Stream: Internet Engineering Task Force (IETF)
RFC: 9411
Obsoletes: 3511
Category: Informational
Published: March 2023
ISSN: 2070-1721

Authors: B. Balarajah
C. Rossenhoevel, *EANTC AG*
B. Monkman, *NetSecOPEN*

RFC 9411
Benchmarking Methodology for Network Security Device Performance

Abstract

This document provides benchmarking terminology and methodology for next-generation network security devices, including next-generation firewalls (NGFWs) and next-generation intrusion prevention systems (NGIPSs). The main areas covered in this document are test terminology, test configuration parameters, and benchmarking methodology for NGFWs and NGIPSs. (It is assumed that readers have a working knowledge of these devices and the security functionality they contain.) This document aims to improve the applicability, reproducibility, and transparency of benchmarks and to align the test methodology with today's increasingly complex layer 7 security-centric network application use cases. As a result, this document makes RFC 3511 obsolete.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

It has been 18 years since the IETF initially recommended test methodology and terminology for firewalls [RFC3511]. Firewalls have evolved significantly from the days of simple access control list (ACL) filters. As the underlying technology progresses and improves, recommending test methodology and terminology for firewalls, requirements, and expectations for network security elements has increased tremendously. Security function implementations have evolved and diversified into intrusion detection and prevention, threat management, analysis of encrypted traffic, and more. In an industry of growing importance, well-defined and reproducible key performance indicators (KPIs) are increasingly needed to enable fair and reasonable comparisons of network security functions. These reasons led to the creation of a new next-generation network security device benchmarking document, which makes [RFC3511] obsolete. The measurement of performance for processing IP-fragmented traffic (see Section 5.9 of [RFC3511])is not included in this document since IP fragmentation does not commonly occur in traffic anymore, unlike how it might have at the time when [RFC3511] was written. It should also be noted that [RFC2647] retains significant value and was consulted frequently while creating this document.

For a more detailed explanation of what an NGFW is, see the Wikipedia article [Wiki-NGFW].

2. Requirements Language

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Scope

This document provides testing terminology and testing methodology for modern and next-generation network security devices that are configured in Active ("Inline", see Figures 1 and 2) mode. It covers the validation of security effectiveness configurations of network security devices, followed by performance benchmark testing. This document focuses on advanced, realistic, and reproducible testing methods. Additionally, it describes testbed environments, test tool requirements, and test result formats.

The performance testing methodology described in this document is not intended for security devices or systems that rely on machine learning or behavioral analysis. If such features are present in a Device Under Test / System Under Test (DUT/SUT), they should be disabled.

4. Test Setup

The test setup defined in this document applies to all benchmarking tests described in Section 7. The test setup **MUST** be contained within an isolated test environment (see Section 3 of [RFC6815]).

4.1. Testbed Configuration

Testbed configuration **MUST** ensure that any performance implications that are discovered during the benchmark testing aren't due to the inherent physical network limitations, such as the number of physical links and forwarding performance capabilities (throughput and latency) of the network devices in the testbed. For this reason, this document recommends avoiding external devices, such as switches and routers, in the testbed wherever possible.

In some deployment scenarios, the network security devices (DUT/SUT) are connected to routers and switches, which will reduce the number of entries in MAC (Media Access Control) or Address Resolution Protocol / Neighbor Discovery (ARP/ND) tables of the DUT/SUT. If MAC or ARP/ND tables have many entries, this may impact the actual DUT/SUT performance due to MAC and ARP/ND table lookup processes. This document also recommends using test equipment with the capability of emulating layer 3 routing functionality instead of adding external routers in the testbed.

The testbed setup for Option 1 (Figure 1) is the **RECOMMENDED** testbed setup for the benchmarking test.

```
+-----------------------+                   +-----------------------+
| +-------------------+ |   +-----------+   | +-------------------+ |
| | Emulated Router(s)| |   |           |   | | Emulated Router(s)| |
| |    (Optional)     | +----- DUT/SUT  +-----+    (Optional)     | |
| +-------------------+ |   |           |   | +-------------------+ |
| +-------------------+ |   +-----------+   | +-------------------+ |
| |     Clients       | |                   | |      Servers      | |
| +-------------------+ |                   | +-------------------+ |
|                       |                   |                       |
|   Test Equipment      |                   |   Test Equipment      |
+-----------------------+                   +-----------------------+
```

Figure 1: Testbed Setup - Option 1

If the test equipment used is not capable of emulating OSI layer 3 routing functionality or if the number of used ports is mismatched between the test equipment and the DUT/SUT (which is needed for test equipment port aggregation), the test setup can be configured as shown in Figure 2.

```
 +-------------------+      +-----------+      +--------------------+
 |Aggregation Switch/|      |           |      | Aggregation Switch/|
 | Router            +------+  DUT/SUT  +------+ Router             |
 |                   |      |           |      |                    |
 +----------+--------+      +-----------+      +--------+-----------+
            |                                           |
            |                                           |
+-----------+-----------+                   +-----------+-----------+
|                       |                   |                       |
| +-------------------+ |                   | +-------------------+ |
| | Emulated Router(s)| |                   | | Emulated Router(s)| |
| |     (Optional)    | |                   | |     (Optional)    | |
| +-------------------+ |                   | +-------------------+ |
| +-------------------+ |                   | +-------------------+ |
| |      Clients      | |                   | |      Servers      | |
| +-------------------+ |                   | +-------------------+ |
|                       |                   |                       |
|    Test Equipment     |                   |    Test Equipment     |
+-----------------------+                   +-----------------------+
```

Figure 2: Testbed Setup - Option 2

4.2. DUT/SUT Configuration

The same DUT/SUT configuration **MUST** be used for all benchmarking tests described in Section 7. Since each DUT/SUT will have its own unique configuration, users **MUST** configure their devices with the same parameters and security features that would be used in the actual deployment of the device or a typical deployment. The DUT/SUT **MUST** be configured in "Inline" mode so that the traffic is actively inspected by the DUT/SUT.

Tables 2 and 3 below describe the **RECOMMENDED** and **OPTIONAL** sets of network security features for NGFWs and NGIPSs, respectively. If the recommended security features are not enabled in the DUT/SUT for any reason, the reason **MUST** be reported with the benchmarking test results. For example, one reason for not enabling the anti-virus feature in an NGFW may be that this security feature was not required for a particular customer deployment scenario. It **MUST** be also noted in the benchmarking test report that not enabling the specific recommended security features may impact the performance of the DUT/SUT. The selected security features **MUST** be consistently enabled on the DUT/SUT for all benchmarking tests described in Section 7.

To improve repeatability, a summary of the DUT/SUT configuration, including a description of all enabled DUT/SUT features, **MUST** be published with the benchmarking results.

The following table provides a brief description of the security feature; these are approximate taxonomies of features commonly found in currently deployed NGFWs and NGIPSs. The features provided by specific implementations may be named differently and not necessarily have configuration settings that align with the taxonomy.

