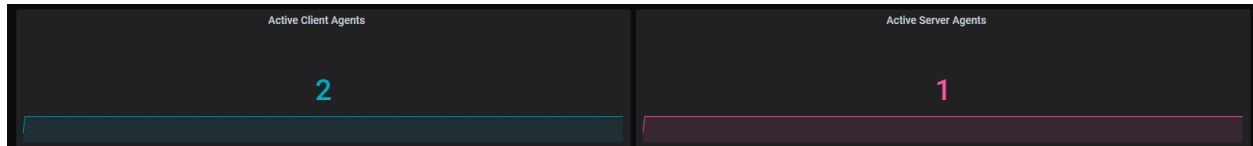
Result Analysis

Real Time Statistics

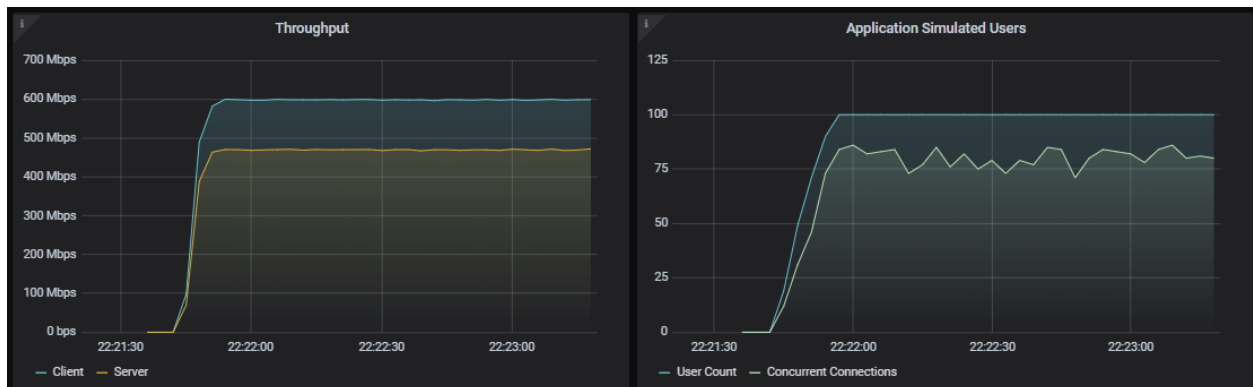
The graphs below provide a view of the real-time statistics for the test.

Dedicated **Zero Trust** statistics will be provided, and we will interpret those below.

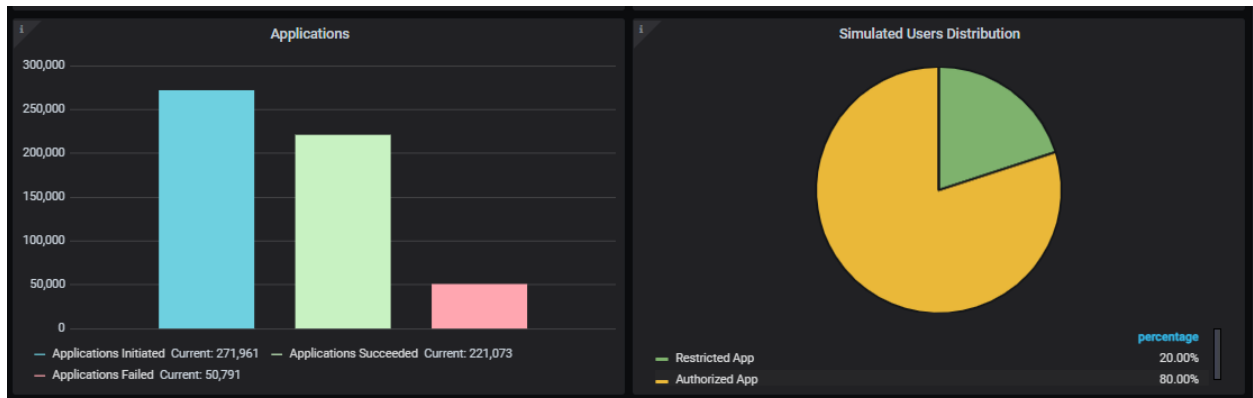
As a first thing, observe that there are two clients and one server that are active and participate in the test, which is as per the configuration and expectations:



Next, in terms of **Simulated Users** and **Throughput**, the statistics indicate that the setup is able to easily reach the 100 Simulated Users and 600 Mbps that has been configured:

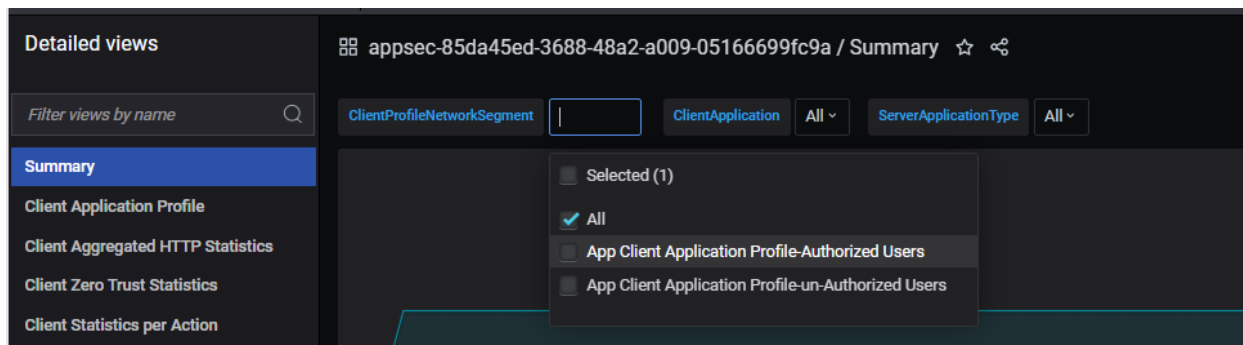


Scrolling down in the **Summary** page we can see that the user distribution is per our configuration as well (i.e. 80 Users running the Authorized app and 20 Users running the Restricted App).

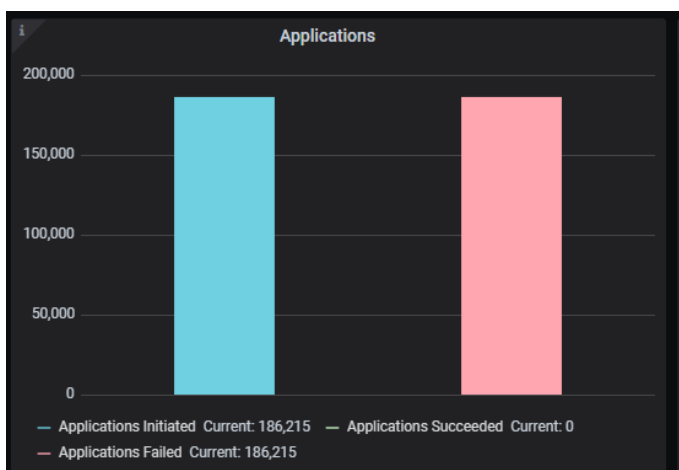


Also, one more important point is that in the **Application** bar chart above we see that there are some applications failing.