DUT/SUT Features	Description
TLS Inspection	The DUT/SUT intercepts and decrypts inbound HTTPS traffic between servers and clients. Once the content inspection has been completed, the DUT/SUT encrypts the HTTPS traffic with ciphers and keys used by the clients and servers. For TLS 1.3, the DUT works as a middlebox (proxy) and holds the certificates and Pre-Shared Keys (PSKs) that are trusted by the client and represent the identity of the real server.
IDS/IPS	The DUT/SUT detects and blocks exploits targeting known and unknown vulnerabilities across the monitored network.
Anti-Malware	The DUT/SUT detects and prevents the transmission of malicious executable code and any associated communications across the monitored network. This includes data exfiltration as well as command and control channels.
Anti-Spyware	Anti-Spyware is a subcategory of Anti-Malware. Spyware transmits information without the user's knowledge or permission. The DUT/SUT detects and blocks the initial infection or transmission of data.
Anti-Botnet	The DUT/SUT detects and blocks traffic to or from botnets.
Anti-Evasion	The DUT/SUT detects and mitigates attacks that have been obfuscated in some manner.
Web Filtering	The DUT/SUT detects and blocks malicious websites, including defined classifications of websites across the monitored network.
Data Loss Protection (DLP)	The DUT/SUT detects and prevents data breaches and data exfiltration, or it detects and blocks the transmission of sensitive data across the monitored network.
Certificate Validation	The DUT/SUT validates certificates used in encrypted communications across the monitored network.
Logging and Reporting	The DUT/SUT logs and reports all traffic at the flow level across the monitored network.
Application Identification	The DUT/SUT detects known applications as defined within the traffic mix selected across the monitored network.
Deep Packet Inspection (DPI)	The DUT/SUT inspects the content of the data packet.

Table 1: Security Feature Description

DUT/SUT (NGFW) Features	RECOMMENDED	OPTIONAL
TLS Inspection	x	
IDS/IPS	x	
Anti-Spyware	x	
Anti-Virus	x	
Anti-Botnet	x	
Anti-Evasion	x	
Web Filtering		x
Data Loss Protection (DLP)		x
DDoS Protection		x
Certificate Validation		x
Application Identification	x	

Table 2: NGFW Security Features

DUT/SUT (NGIPS) Features	RECOMMENDED	OPTIONAL
TLS Inspection	x	
Anti-Malware	x	
Anti-Spyware	x	
Anti-Botnet	x	
Application Identification	x	
Deep Packet Inspection (DPI)	x	
Anti-Evasion	x	

Table 3: NGIPS Security Features

Note: With respect to TLS Inspection, there are scenarios where it will be optional.

Below is a summary of the DUT/SUT configuration:

- The DUT/SUT **MUST** be configured in "Inline" mode.
- "Fail-Open" behavior **MUST** be disabled.

- All **RECOMMENDED** security features are enabled.
- Logging and reporting **MUST** be enabled. The DUT/SUT **SHOULD** log all traffic at the flow level (5-tuple). If the DUT/SUT is designed to log all traffic at different levels (e.g., IP packet levels), it is acceptable to conduct tests. However, this **MUST** be noted in the test report. Logging to an external device is permissible.
- Geographical location filtering **SHOULD** be configured. If the DUT/SUT is not designed to perform geographical location filtering, it is acceptable to conduct tests without this feature. However, this **MUST** be noted in the test report.
- Application Identification and Control **MUST** be configured to trigger applications from the defined traffic mix.

In addition, a realistic number of access control lists (ACLs) **SHOULD** be configured on the DUT/SUT where ACLs are configurable and reasonable based on the deployment scenario. For example, it is acceptable not to configure ACLs in an NGIPS since NGIPS devices do not require the use of ACLs in most deployment scenarios. This document determines the number of access policy rules for four different classes of the DUT/SUT: Extra Small (XS), Small (S), Medium (M), and Large (L). A sample DUT/SUT classification is described in Appendix B.

The ACLs defined in Table 4 **MUST** be configured from top to bottom in the correct order, as shown in the table. This is due to ACL types listed in specificity-decreasing order, with "block" first, followed by "allow", representing a typical ACL-based security policy. The ACL entries **MUST** be configured with routable IP prefixes by the DUT/SUT, where applicable. (Note: There will be differences between how security vendors implement ACL decision making.) The configured ACL **MUST NOT** block the test traffic used for the benchmarking tests.

				DUT/SUT Classification # Rules			
Rules Type	**Match Criteria**	**Description**	**Action**	**XS**	**S**	**M**	**L**
Application layer	Application	Any application not included in the measurement traffic	block	5	10	20	50
Transport layer	SRC IP and TCP/UDP DST ports	Any SRC IP prefix used and any DST ports not used in the measurement traffic	block	25	50	100	250
IP layer	SRC/DST IP	Any SRC/DST IP subnet not used in the measurement traffic	block	25	50	100	250

				DUT/SUT Classification # Rules			
Rules Type	Match Criteria	Description	Action	XS	S	M	L
Application layer	Application	Half of the applications included in the measurement traffic (see the note below)	allow	10	10	10	10
Transport layer	SRC IP and TCP/UDP DST ports	Half of the SRC IPs used and any DST ports used in the measurement traffic (one rule per subnet)	allow	>1	>1	>1	>1
IP layer	SRC IP	The rest of the SRC IP prefix range used in the measurment traffic (one rule per subnet)	allow	>1	>1	>1	>1

Table 4: DUT/SUT Access List

Note 1: Based on the test customer's specific use case, the testers can increase the number of rules.

Note 2: If half of the applications included in the test traffic are less than 10, the missing number of ACL entries (placeholder rules) can be configured for any application traffic not included in the test traffic.

Note 3: In the event that the DUT/SUT is designed to not use ACLs, it is acceptable to conduct tests without them. However, this **MUST** be noted in the test report.

4.2.1. Security Effectiveness Configuration

The selected security features (defined in Tables 2 and 3) of the DUT/SUT **MUST** be configured effectively to detect, prevent, and report the defined security vulnerability sets. This section defines the selection of the security vulnerability sets from the Common Vulnerabilities and Exposures (CVEs) list [CVE] for testing. The vulnerability set should reflect a minimum of 500 CVEs from no older than 10 calendar years to the current year. These CVEs should be selected with a focus on in-use software commonly found in business applications, with a Common Vulnerability Scoring System (CVSS) Severity of High (7-10).

This document is primarily focused on performance benchmarking. However, it is **RECOMMENDED** to validate the security features configuration of the DUT/SUT by evaluating the security effectiveness as a prerequisite for performance benchmarking tests defined in Section 7.

In case the benchmarking tests are performed without evaluating security effectiveness, the test report **MUST** explain the implications of this. The methodology for evaluating security effectiveness is defined in Appendix A.

4.3. Test Equipment Configuration

In general, test equipment allows configuring parameters in different protocol layers. Extensive proof-of-concept tests conducted to support preparation of this document showed that benchmarking results are strongly affected by the choice of protocol stack parameters, especially OSI layer 4 transport protocol parameters. For more information on how TCP and QUIC parameters will impact performance, review [fastly]. To achieve reproducible results that will be representative of real deployment scenarios, careful specification and documentation of the parameters are required.

This section specifies common test equipment configuration parameters applicable for all benchmarking tests defined in Section 7. Any benchmarking-test-specific parameters are described under the test setup section of each benchmarking test individually.

4.3.1. Client Configuration

This section specifies which parameters should be considered while configuring emulated client endpoints in the test equipment. Also, this section specifies the **RECOMMENDED** values for certain parameters. The values are the defaults typically used in most of the client operating system types.