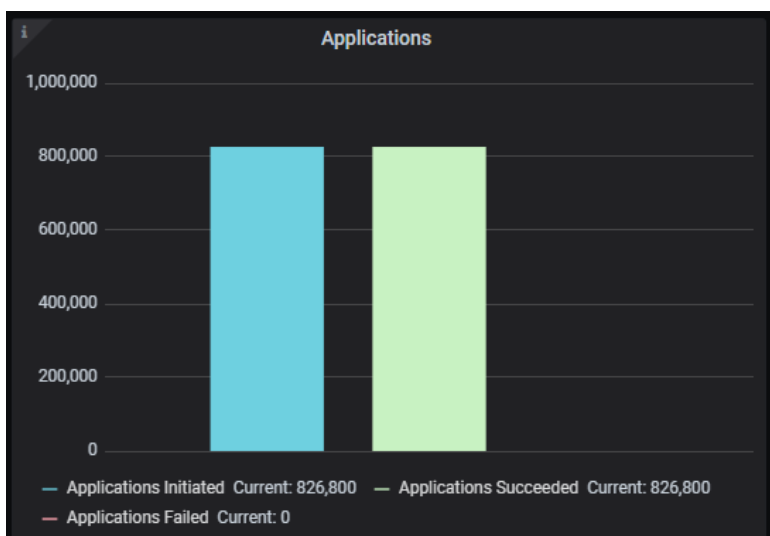
To further debug this, from the upper right corner of the UI we can apply filters for each group of users as per below screenshot:



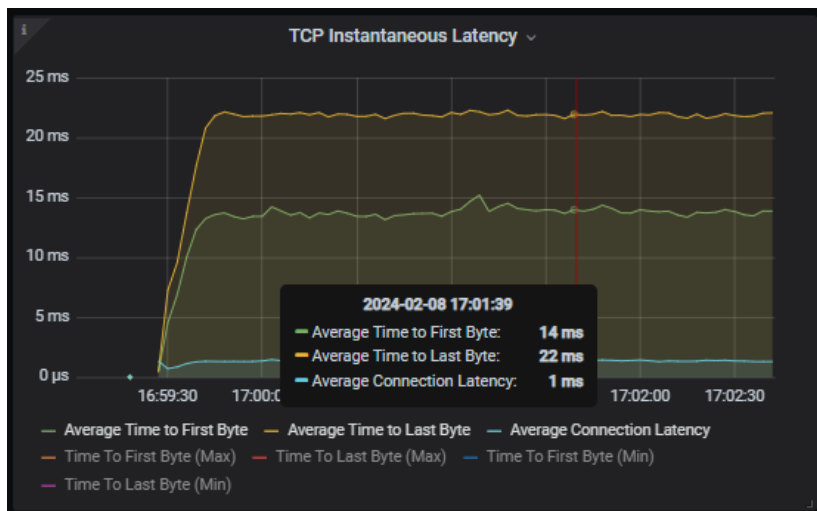
Therefore, if we select only “**un-Authorized Users**” for the **ClientProfileNetworkSegment** filter, we will notice that all the applications are failing, which is per our expectations:



Similarly, if we apply filters to only see “**Authorized Users**” (using the same menu), we will notice that all the applications are successful, which is again per our expectations:



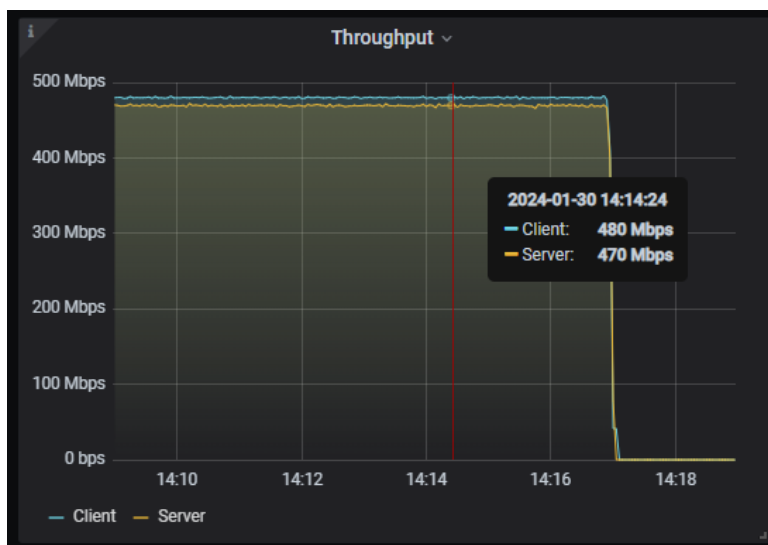
Keeping the filter for the “**Authorized Users**”, we can look at other important metrics to understand the performance of the system for this user group:



In the **Instantaneous Latency** view, you can distinguish the following three important latency related metrics:

- Connection Latency: the average time it took for the TCP connection to establish (in our example around 1ms)
- Time to First Byte: the average time to receive the first data byte since the request was issued (in our example around 14ms)
- Time to Last Byte: the average time to receive the full data object/page since the request was issued (in our example around 22ms)

Throughput is also very consistent for the “**Authorized Users**” as well (around 480Mbps):



Another important view is the “**Client Zero Trust Statistic**” (from the left-hand navigation pane):

Detailed views

appsec-85da45ed-3688-48a2-a009-05166699fc9a / Client Zero Trust Str

🌐🔍🔒🔑🔧🔗🔗

In this view we can see two important aspects:

- First, the **Authorized** users have all successful authentications and 0 failures. Moreover, we can also see that there is a fairly large number of “*Authentication Reused*” which is consistent with our configuration where each stateful authentication triggers an 10x re-use of the same token.
- Another important point is that the **Restricted** application running from the **un-Authorized** users has 0 authentications started but if we look in the second table, we see that all the application requested had a 401 response received: and as we know, this is the code for “Unauthorized”.
 - The behavior of this particular ZTNA PEP is that for each restricted or un-Authorized application request, the DUT will immediately send an 401 Unauthorized HTTP response code, without even asking the users to authenticate.

Next, for the current test, you can further look into more granular details.

The **TCP Statistics** dashboard will reveal any significant packet drops (mostly signaled through Data Retransmissions) as well as other error statistics like Connection Failures or Retransmissions, Aborts:

| | | | | | | | | |
|-----------------------------------|---|-------------------|-----------------------------|--------------------|-----------------------------|------------------------------|--------------------------------|--------------|
| Summary | TCP Client Statistics (NetworkSegment: All) | | | | | | | |
| Client Application Profile | Application Profile - Network Segment | SYN Sent | SYN Received | SYN Failed | Connections Established | Connection Initiation Failed | FIN Sent | FIN Received |
| Client Aggregated HTTP Statistics | App Client Application Profile-un-Authorized Users | 749,120 | 0 | 0 | 749,120 | 0 | 304,330 | 583,207 |
| Client Zero Trust Statistics | App Client Application Profile-Authorized Users | 831,638 | 0 | 0 | 831,638 | 0 | 0 | 831,638 |
| Client Statistics per Action | | | | | | | | |
| Server Statistics | | | | | | | | |
| Server HTTP Statistics | | | | | | | | |
| TCP Statistics | | | | | | | | |
| Agent Traffic Statistics | | | | | | | | |
| Agent Resource Metrics | TCP Client Retransmissions Statistics (NetworkSegment: All) | | | | | | | |
| HTTP | Application Profile - Network Segment | SYN Retransmitted | SYN Retransmissions Aborted | Data Retransmitted | Data Retransmission Aborted | SYN-ACK Retransmitted | SYN-ACK Retransmission Aborted | |
| Portal | App Client Application Profile-un-Authorized Users | 0 | 0 | 0 | 0 | 0 | 0 | |
| | App Client Application Profile-Authorized Users | 0 | 0 | 0 | 0 | 0 | 0 | |

The **Agent Resource Metrics** dashboard will show in real time the resource utilization of the CyPerf test agents to understand if the test agents are the bottleneck (i.e., running out of resources):

| | | | | |
|--------------------------|------------------------|-----------|--------------------|------------------|
| Server Statistics | Agent Resource Metrics | | | |
| Server HTTP Statistics | Agent | CPU Usage | Max CPU Core Usage | Memory Available |
| TCP Statistics | 172.16.2.87 | 0.16% | 0.50% | 11.29 GiB |
| Agent Traffic Statistics | 172.16.2.146 | 0.14% | 0.40% | 11.27 GiB |
| Agent Resource Metrics | 172.16.2.140 | 0.10% | 0.20% | 11.31 GiB |
| HTTP | | | | |
| Portal | | | | |

This is especially useful when it is not clear if the DUT/SUT reached its performance limit or the CyPerf test agents are running out of resources. Of course, if the CyPerf test agents are indeed running out of resources users can always add more test agents into the test for increased capacity.

In the above graph we can see that the CyPerf agents are barely utilizing the assigned resources indicating that they are capable of much higher performance.

Real-time statistics also provide instant access to key statistics that must be examined for failures at the TCP, TLS, and HTTP protocol level.

Conclusions

The purpose of this test, was to validate that the least privilege access policies have been properly configured and are being enforced accordingly on the ZTNA PEP, all under traffic load. We have also looked at other related statistics such as latencies and potential failures.

This was the first basic step to validate the functionality of ZTNA proving the main building blocks before production launch making sure the expected outcome is achieved and real production users don't become beta testers.

Lab 2: Security effectiveness

Description

For the second lab we will focus on validating the ZTNA PEP's capabilities to block security attacks coming from a compromised user as well as its impact on the legitimate user traffic.

Therefore, in this lab, we will add one more CyPerf client test agent emulating the following role:

- **Compromised users:** these are remote users, with valid credentials that have been previously compromised and they will attempt certain security attacks against the server test agent that is protected by the ZTNA PEP.
 - These users will test that the DUT has been properly configured and is able to properly identify and block security attacks

The new CyPerf test agent that will emulate the compromised users, will first statefully authenticate with the ZTNA PEP (as those users have valid credentials as well) and once the authentication is successful they will try to transfer a malware package to the server test agent (as detailed in below step-by-step instructions).

Similar to the previous lab, the emulated security attacks will also be using TLS encryption which will again test the ZTNA PEP capability to detect encrypted attacks.

Objective

This objective of this test is to validate that the ZTNA PEP has been properly configured and is able to identify and block security attacks coming from internal users with valid credentials (lateral movement). In addition, we will also compare various other statistics with the previous test to understand the impact on the valid, authorized user traffic under attack scenarios.

To achieve this objective, we will monitor various CyPerf relevant Key Performance Indicators (KPIs), such as:

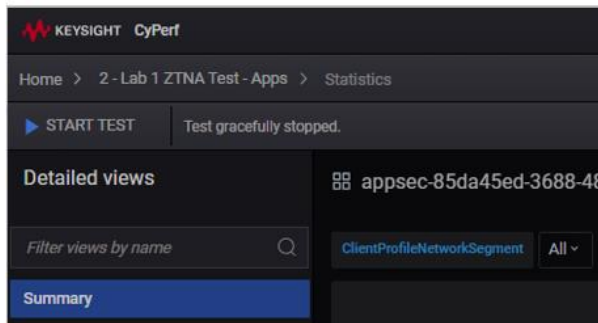
- Total attacks sent vs blocked or allowed
- Throughput
- Concurrent Simulated Users (or Concurrent Connections)
- Simulated Users per Second (or Connections per Second)
- Latencies (connection latency, first response latency, object transfer latency)
- Application iterations successful/failed
- Authentications successful/failed.
- TCP statistics (connections initiated vs failures, TCP retries/aborts etc)