Pre-standard evaluations have shown that it is possible to set a wide range of arbitrary parameters for OSI layer 4 transport protocols on test equipment leading to optimization of client-specific results; however, only well-defined common parameter sets help to establish meaningful and comparable benchmarking results. For these reasons, this document recommends specific sets of transport protocol parameters to be configured on test equipment used for benchmarking.

4.3.1.1. TCP Stack Attributes

The TCP stack of the emulated client endpoints **MUST** fulfill the TCP requirements defined in Appendix B of [RFC9293]. In addition, this section specifies the **RECOMMENDED** values for TCP parameters configured using the parameters described below.

The IPv4 and IPv6 Maximum Segment Sizes (MSSs) are set to 1460 bytes and 1440 bytes, respectively. TX and RX initial receive window sizes are set to 65535 bytes. The client's initial congestion window should not exceed 10 times the MSS. Delayed ACKs are permitted, and the maximum client delayed ACK should not exceed 10 times of the MSS before a forced ACK; also, the maximum delayed ACK timer is allowed to be set to 200 ms. Up to three retries are allowed before a timeout event is declared. The TCP PSH flag is set to high in all traffic. The source port range is 1024-65535. The clients initiate TCP connections via a three-way handshake (SYN, SYN/ACK, ACK) and close TCP connections via either a TCP three-way close (FIN, FIN/ACK, ACK) or a TCP four-way close (FIN, ACK, FIN, ACK).

4.3.1.2. QUIC Specification

QUIC stack emulation on the test equipment **MUST** conform to [RFC9000] and [RFC9001]. This section specifies the **RECOMMENDED** values for certain QUIC parameters to be configured on test equipment used for benchmarking purposes only. The QUIC stream type (defined in Section 2.1 of [RFC9000]) is set to "Client-Initiated, Bidirectional". 0-RTT and early data are disabled. The QUIC connection termination method is an immediate close (Section 10.2 of [RFC9000]). Flow control is enabled. UDP payloads are set to the datagram size of 1232 bytes for IPv6 and 1252 bytes for IPv4. In addition, transport parameters and default values defined in Section 18.2 of [RFC9000] are **RECOMMENDED** to configure on test equipment. Also, this document references Appendices B.1 and B.2 of [RFC9002] for congestion-control-related constants and variables. Any configured QUIC and UDP parameter **MUST** be documented in the test report.

4.3.1.3. Client IP Address Space

The client IP space contains the following attributes.

- If multiple IP blocks are used, they **MUST** consist of multiple unique, discontinuous static address blocks.
- A default gateway **MAY** be used.
- The differentiated services code point (DSCP) marking should be set to Default Forwarding (DF) '000000' on the IPv4 Type of Service (ToS) field and IPv6 Traffic Class field.
- One or more extension headers **MAY** be used for IPv6 clients. If multiple extension headers are needed for traffic emulation, this document references [RFC8200] to choose the correct order of the extension headers within an IPv6 packet. Testing with one or more extension headers may impact the performance of the DUT. The extension headers **MUST** be documented and reported.

The following equation can be used to define the total number of client IP addresses that need to be configured on the test equipment.

> Desired total number of client IP addresses = Target throughput [Mbit/s] / Average throughput per IP address [Mbit/s]

As shown in the example list below, the value for "Average throughput per IP address" can be varied depending on the deployment and use case scenario.

Example 1 DUT/SUT deployment scenario 1: 6-7 Mbit/s per IP (e.g., 1,400-1,700 IPs per 10 Gbit/s of throughput)

Example 2 DUT/SUT deployment scenario 2: 0.1-0.2 Mbit/s per IP (e.g., 50,000-100,000 IPs per 10 Gbit/s of throughput)

Client IP addresses **MUST** be distributed between IPv4 and IPv6 based on the deployment and use case scenario. The following options **MAY** be considered for a selection of ratios for both IP addresses and traffic load distribution.

Option 1	100 % IPv4, no IPv6
Option 2	80 % IPv4, 20% IPv6
Option 3	50 % IPv4, 50% IPv6
Option 4	20 % IPv4, 80% IPv6
Option 5	no IPv4, 100% IPv6

Note: IANA has assigned IP address ranges for testing purposes, as described in Section 8. If the test scenario requires more IP addresses or subnets than IANA has assigned, this document recommends using private IPv4 address ranges or Unique Local Address (ULA) IPv6 address ranges for the testing.

4.3.1.4. Emulated Web Browser Attributes

The client (emulated web browser) contains attributes that will materially affect the traffic load. The objective is to emulate modern, typical browser attributes to improve the relevance of the result set for typical deployment scenarios.

The emulated browser **MUST** negotiate HTTP version 1.1 or higher. The emulated browser **SHOULD** advertise a User-Agent header. The emulated browser **MUST** enforce content length validation. HTTP header compression **MAY** be set to enable. If HTTP header compression is configurable in the test equipment, it **MUST** be documented if it was enabled or disabled. Depending on test scenarios and the chosen HTTP version, the emulated browser **MAY** open multiple TCP or QUIC connections per server endpoint IP at any time, depending on how many sequential transactions need to be processed.

For HTTP/2 traffic emulation, the emulated browser opens multiple concurrent streams per connection (multiplexing). For HTTPS requests, the emulated browser **MUST** send an "h2" protocol identifier using the TLS extension Application-Layer Protocol Negotiation (ALPN). The following default values (see [Undertow]) are the **RECOMMENDED** settings for certain HTTP/2 parameters to be configured on test equipment used for benchmarking purposes only:

- Maximum frame size: 16384 bytes
- Initial window size: 65535 bytes
- HPACK header field table size: 4096 bytes
- Server push enable: false (Note: In [Undertow], the default setting is true. However, for testing purposes, this document recommends setting the value to false for server push.)

This document refers to [RFC9113] for further details of HTTP/2. If any additional parameters are used to configure the test equipment, they **MUST** be documented.

For HTTP/3 traffic emulation, the emulated browsers initiate secure QUIC connections using TLS 1.3 ([RFC9001] describes how TLS is used to secure QUIC). This document refers to [RFC9114] for HTTP/3 specifications. The specification for transport protocol parameters is defined in Section 4.3.1.2. QPACK configuration settings, such as MAX_TABLE_CAPACITY and QPACK_BLOCKED_STREAMS, are set to zero (default), as defined in [RFC9204]. Any HTTP/3 parameters used for test equipment configuration **MUST** be documented.

For encrypted traffic, the following attributes are defined as the negotiated encryption parameters. The test clients **MUST** use TLS version 1.2 or higher. The TLS record size **MAY** be optimized for the HTTPS response object size, up to a record size of 16 KB. If Server Name Indication (SNI) is required (especially if the server is identified by a domain name), the client endpoint **MUST** send TLS extension SNI information when opening a security tunnel. Each client connection **MUST** perform a full TLS handshake, and session reuse or resumption **MUST** be disabled. (Note: Real web browsers use session reuse or resumption. However, for testing purposes, this feature must not be used to measure the DUT/SUT performance in the worst-case scenario.)

The following ciphers and keys supported by TLS 1.2 are **RECOMMENDED** for the HTTPS-based benchmarking tests defined in Section 7.

1. ECDHE-ECDSA-AES128-GCM-SHA256 with Prime256v1 (Signature Hash Algorithm: ecdsa_secp256r1_sha256 and Supported group: secp256r1)
2. ECDHE-RSA-AES128-GCM-SHA256 with RSA 2048 (Signature Hash Algorithm: rsa_pkcs1_sha256 and Supported group: secp256r1)
3. ECDHE-ECDSA-AES256-GCM-SHA384 with Secp384r1 (Signature Hash Algorithm: ecdsa_secp384r1_sha384 and Supported group: secp384r1)
4. ECDHE-RSA-AES256-GCM-SHA384 with RSA 4096 (Signature Hash Algorithm: rsa_pkcs1_sha384 and Supported group: secp384r1)

Note: The above ciphers and keys were those commonly used for enterprise-grade encryption cipher suites for TLS 1.2 at of the time of publication (2023). Individual certification bodies should use ciphers and keys that reflect evolving use cases. These choices **MUST** be documented in the resulting test reports with detailed information on the ciphers and keys used, along with reasons for the choices.

IANA recommends the following cipher suites for use with TLS 1.3, as defined in [RFC8446].

1. TLS_AES_128_GCM_SHA256
2. TLS_AES_256_GCM_SHA384
3. TLS_CHACHA20_POLY1305_SHA256
4. TLS_AES_128_CCM_SHA256

4.3.2. Backend Server Configuration

This section specifies which parameters should be considered while configuring emulated backend servers using test equipment.

4.3.2.1. TCP Stack Attributes

The TCP stack on the server-side **MUST** be configured similarly to the client-side configuration described in Section 4.3.1.1.

4.3.2.2. QUIC Specification

The QUIC parameters on the server-side **MUST** be configured similarly to the client-side configuration. Any configured QUIC parameter **MUST** be documented in the report.

4.3.2.3. Server Endpoint IP Addressing

The sum of the server IP space **MUST** contain the following attributes.

- The server IP blocks **MUST** consist of unique, discontinuous static address blocks with one IP per server Fully Qualified Domain Name (FQDN) endpoint per test port.
- A default gateway is permitted. The DSCP marking is set to DF '000000' on the IPv4 ToS field and IPv6 Traffic Class field. One or more extension headers for the IPv6 server are permitted. If multiple extension headers are required, this document references [RFC8200] to choose the correct order of the extension headers within an IPv6 packet.
- The server IP address distribution between IPv4 and IPv6 **MUST** be identical to the client IP address distribution ratio.

Note: IANA has assigned IP address blocks for the testing purpose described in Section 8. If the test scenario requires more IP addresses or address blocks than IANA has assigned, this document recommends using private IPv4 address ranges or Unique Local Address (ULA) IPv6 address ranges for the testing.

4.3.2.4. HTTP/HTTPS Server Pool Endpoint Attributes

The HTTP 1.1 and HTTP/2 server pools listen on TCP ports 80 and 443 for HTTP and HTTPS. The HTTP/3 server pool listens on any UDP port. The server **MUST** emulate the same HTTP version (HTTP 1.1, HTTP/2, or HTTP/3) and settings chosen by the client (emulated web browser). For the HTTPS server, TLS version 1.2 or higher **MUST** be used with a maximum record size of 16 KB. Ticket resumption or session ID reuse **MUST NOT** be used for TLS 1.2; also, session ticket or session cache **MUST NOT** be used for TLS 1.3. The server **MUST** serve a certificate to the client. The cipher suite and key size on the server-side **MUST** be configured similarly to the client-side configuration described in Section 4.3.1.4.

4.3.3. Traffic Flow Definition

At the beginning of the test (the init phase; see Section 4.3.4), the server endpoint initializes, and the server endpoint will be ready to accept TCP or QUIC connections as well as inbound HTTP and HTTPS requests. The client endpoints initialize and are given attributes such as a MAC and IP address. After the init phase of the test, each client sweeps through the given server IP space, generating a service recognizable by the DUT. Sequential and pseudorandom sweep methods are acceptable. The method used **MUST** be stated in the final report. Thus, a balanced mesh between client endpoints and server endpoints will be generated in a client IP and port to server IP and port combination. Each client endpoint performs the same actions as other endpoints, with the difference being the source IP of the client endpoint and the target server IP pool. The client **MUST** use the server IP address or FQDN in the host header.

4.3.3.1. Description of Intra-Client Behavior

Client endpoints are independent of other clients that are concurrently executing. When a client endpoint initiates traffic, this section describes how the client steps through different services. Once the test is initialized, the client endpoints randomly hold (perform no operation) for a few milliseconds for better randomization of the start of client traffic. Each client (HTTP 1.1 or HTTP/2) will either open a new TCP connection or connect to an HTTP persistent connection that is still open to that specific server. HTTP/3 clients will open UDP streams within QUIC connections. At any point that the traffic profile may require encryption, a TLS encryption tunnel will form, presenting the URL or IP address request to the server. If using SNI, the server **MUST** then perform an SNI name check by comparing the proposed FQDN to the domain embedded in the certificate. Only when correct will the server process the HTTPS response object. The initial response object to the server is based on benchmarking tests described in Section 7. Multiple additional sub-URLs (response objects on the service page) **MAY** be requested simultaneously. This **MAY** be to the same server IP as the initial URL. Each sub-object will also use a canonical FQDN and URL path.

4.3.4. Traffic Load Profile

The loading of traffic is described in this section. The loading of a traffic load profile has five phases: Init, ramp up, sustain, ramp down, and collection.

Init phase:
: Testbed devices, including the client and server endpoints, should negotiate layer 2-3 connectivity, such as MAC learning and ARP/ND. Only after successful MAC learning or ARP/ND **SHALL** the test iteration move to the next phase. No measurements are made in this phase. The minimum recommended time for the Init phase is 5 seconds. During this phase, the emulated clients **MUST NOT** initiate any sessions with the DUT/SUT; in contrast, the emulated servers should be ready to accept requests from the DUT/SUT or emulated clients.

Ramp Up phase:
: The test equipment **MUST** start to generate the test traffic. It **MUST** use a set of the approximate number of unique client IP addresses to generate traffic. The traffic **MUST** ramp up from zero to the desired target objective. The target objective is defined for each benchmarking test. The duration for the ramp up phase **MUST** be configured long enough that the test equipment does not overwhelm the DUT's/SUT's stated performance metrics defined in Section 6.3, namely TCP or QUIC connections per second, inspected throughput, concurrent TCP or QUIC connections, and application transactions per second. No measurements are made in this phase.

Sustain phase:
: This phase starts when all required clients are active and operating at their desired load condition. In the sustain phase, the test equipment **MUST** continue generating traffic to a constant target value for a constant number of active clients. The minimum **RECOMMENDED** time duration for the sustain phase is 300 seconds. This is the phase where measurements

occur. The test equipment **MUST** measure and record statistics continuously. The sampling interval for collecting the raw results and calculating the statistics **MUST** be less than 2 seconds.

Ramp Down phase:
The test traffic slows down from the target number to 0, and no measurements are made.

Collection phase:
The last phase is administrative and will occur when the test equipment merges and collates the report data.