Setup

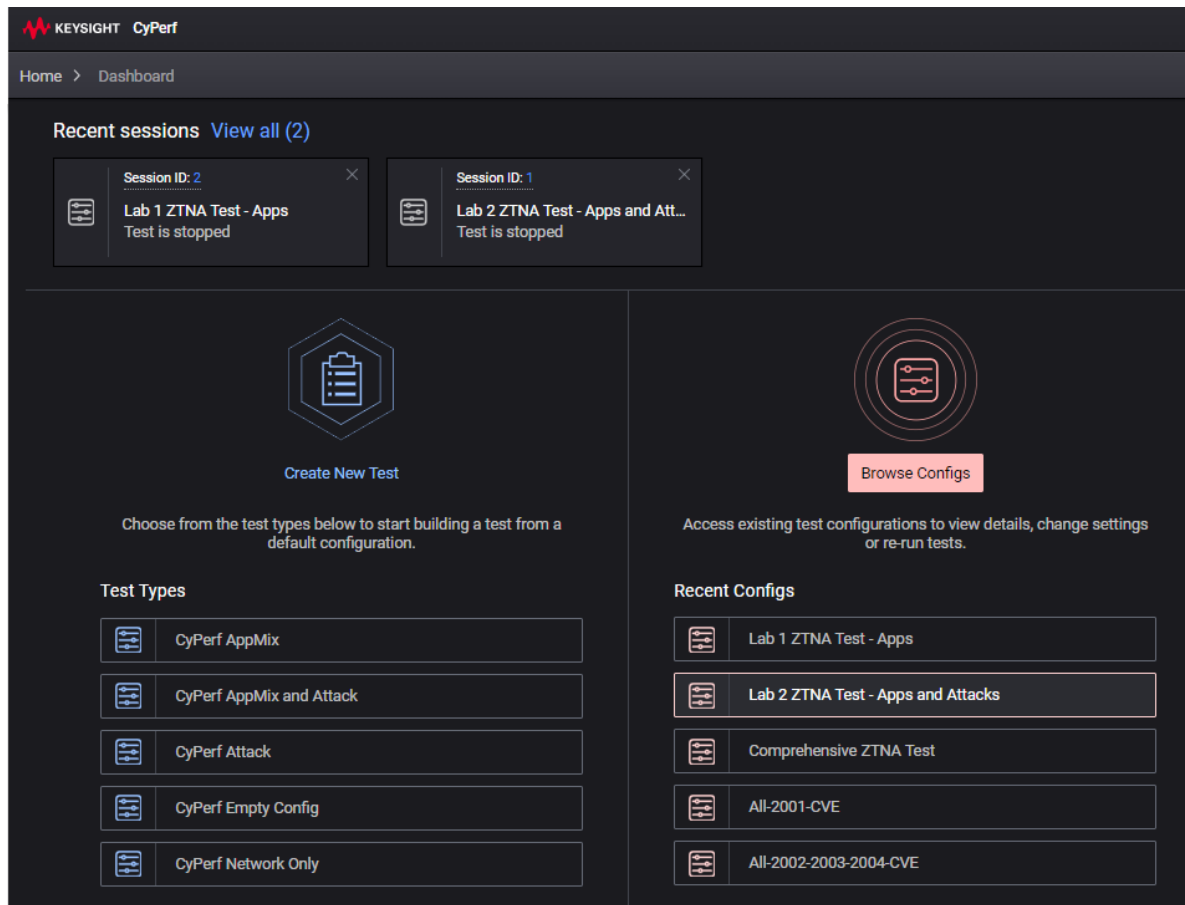
The setup for this lab is the same as the one described in the Introduction section.

Step-by-Step Instructions

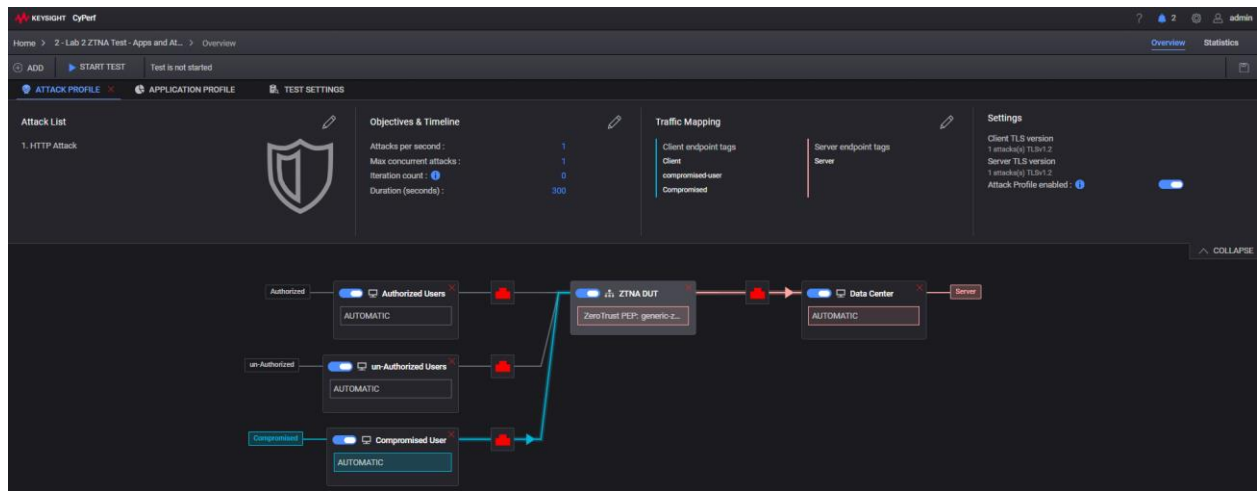
1. After completing the previous test, navigate to CyPerf Controller's landing page by clicking on the **Home** button:



2. In the Recent Configs area of the **Browse Configs** section click on the config “Lab 2: ZTNA Test – Apps and Attacks”:



The new config file loads into a new CyPerf Session:

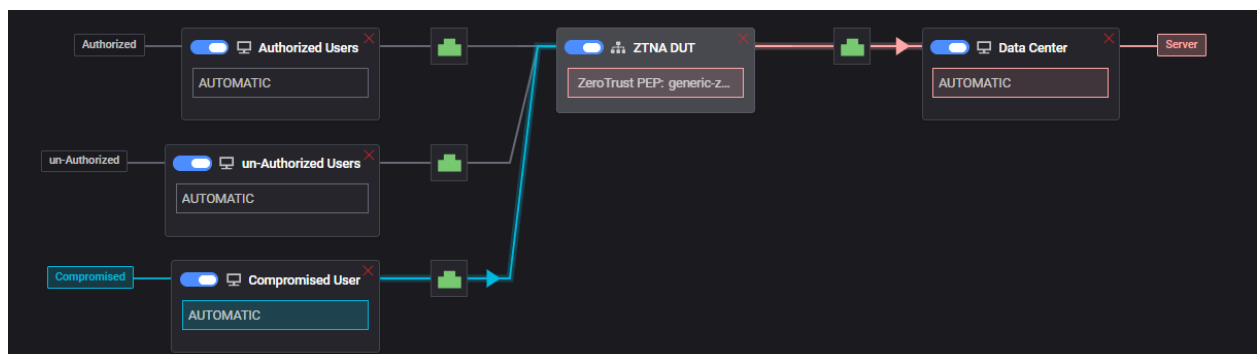


Compared to the first lab, you will notice one more user group: **Compromised Users**.