5. Testbed Considerations

This section describes steps for a reference test (pre-test) that controls the test environment, including test equipment, focusing on physical and virtualized environments and test equipment. Below are the **RECOMMENDED** steps for the reference test.

1. Perform the reference test either by configuring the DUT/SUT in the most trivial setup (fast forwarding) or without the presence of the DUT/SUT.
2. Generate traffic from the traffic generator. Choose a traffic profile used for the HTTP or HTTPS throughput performance test with the smallest object size.
3. Ensure that any ancillary switching or routing functions added in the test equipment do not limit performance by introducing packet loss or latency. This is specifically important for virtualized components (e.g., vSwitches or vRouters).
4. Verify that the generated traffic (performance) of the test equipment matches and reasonably exceeds the expected maximum performance of the DUT/SUT.
5. Record the network performance metrics packet loss and latency introduced by the test environment (without the DUT/SUT).
6. Assert that the testbed characteristics are stable during the entire test session. Several factors might influence stability, specifically for virtualized testbeds, for example, additional workloads in a virtualized system, load balancing, and movement of virtual machines during the test or simple issues, such as additional heat created by high workloads leading to an emergency CPU performance reduction.

The reference test **MUST** be performed before the benchmarking tests (described in Section 7) start.

6. Reporting

This section describes how the benchmarking test report should be formatted and presented. It is **RECOMMENDED** to include two main sections in the report: the introduction and the detailed test results sections.

6.1. Introduction

The following attributes should be present in the introduction section of the test report.

1. Time and date of the execution of the tests
2. Summary of testbed software and hardware details
   a. DUT/SUT hardware/virtual configuration
      - Make and model of the DUT/SUT, which should be clearly identified
      - Port interfaces, including speed and link information
      - If the DUT/SUT is a Virtual Network Function (VNF)
      - Host (server) hardware and software details
      - Interface acceleration type (such as Data Plane Development Kit (DPDK) and single-root input/output virtualization (SR-IOV))
      - Used CPU cores
      - Used RAM
      - Resource sharing (e.g., pinning details and Non-Uniform Memory Access (NUMA) node) configuration details
      - Hypervisor version
      - Virtual switch version
      - Details of any additional hardware relevant to the DUT/SUT, such as controllers

   b. DUT/SUT software
      - Operating system name
      - Version
      - Specific configuration details (if any)

   c. DUT-/SUT-enabled features
      - Configured DUT/SUT features (see Tables 2 and 3)
      - Attributes of the abovementioned features
      - Any additional relevant information about the features

   d. Test equipment hardware and software
      - Test equipment vendor name
      - Hardware details, including model number and interface type
      - Test equipment firmware and test application software version
      - If the test equipment is a virtual solution
      - The host (server) hardware and software details
      - Interface acceleration type (such as DPDK and SR-IOV)
      - Used CPU cores
      - Used RAM

- Resource sharing (e.g., pinning details and NUMA node) configuration details
- Hypervisor version
- Virtual switch version

e. Key test parameters

- Used cipher suites and keys
- IPv4 and IPv6 traffic distribution
- Number of configured ACLs
- TCP and UDP stack parameter, if tested
- QUIC, HTTP/2, and HTTP/3 parameters, if tested

f. Details of the application traffic mix used in the benchmarking test Throughput Performance with Application Traffic Mix (Section 7.1)

- Name of applications and layer 7 protocols
- Percentage of emulated traffic for each application and layer 7 protocols
- Percentage of encrypted traffic, used cipher suites, and keys (the **RECOMMENDED** ciphers and keys are defined in Section 4.3.1.4)
- Used object sizes for each application and layer 7 protocols

3. Results Summary / Executive Summary

a. Results should be presented with an introduction section documenting the summary of results in a prominent, easy-to-read block.

6.2. Detailed Test Results

In the results section of the test report, the following attributes should be present for each benchmarking test.

a. KPIs **MUST** be documented separately for each benchmarking test. The format of the KPI metrics **MUST** be presented as described in Section 6.3.

b. The next level of details should be graphs showing each of these metrics over the duration (sustain phase) of the test. This allows the user to see the measured performance stability changes over time.

6.3. Benchmarks and Key Performance Indicators

This section lists key performance indicators (KPIs) for overall benchmarking tests. All KPIs **MUST** be measured during the sustain phase of the traffic load profile described in Section 4.3.4. Also, the KPIs **MUST** be measured from the result output of test equipment.

Concurrent TCP Connections

The aggregate number of simultaneous connections between hosts across the DUT/SUT or between hosts and the DUT/SUT (defined in [RFC2647]).

Concurrent QUIC Connections

The aggregate number of simultaneous connections between hosts across the DUT/SUT.

TCP Connections Per Second

The average number of successfully established TCP connections per second between hosts across the DUT/SUT or between hosts and the DUT/SUT. As described in Section 4.3.1.1, the TCP connections are initiated by clients via a TCP three-way handshake (SYN, SYN/ACK, ACK). Then, the TCP session data is sent, and then the TCP sessions are closed via either a TCP three-way close (FIN, FIN/ACK, ACK) or a TCP four-way close (FIN, ACK, FIN, ACK). The TCP sessions **MUST NOT** be closed by RST.

QUIC Connections Per Second

The average number of successfully established QUIC connections per second between hosts across the DUT/SUT. As described in Section 4.3.1.2, the QUIC connections are initiated by clients. Then, the data is sent, and then the QUIC sessions are closed by the "immediate close" method.

Since the QUIC specification defined in Section 4.3.1.2 recommends disabling 0-RTT and early data, this KPI is focused on the 1-RTT handshake. If required, 0-RTT can be also measured in separate test runs while enabling 0-RTT and early data in the test equipment.

Application Transactions Per Second

The average number of successfully completed transactions per second. For a particular transaction to be considered successful, all data **MUST** have been transferred in its entirety. In case of an HTTP(S) transaction, it **MUST** have a valid status code (200 OK).

TLS Handshake Rate

The average number of successfully established TLS connections per second between hosts across the DUT/SUT or between hosts and the DUT/SUT.

For TLS 1.3, the handshake rate can be measured with the 0-RTT or 1-RTT handshake. The transport protocol can be either TCP or QUIC.

Inspected Throughput

The number of bits per second of examined and allowed traffic a network security device is able to transmit to the correct destination interface(s) in response to a specified offered load. The throughput benchmarking tests defined in Section 7 **SHOULD** measure the average layer 2 throughput value when the DUT/SUT is "inspecting" traffic. It is also acceptable to measure other OSI layer throughput. However, the measured layer (e.g., layer 3 throughput) **MUST** be noted in the report, and the user **MUST** be aware of the implication while comparing the throughput performance of multiple DUTs/SUTs measured in different OSI layers. This document recommends presenting the inspected throughput value in Gbit/s rounded to two places of precision with a more specific kbit/s in parenthesis.

Time to First Byte (TTFB)

The elapsed time between the start of sending the TCP SYN packet or QUIC initial Client Hello from the client and the client receiving the first packet of application data from the server via the DUT/SUT. The benchmarking tests HTTP transaction latency (Section 7.4) and HTTPS transaction latency (Section 7.8) measure the minimum, average, and maximum TTFB. The value should be expressed in milliseconds.