First, similar to the first lab, let's assign CyPerf test agents to each Network Segment as follows:

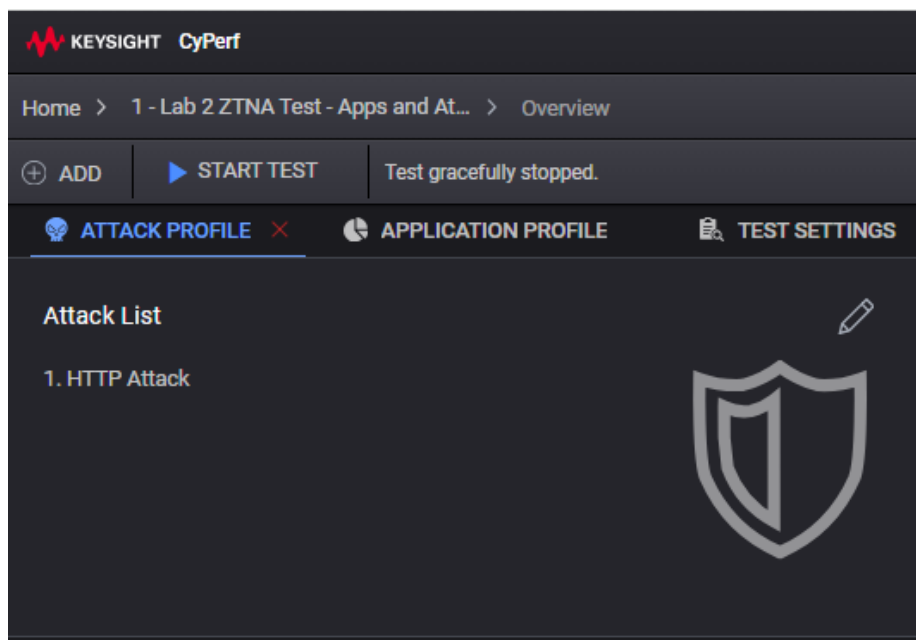
- For the **Authorized Users** Network Segment (click on the red port button) and assign the CyPerf agent with IP address: 172.16.1.11
- Repeat above steps for the **un-Authorized Users** Network Segment and assign the CyPerf agent with IP address: 172.16.1.12
- Repeat above steps for the **Compromised Users** Network Segment and assign the CyPerf agent with IP address: 172.16.1.13
- Finally, repeat above steps for the **Data Center** Network Segment and assign the CyPerf agent with IP address: 172.16.1.183

Now all Network Segments need to have a green port associated. If there is a single red port, please re-visits steps a-c above and make sure all Network Segments have a green port before proceeding.



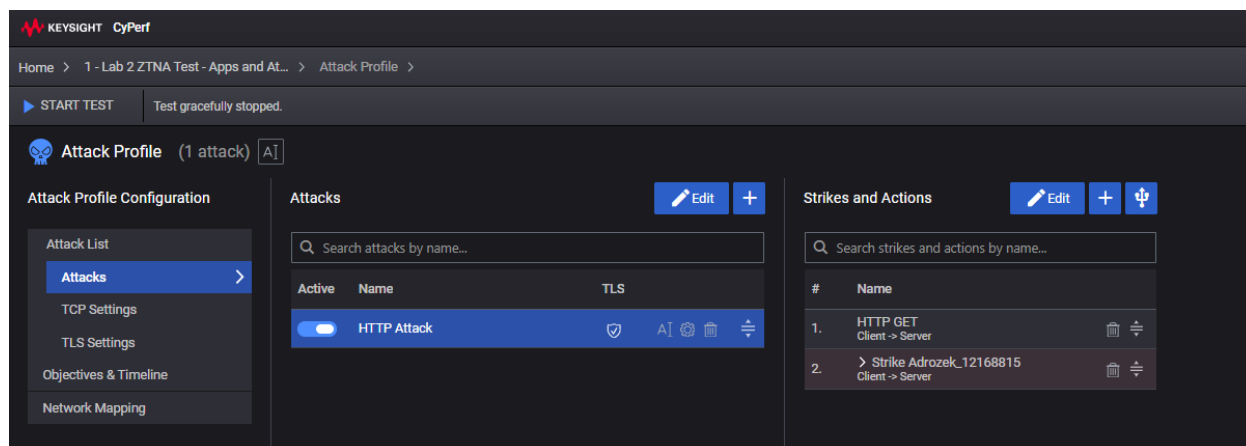
Also, you will notice that in the upper area of the UI, there is also an **Attack Profile** available along with the **Application Profile** (used in the previous test). Next, let's have a look at the configuration of this new **Attack Profile**.

3. **Attack Profile:** Select the edit icon () in the **Attack Profile** section:




The new menu provides access to all the Attack Profile options, like:

- **Attack list:** just as an example in this test we are using a simple attack (i.e. **HTTP Attack**) that transfers an malware sample known as “Adrozek” which can hook into web browsers and steal login credentials:



Also, we can notice that the actual malicious action is preceded by an legitimate HTTP GET as in a real attack (and of course since this is also a ZTNA user, it will first authenticate before sending any traffic):

Tip: CyPerf offers you the ability to add the attacks into the test that are more relevant for the type of DUT being tested and obviously the production environment as well:

- To access the attack library select the add button ( icon):

KEYSIGHT CyPerf

Home > 1 - Lab 2 ZTNA Test - Apps and At... > Attack Profile >

STOP TEST ABORT TEST

Attack Profile (1 attack) AI

Attack Profile Configuration

- Attack List
- Attacks**
- TCP Settings
- TLS Settings
- Objectives & Timeline
- Network Mapping

Attacks

Search attacks by name...

| Active | Name | TLS |
|-------------------------------------|-------------|--------------|
| <input checked="" type="checkbox"/> | HTTP Attack | ✓ AI ⚙️ 🗑️ ⚡ |

- The attack library comprises a comprehensive list of attacks containing multiple strikes, grouped based on certain categories, as well as individual strikes

Add Attack(s)

ATTACK LIBRARY CUSTOMIZE ATTACK

Select attack(s) from table

Filter attacks by name, description

50 attacks on this page

| | Name | Severity | Strikes | Direction |
|--------------------------|--|-----------------------|---------|-----------|
| <input type="checkbox"/> | All Encrypted Attacks | 7 critical, 9 high... | 24 | c2s |
| <input type="checkbox"/> | Apple Browser Attacks | 1 critical, 30 hig... | 56 | mixed |
| <input type="checkbox"/> | Auth Bypass Attacks | 2 critical, 6 high... | 15 | c2s |
| <input type="checkbox"/> | Brute Force Attack Top Usernames And Passwords | None | 1 | c2s |
| <input type="checkbox"/> | Chrome Browser Attacks | 1 critical, 13 hig... | 21 | s2c |
| <input type="checkbox"/> | Critical Strikes | Critical | 131 | mixed |
| <input type="checkbox"/> | CSRF Attacks | 1 high, 2 medium | 3 | mixed |
| <input type="checkbox"/> | DoS Attacks | 4 critical, 29 hig... | 93 | mixed |