URL Response Time / Time to Last Byte (TTLB)

The elapsed time between the start of sending the TCP SYN packet or QUIC initial Client Hello from the client and the client receiving the last packet of application data from the server via the DUT/SUT. The benchmarking tests HTTP transaction latency (Section 7.4) and HTTPS transaction latency (Section 7.8) measure the minimum, average, and maximum TTLB. The value should be expressed in milliseconds.

7. Benchmarking Tests

This section mainly focuses on the benchmarking tests with HTTP/1.1 or HTTP/2 traffic, which uses TCP as the transport protocol. In particular, this section does not define specific benchmarking tests for KPIs related to QUIC or HTTP/3. However, the test methodology defined in the benchmarking tests TCP or QUIC connections per second with HTTPS traffic (Section 7.6), HTTPS transaction latency (Section 7.8), HTTPS throughput (Section 7.7), and concurrent TCP or QUIC connection capacity with HTTPS traffic (Section 7.9) can be used to test KPIs related to QUIC or HTTP/3. The throughput performance test with the application traffic mix defined in Section 7.1 can be performed with any other application traffic, including HTTP/3.

7.1. Throughput Performance with Application Traffic Mix

7.1.1. Objective

Using a relevant application traffic mix, determine the sustainable inspected throughput supported by the DUT/SUT.

Based on the test customer's specific use case, testers can choose the relevant application traffic mix for this test. The details about the traffic mix **MUST** be documented in the report. At least, the following traffic mix details **MUST** be documented and reported together with the test results:

- Name of applications and layer 7 protocols
- Percentage of emulated traffic for each application and layer 7 protocol
- Percentage of encrypted traffic and used cipher suites and keys (the **RECOMMENDED** ciphers and keys are defined in Section 4.3.1.4)
- Used object sizes for each application and layer 7 protocols

7.1.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any benchmarking-test-specific testbed configuration changes **MUST** be documented.

7.1.3. Test Parameters

In this section, the benchmarking-test-specific parameters are defined.

7.1.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented. If the DUT/SUT is configured without TLS inspection, the test report **MUST** explain how this impacts the encrypted traffic of the relevant application traffic mix.

7.1.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be documented for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3
- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- Target inspected throughput: Aggregated line rate of one or more interfaces used in the DUT/SUT or the value defined based on the requirement for a specific deployment scenario
- Initial throughput: 10% of the "Target inspected throughput"

  Note: Initial throughput is not a KPI to report. This value is configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.1.4.
- One of the ciphers and keys defined in Section 4.3.1.4 is **RECOMMENDED** to use for this benchmarking test.

7.1.3.3. Traffic Profile

This test **MUST** be run with a relevant application traffic mix profile.

7.1.3.4. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole sustain phase of the traffic load profile.

a. The number of failed application transactions **MUST** be less than 0.001% (1 out of 100,000 transactions) of the attempted transactions.
b. The number of terminated TCP connections due to unexpected TCP RST sent by the DUT/SUT **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated TCP connections.
c. If HTTP/3 is used, the number of failed QUIC connections due to unexpected HTTP/3 error codes **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated QUIC connections.

7.1.3.5. Measurement

The following KPI metrics **MUST** be reported for this benchmarking test:

- Mandatory KPIs (benchmarks): inspected throughput and application transactions per second

  Note: The TTLB **MUST** be reported along with the object size used in the traffic profile.

- Optional TCP-stack-related KPIs: TCP connections per second, TLS handshake rate, TTFB (minimum, average, and maximum), TTLB (minimum, average, and maximum)
- Optional QUIC-stack-related KPIs: QUIC connections per second and concurrent QUIC connections

7.1.4. Test Procedures and Expected Results

The test procedures are designed to measure the inspected throughput performance of the DUT/SUT at the sustaining period of the traffic load profile. The test procedure consists of three major steps. Step 1 ensures the DUT/SUT is able to reach the performance value (initial throughput) and meets the test results validation criteria when it was very minimally utilized. Step 2 determines whether the DUT/SUT is able to reach the target performance value within the test results validation criteria. Step 3 determines the maximum achievable performance value within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IP types: IPv4 only, IPv6 only, and IPv4 and IPv6 mixed traffic distribution.

7.1.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure the traffic load profile of the test equipment to generate test traffic at the "initial throughput" rate, as described in Section 7.1.3.2. The test equipment **MUST** follow the traffic load profile definition described in Section 4.3.4. The DUT/SUT **MUST** reach the "initial throughput" during the sustain phase. Measure all KPIs, as defined in Section 7.1.3.5. The measured KPIs during the sustain phase **MUST** meet all the test results validation criteria defined in Section 7.1.3.4.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

7.1.4.2. Step 2: Test Run with Target Objective

Configure test equipment to generate traffic at the "Target inspected throughput" rate defined in Section 7.1.3.2. The test equipment **MUST** follow the traffic load profile definition described in Section 4.3.4. The test equipment **MUST** start to measure and record all specified KPIs. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT is expected to reach the desired value of the target objective ("Target inspected throughput") in the sustain phase. Follow Step 3 if the measured value does not meet the target value or does not fulfill the test results validation criteria.

7.1.4.3. Step 3: Test Iteration

Determine the achievable average inspected throughput within the test results validation criteria. The final test iteration **MUST** be performed for the test duration defined in Section 4.3.4.

7.2. TCP Connections Per Second with HTTP Traffic

7.2.1. Objective

Using HTTP traffic, determine the sustainable TCP connection establishment rate supported by the DUT/SUT under different throughput load conditions.

To measure connections per second, test iterations **MUST** use different fixed HTTP response object sizes (the different load conditions) defined in Section 7.2.3.2.

7.2.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any specific testbed configuration changes (number of interfaces, interface type, etc.) **MUST** be documented.

7.2.3. Test Parameters

In this section, benchmarking-test-specific parameters are defined.

7.2.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented.

7.2.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be documented for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3
- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- Target connections per second: Initial value from the product datasheet or the value defined based on the requirement for a specific deployment scenario
- Initial connections per second: 10% of "Target connections per second"

  Note: Initial connections per second is not a KPI to report. This value is configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.2.4.
- The **RECOMMENDED** response object sizes are 1, 2, 4, 16, and 64 KB.

The client **MUST** negotiate HTTP and close the connection with FIN immediately after the completion of one transaction. In each test iteration, the client **MUST** send a GET request requesting a fixed HTTP response object size.

7.2.3.3. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole sustain phase of the traffic load profile.

a. The number of failed application transactions (receiving any HTTP response code other than 200 OK) **MUST** be less than 0.001% (1 out of 100,000 transactions) of the attempted transactions.
b. The number of terminated TCP connections due to unexpected TCP RST sent by the DUT/SUT **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated TCP connections.
c. During the sustain phase, traffic **MUST** be forwarded at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).
d. Concurrent TCP connections **MUST** be constant during steady state, and any deviation of concurrent TCP connections **MUST** be less than 10%. This confirms the DUT opens and closes TCP connections at approximately the same rate.

7.2.3.4. Measurement

TCP connections per second **MUST** be reported for each test iteration (for each object size).

7.2.4. Test Procedures and Expected Results

The test procedure is designed to measure the DUT/SUT's rate of TCP connections per second during the sustaining period of the traffic load profile. The test procedure consists of three major steps. Step 1 ensures the DUT/SUT is able to reach the performance value (Initial connections per second) and meets the test results validation criteria when it was very minimally utilized. Step 2 determines whether the DUT/SUT is able to reach the target performance value within the test results validation criteria. Step 3 determines the maximum achievable performance value within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IP types: IPv4 only, IPv6 only, and IPv4 and IPv6 mixed traffic distribution.

7.2.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure the traffic load profile of the test equipment to establish "Initial connections per second", as defined in Section 7.2.3.2. The traffic load profile **MUST** be defined as described in Section 4.3.4.

The DUT/SUT **MUST** reach the "Initial connections per second" before the sustain phase. The measured KPIs during the sustain phase **MUST** meet all the test results validation criteria defined in Section 7.2.3.3.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** continue to Step 2.

7.2.4.2. Step 2: Test Run with Target Objective

Configure test equipment to establish the target objective ("Target connections per second") defined in Section 7.2.3.2. The test equipment **MUST** follow the traffic load profile definition described in Section 4.3.4.

During the ramp up and sustain phases of each test iteration, other KPIs, such as inspected throughput, concurrent TCP connections, and application transactions per second, **MUST NOT** reach the maximum value the DUT/SUT can support. The test results for specific test iterations **MUST NOT** be reported as valid results if the abovementioned KPI (especially inspected throughput) reaches the maximum value. (For example, if the test iteration with 64 KB of HTTP response object size reached the maximum inspected throughput limitation of the DUT/SUT, the test iteration **MAY** be interrupted and the result for 64 KB must not be reported.)

The test equipment **MUST** start to measure and record all specified KPIs. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT is expected to reach the desired value of the target objective ("Target connections per second") in the sustain phase. Follow Step 3 if the measured value does not meet the target value or does not fulfill the test results validation criteria.

7.2.4.3. Step 3: Test Iteration

Determine the achievable TCP connections per second within the test results validation criteria.

7.3. HTTP Throughput

7.3.1. Objective

Determine the sustainable inspected throughput of the DUT/SUT for HTTP transactions varying the HTTP response object size.

7.3.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any specific testbed configuration changes (number of interfaces, interface type, etc.) **MUST** be documented.

7.3.3. Test Parameters

In this section, benchmarking-test-specific parameters are defined.

7.3.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented.

7.3.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be documented for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3
- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- Target inspected throughput: Aggregated line rate of one or more interfaces used in the DUT/SUT or the value defined based on the requirement for a specific deployment scenario
- Initial throughput: 10% of "Target inspected throughput"

  Note: Initial throughput is not a KPI to report. This value is configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.3.4.
- Number of HTTP response object requests (transactions) per connection: 10
- **RECOMMENDED** HTTP response object size: 1, 16, 64, and 256 KB and mixed objects defined in Table 5

Object size (KB)	Number of requests / Weight
0.2	1
6	1
8	1
9	1
10	1
25	1
26	1
35	1
59	1
347	1

Table 5: Mixed Objects

7.3.3.3. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole sustain phase of the traffic load profile.

a. The number of failed application transactions (receiving any HTTP response code other than 200 OK) **MUST** be less than 0.001% (1 out of 100,000 transactions) of the total attempted transactions.
b. Traffic **MUST** be forwarded at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).
c. Concurrent TCP connections **MUST** be constant during steady state, and any deviation of concurrent TCP connections **MUST** be less than 10%. This confirms the DUT opens and closes TCP connections at approximately the same rate.

7.3.3.4. Measurement

Inspected throughput and HTTP transactions per second **MUST** be reported for each object size.

7.3.4. Test Procedures and Expected Results

The test procedure is designed to measure HTTP throughput of the DUT/ SUT. The test procedure consists of three major steps. Step 1 ensures the DUT/SUT is able to reach the performance value (initial throughput) and meets the test results validation criteria when it was very minimally utilized. Step 2 determines whether the DUT/SUT is able to reach the target performance value within the test results validation criteria. Step 3 determines the maximum achievable performance value within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IPv4 and IPv6 traffic distributions and HTTP response object sizes.

7.3.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure the traffic load profile of the test equipment to establish "initial throughput", as defined in Section 7.3.3.2.

The traffic load profile **MUST** be defined as described in Section 4.3.4. The DUT/SUT **MUST** reach the "initial throughput" during the sustain phase. Measure all KPIs, as defined in Section 7.3.3.4.

The measured KPIs during the sustain phase **MUST** meet the test results validation criteria "a" defined in Section 7.3.3.3. The test results validation criteria "b" and "c" are **OPTIONAL** for Step 1.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

7.3.4.2. Step 2: Test Run with Target Objective

Configure test equipment to establish the target objective ("Target inspected throughput") defined in Section 7.3.3.2. The test equipment **MUST** start to measure and record all specified KPIs. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT is expected to reach the desired value of the target objective in the sustain phase. Follow Step 3 if the measured value does not meet the target value or does not fulfill the test results validation criteria.

7.3.4.3. Step 3: Test Iteration

Determine the achievable inspected throughput within the test results validation criteria and measure the KPI metric transactions per second. The final test iteration **MUST** be performed for the test duration defined in Section 4.3.4.

7.4. HTTP Transaction Latency

7.4.1. Objective

Using HTTP traffic, determine the HTTP transaction latency when the DUT is running with sustainable HTTP transactions per second supported by the DUT/SUT under different HTTP response object sizes.

Test iterations **MUST** be performed with different HTTP response object sizes in two different scenarios: one with a single transaction and the other with multiple transactions within a single TCP connection. For consistency, both the single and multiple transaction tests **MUST** be configured with the same HTTP version.

Scenario 1: The client **MUST** negotiate HTTP and close the connection with FIN immediately after the completion of a single transaction (GET and RESPONSE).

Scenario 2: The client **MUST** negotiate HTTP and close the connection with FIN immediately after the completion of 10 transactions (GET and RESPONSE) within a single TCP connection.

7.4.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any specific testbed configuration changes (number of interfaces, interface type, etc.) **MUST** be documented.

7.4.3. Test Parameters

In this section, benchmarking-test-specific parameters are defined.

7.4.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented.

7.4.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be documented for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3
- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- Target objective for scenario 1: 50% of the connections per second measured in the benchmarking test TCP connections per second with HTTP traffic (Section 7.2)
- Target objective for scenario 2: 50% of the inspected throughput measured in the benchmarking test HTTP throughput (Section 7.3)
- Initial objective for scenario 1: 10% of "Target objective for scenario 1"
- Initial objective for scenario 2: 10% of "Target objective for scenario 2"

  Note: The initial objectives are not KPIs to report. These values are configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.4.4.

- HTTP transaction per TCP connection: Test scenario 1 with a single transaction and test scenario 2 with 10 transactions
- HTTP with GET request requesting a single object: The **RECOMMENDED** object sizes are 1, 16, and 64 KB. For each test iteration, the client **MUST** request a single HTTP response object size.