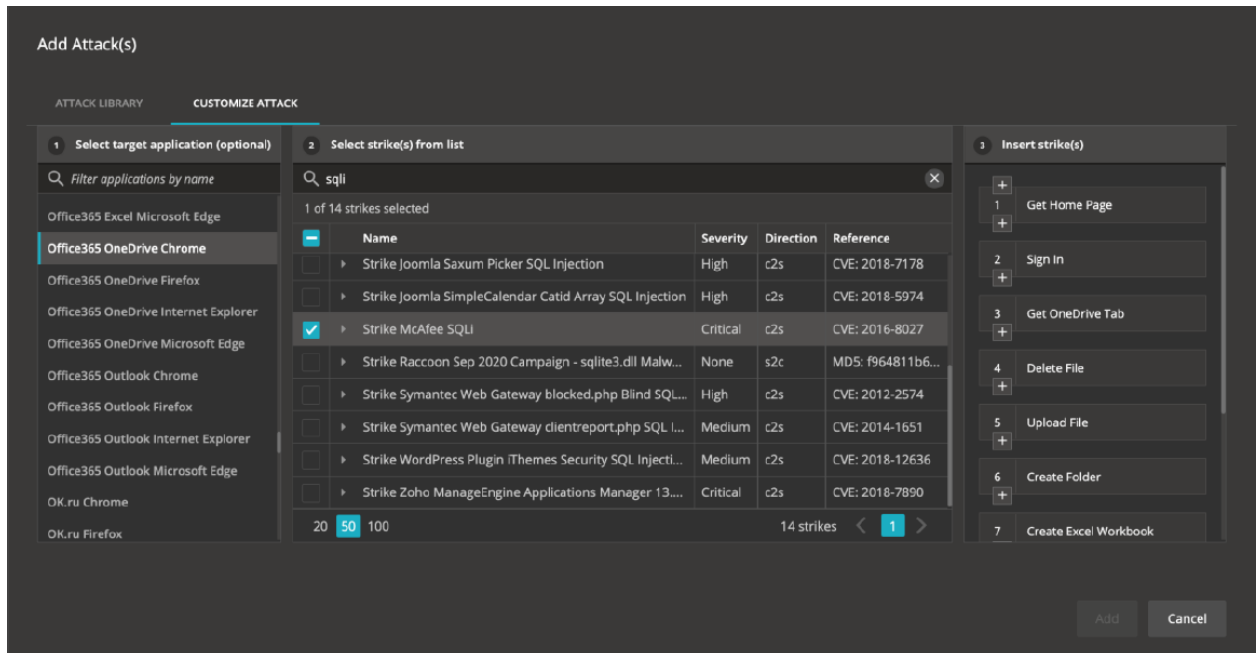
20 50 100 2937 attacks < 1 2 3 4 5 ... 59 >

Summary of added attacks (0)

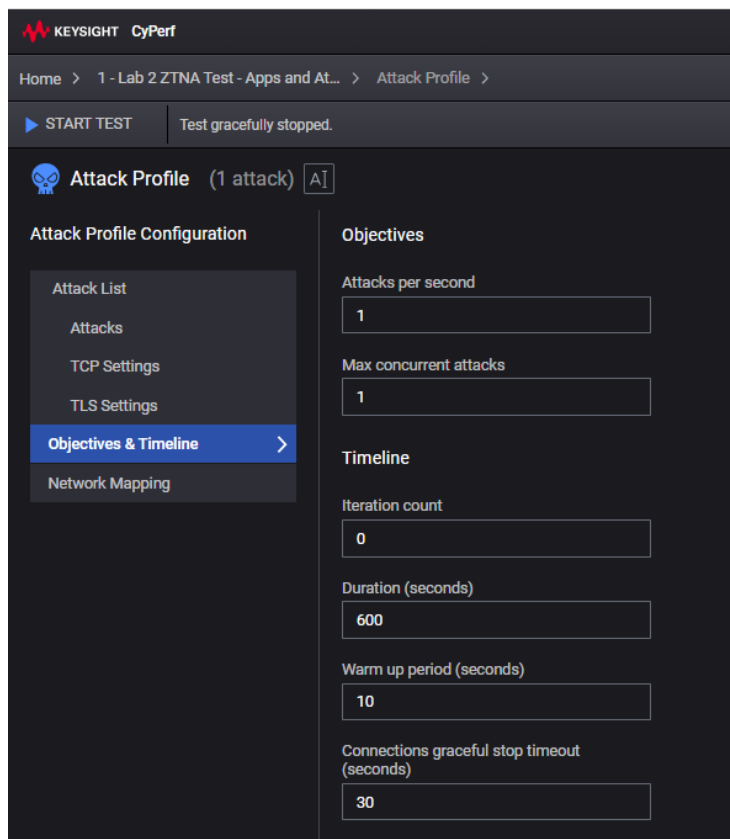
Select attack(s) and then hit the "Add to list" button to add it/them to your list

Add Cancel

Tip: You can also create custom attacks by combining legitimate application actions with malicious strikes allowing an unprecedented level of realism. This can be done either manually by adding desired application actions to an attack or through the Customize Attack pane in the Add Attack dialog:

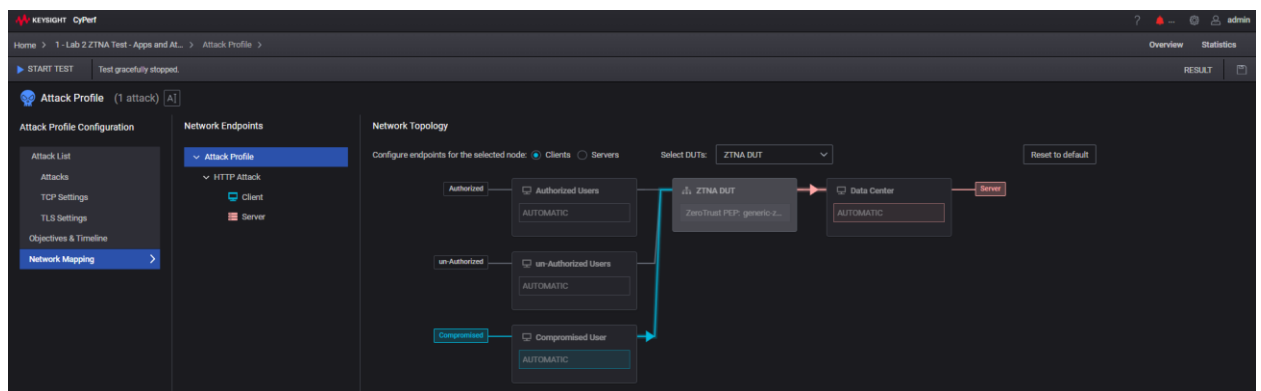


- For the same **Attack Profile**, under the **Objectives and Timeline** (in the left-hand navigation pane) the load objective and duration of the Attack profile has been configured as follows:



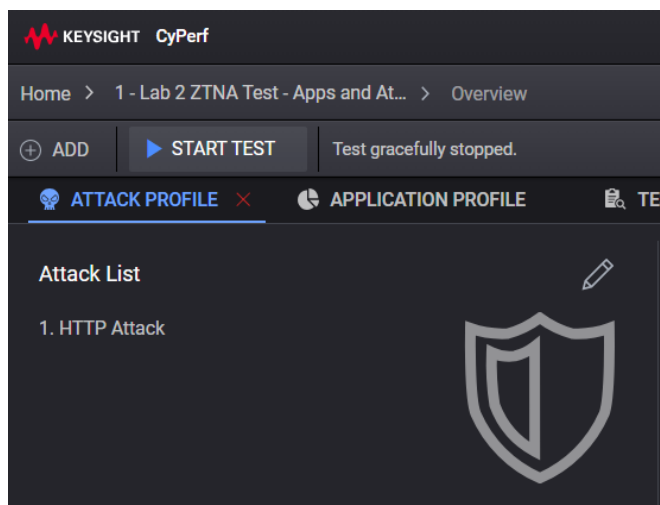
For simplicity we have configured to execute one attack per second for 600 seconds.

- For the same **Attack Profile**, under the Network Mapping (in the left-hand navigation pane) you can notice that this Attack Profile has been configured to run only from the **Compromised User** segment:



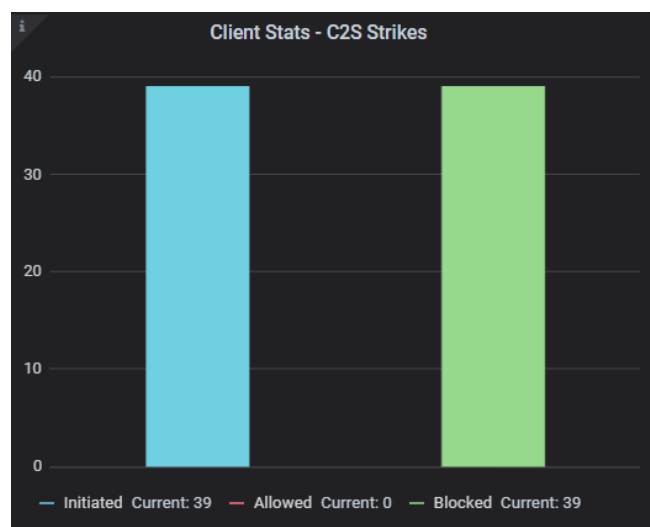
Now that we have seen the most important test parameter, we can run the test and interpret the results.

4. Select the **Start Test** button from the upper-left corner:



Real-Time Statistics

First, we will notice in the Summary dashboard new graphs for the security attacks being emulated by the “**compromised users**”:



All the attacks used in this test are client-to-server (c2s) and as we can see in above graph all the attacks are properly blocked by the ZTNA PEP. This is of course, a good and expected result as the emulated users running the attacks are having valid credentials only that those users have been compromised before hence now running attacks.

Indeed, if we navigate to **Client Zero Trust Statistics** dashboard, we can see that the authentication request of the **Compromised User** group has been successful:

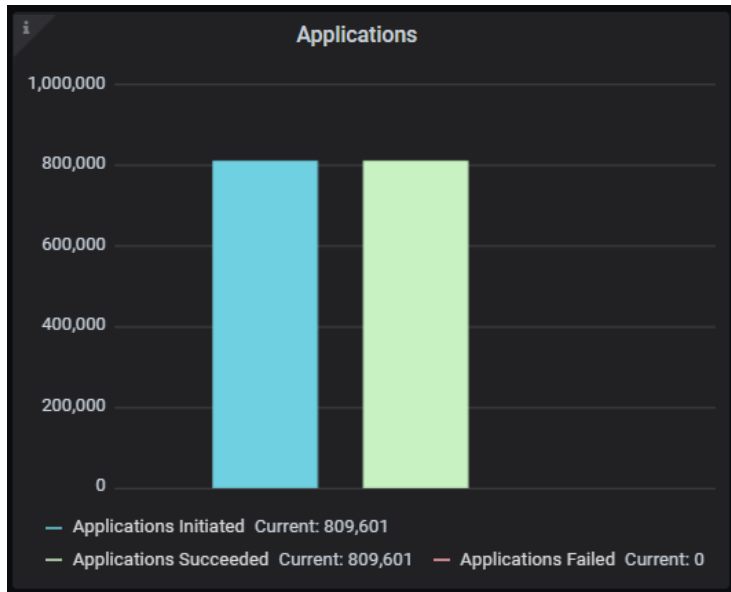
The screenshot shows the KeySight CyPerf interface with the 'Statistics' tab selected. The main content area displays 'Client Authentication Flow Statistics' for the profile 'appsec-9622975d-2bb9-4499-a212-b9894ec642ea / Client Zero Trust St'. The table below shows the statistics for different application and attack profiles.

| Profile - Network Segment | Application & Attack | IdP Requests Succeeded | Authentications Started | Authentications Succeeded | Authentications Failed | Authentications Reused |
|--|----------------------|------------------------|-------------------------|---------------------------|------------------------|------------------------|
| Attack Client Attack Profile-Compromised User | HTTP Attack | 1 | 1 | 1 | 0 | 0 |
| App Client Application Profile-un-Authorized Users | Restricted App | 0 | 0 | 0 | 0 | 0 |
| App Client Application Profile-Authorized Users | Authorized App | 491 | 491 | 491 | 0 | 459266 |

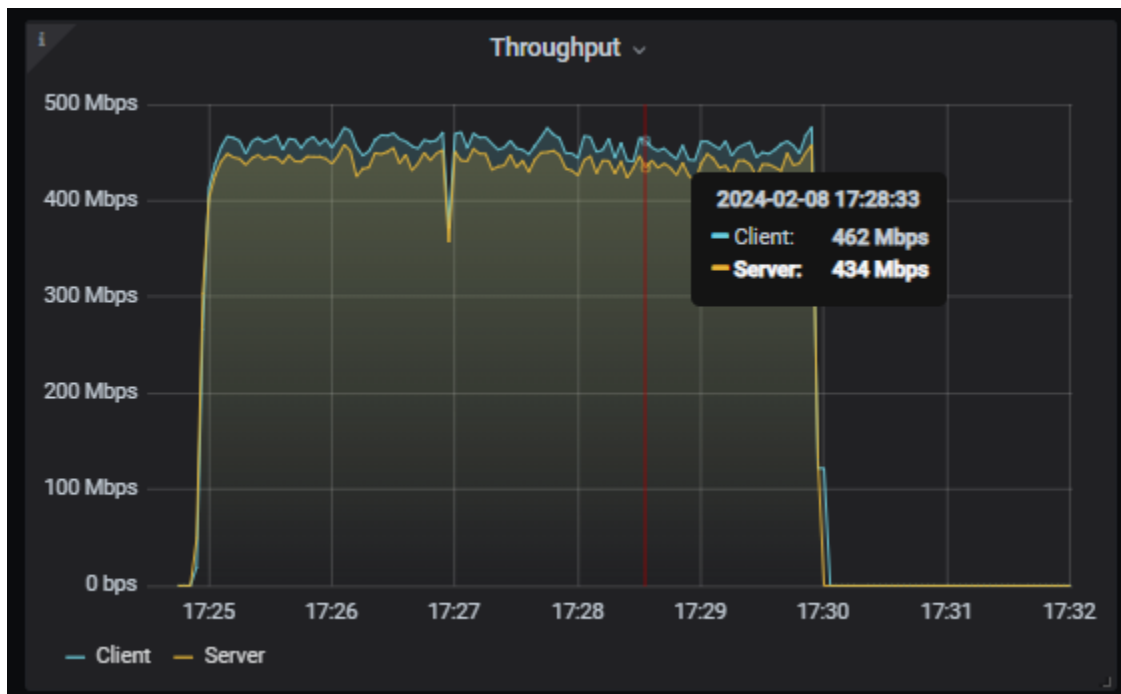
Next, another important point would be to quantify the impact of both the attacks as well as un-authorized requests over the legitimate, authorized users. For this, we will navigate back to the **Summary** dashboard, and we will apply a filter on **Authorized Users** (for the ClientProfileNetworkSegment in the upper-right corner) to only see the corresponding stats:

The screenshot shows the KeySight CyPerf interface with the 'Summary' tab selected. The main content area displays the 'Summary' dashboard for the profile 'appsec-9622975d-2bb9-4499-a212-b9894ec642ea'. A filter is applied to 'ClientProfileNetworkSegment' with the value 'App Client Application Profile-Authorized Users' selected from the dropdown menu.

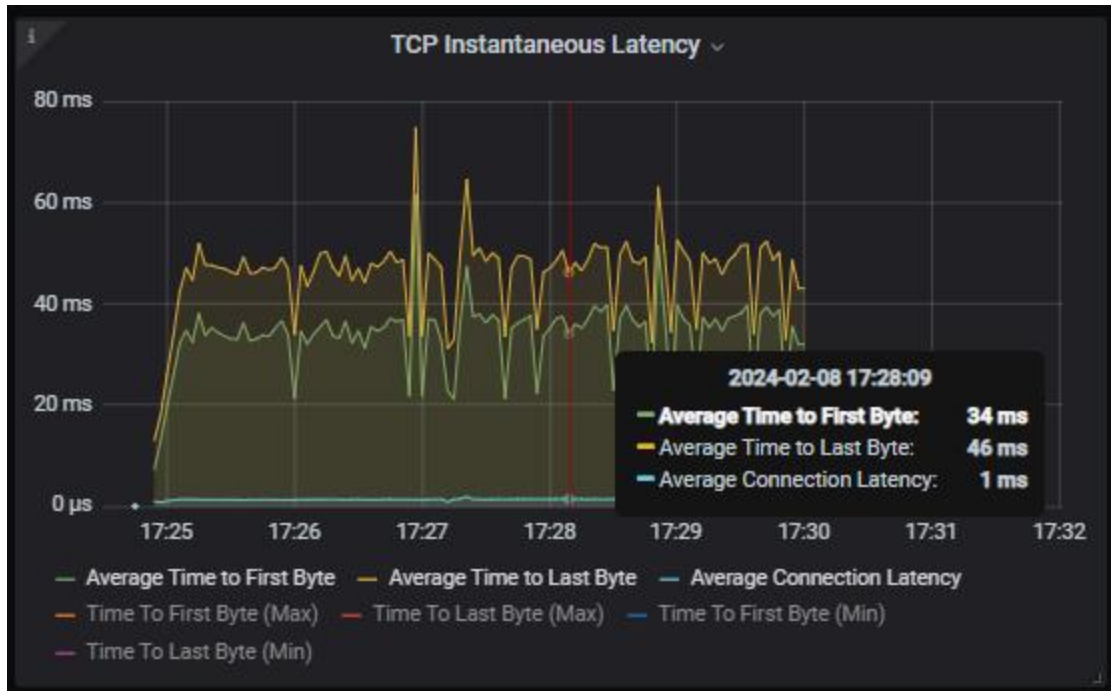
First thing to check is to if there are any application failures and in this case we can see that all the applications are successfully completed:



Also, using the same filter applied on **Authorized Users** we can see that that Throughput is just slightly lower compared to the results in the first Lab (462 Mbps vs 480Mbps):



Next important metrics are the application latencies:



Here, what we can notice is that there is a more significant increase for both the Time to First Byte (34ms vs 14ms) and Time to Last Byte (46ms vs 22ms) which of course is due to the extra processing that the DUT needs to perform while under attack. Depending on the end-user type of traffic and latency requirements this might or might not be something to further investigate.

Important

In general, adding or enabling more policies, features and functionalities on Network Security devices lead to an impact to the maximum performance the device can handle. Therefore, it is very important, when characterizing the performance of any DUT to make sure that its configuration reflects the production needs and requirements.

Furthermore, we can also check other detailed stats. For example if we navigate to **Client Statistics per Action** dashboard, we can see granular stats for each action of all applications:

| Profile - Network Segment | Application & Attack | Action | Started | Succeeded | Failed |
|--|----------------------|-------------------------|---------|-----------|---------|
| Attack Client Attack Profile-Compromised User | HTTP Attack | Strike Adrozek_12168815 | 201 | 0 | 201 |
| Attack Client Attack Profile-Compromised User | HTTP Attack | HTTP GET | 201 | 201 | 0 |
| App Client Application Profile-un-Authorized Users | Restricted App | Upload Image | 190,170 | 0 | 190,155 |
| App Client Application Profile-un-Authorized Users | Restricted App | Logout | 190,170 | 0 | 190,155 |
| App Client Application Profile-un-Authorized Users | Restricted App | Login | 190,170 | 0 | 190,170 |
| App Client Application Profile-Authorized Users | Authorized App | HTTP GET | 809,601 | 809,601 | 0 |

One thing to notice is that under the HTTP Attack there are two actions:

- the HTTP GET action which only has successful iterations
- the actual malicious strike (i.e., Strike Adrozek) which is blocked for each attempt.

Conclusions

In this second test, the true performance of the ZTNA PEP (DUT) is characterized using malicious traffic in conjunction with authorized and un-authorized traffic.

First, the ability of the ZTNA PEP (DUT) to detect and block malicious activity coming from authenticated users was tested. This is an important test as ZTNA is promising to protect against lateral movement therefore even authenticated users with valid credentials should have their request inspected as well.

In the same time, the impact of the security attack on the legitimate traffic was evaluated as well and we noticed a slight decrease in legitimate traffic throughput as well as an increase in terms of latencies.

In the rush to adopt new emerging technologies (like ZTNA, SASE etc.), organizations are finding difficulty in rightsizing resources and optimizing cost while minimizing user disruptions. Testing the performance of such environments and isolating bottlenecks is also challenging.

Install Keysight’s CyPerf agents to generate authenticated, unauthenticated and un-authorized application traffic and security attacks to validate zero trust authentication policies. Characterize the performance, scale, and responsiveness of the zero trust implementations by generating thousands of authentication requests.



For more information on Keysight Technologies' products, applications, or services, please visit: www.keysight.com

This information is subject to change without notice. © Keysight Technologies, 2022, Published in USA, July 07, 2022