7.4.3.3. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole sustain phase of the traffic load profile.

a. The number of failed application transactions (receiving any HTTP response code other than 200 OK) **MUST** be less than 0.001% (1 out of 100,000 transactions) of the total attempted transactions.

b. The number of terminated TCP connections due to unexpected TCP RST sent by the DUT/SUT **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated TCP connections.

c. During the sustain phase, traffic **MUST** be forwarded at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).

d. Concurrent TCP connections **MUST** be constant during steady state, and any deviation of concurrent TCP connections **MUST** be less than 10%. This confirms the DUT opens and closes TCP connections at approximately the same rate.

e. After ramp up, the DUT **MUST** achieve the target objectives defined in Section 7.4.3.2 and remain in that state for the entire test duration (sustain phase).

7.4.3.4. Measurement

The TTFB (minimum, average, and maximum) and TTLB (minimum, average, and maximum) **MUST** be reported for each object size.

7.4.4. Test Procedures and Expected Results

The test procedure is designed to measure the TTFB or TTLB when the DUT/SUT is operating close to 50% of its maximum achievable connections per second or inspected throughput. The test procedure consists of two major steps. Step 1 ensures the DUT/SUT is able to reach the initial performance values and meets the test results validation criteria when it was very minimally utilized. Step 2 measures the latency values within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IP types (IPv4 only, IPv6 only, and IPv4 and IPv6 mixed traffic distribution), HTTP response object sizes, and single and multiple transactions per connection scenarios.

7.4.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure the traffic load profile of the test equipment to establish the initial objectives, as defined in Section 7.4.3.2. The traffic load profile **MUST** be defined as described in Section 4.3.4.

The DUT/SUT **MUST** reach the initial objectives before the sustain phase. The measured KPIs during the sustain phase **MUST** meet all the test results validation criteria defined in Section 7.4.3.3.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

7.4.4.2. Step 2: Test Run with Target Objective

Configure test equipment to establish the target objectives defined in Section 7.4.3.2. The test equipment **MUST** follow the traffic load profile definition described in Section 4.3.4.

The test equipment **MUST** start to measure and record all specified KPIs. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT **MUST** reach the desired value of the target objective in the sustain phase.

Measure the minimum, average, and maximum values of the TTFB and TTLB.

7.5. Concurrent TCP Connection Capacity with HTTP Traffic

7.5.1. Objective

Determine the number of concurrent TCP connections that the DUT/SUT sustains when using HTTP traffic.

7.5.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any specific testbed configuration changes (number of interfaces, interface type, etc.) **MUST** be documented.

7.5.3. Test Parameters

In this section, benchmarking-test-specific parameters are defined.

7.5.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented.

7.5.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be noted for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3
- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- Target concurrent connection: Initial value from the product datasheet or the value defined based on the requirement for a specific deployment scenario
- Initial concurrent connection: 10% of "Target concurrent connection"

  Note: Initial concurrent connection is not a KPI to report. This value is configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.5.4.
- Maximum connections per second during ramp up phase: 50% of maximum connections per second measured in the benchmarking test TCP connections per second with HTTP traffic (Section 7.2)
- Ramp up time (in traffic load profile for "Target concurrent connection"): "Target concurrent connection" / "Maximum connections per second during ramp up phase"
- Ramp up time (in traffic load profile for "Initial concurrent connection"): "Initial concurrent connection" / "Maximum connections per second during ramp up phase"

The client **MUST** negotiate HTTP, and each client **MAY** open multiple concurrent TCP connections per server endpoint IP.

Each client sends 10 GET requests requesting 1 KB HTTP response object in the same TCP connection (10 transactions / TCP connections), and the delay (think time) between each transaction **MUST** be X seconds, where X is as follows.

X = ("Ramp up time" + "steady state time") / 10

The established connections **MUST** remain open until the ramp down phase of the test. During the ramp down phase, all connections **MUST** be successfully closed with FIN.

7.5.3.3. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole sustain phase of the traffic load profile.

a. The number of failed application transactions (receiving any HTTP response code other than 200 OK) **MUST** be less than 0.001% (1 out of 100,000 transactions) of the total attempted transactions.
b. The number of terminated TCP connections due to unexpected TCP RST sent by the DUT/SUT **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated TCP connections.
c. During the sustain phase, traffic **MUST** be forwarded at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).

7.5.3.4. Measurement

Average concurrent TCP connections **MUST** be reported for this benchmarking test.

7.5.4. Test Procedures and Expected Results

The test procedure is designed to measure the concurrent TCP connection capacity of the DUT/SUT at the sustaining period of the traffic load profile. The test procedure consists of three major steps. Step 1 ensures the DUT/SUT is able to reach the performance value (Initial concurrent connection) and meets the test results validation criteria when it was very minimally utilized. Step 2 determines whether the DUT/SUT is able to reach the target performance value within the test results validation criteria. Step 3 determines the maximum achievable performance value within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IPv4 and IPv6 traffic distributions.

7.5.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure test equipment to establish "Initial concurrent connections" defined in Section 7.5.3.2. Except ramp up time, the traffic load profile **MUST** be defined as described in Section 4.3.4.

During the sustain phase, the DUT/SUT **MUST** reach the "Initial concurrent connections". The measured KPIs during the sustain phase **MUST** meet all the test results validation criteria defined in Section 7.5.3.3.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

7.5.4.2. Step 2: Test Run with Target Objective

Configure test equipment to establish the target objective ("Target concurrent TCP connections"). The test equipment **MUST** follow the traffic load profile definition (except ramp up time) as described in Section 4.3.4.

During the ramp up and sustain phases, the other KPIs, such as inspected throughput, TCP connections per second, and application transactions per second, **MUST NOT** reach the maximum value the DUT/SUT can support.

The test equipment **MUST** start to measure and record KPIs defined in Section 7.5.3.4. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT is expected to reach the desired value of the target objective in the sustain phase. Follow Step 3 if the measured value does not meet the target value or does not fulfill the test results validation criteria.

7.5.4.3. Step 3: Test Iteration

Determine the achievable concurrent TCP connections capacity within the test results validation criteria.

7.6. TCP or QUIC Connections per Second with HTTPS Traffic

7.6.1. Objective

Using HTTPS traffic, determine the sustainable TLS session establishment rate supported by the DUT/SUT under different throughput load conditions.

Test iterations **MUST** include common cipher suites and key strengths, as well as forward-looking stronger keys. Specific test iterations **MUST** include ciphers and keys defined in Section 7.6.3.2.

For each cipher suite and key strength, test iterations **MUST** use a single HTTPS response object size defined in Section 7.6.3.2 to measure connections per second performance under a variety of DUT/SUT security inspection load conditions.

7.6.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any specific testbed configuration changes (number of interfaces, interface type, etc.) **MUST** be documented.

7.6.3. Test Parameters

In this section, benchmarking-test-specific parameters are defined.

7.6.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented.

7.6.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be documented for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3
- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- Target connections per second: Initial value from the product datasheet or the value defined based on the requirement for a specific deployment scenario
- Initial connections per second: 10% of "Target connections per second"

  Note: Initial connections per second is not a KPI to report. This value is configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.6.4.)
- **RECOMMENDED** ciphers and keys defined in Section 4.3.1.4
- The **RECOMMENDED** object sizes are 1, 2, 4, 16, and 64 KB.

The client **MUST** negotiate HTTPS and close the connection without error immediately after the completion of one transaction. In each test iteration, the client **MUST** send a GET request requesting a fixed HTTPS response object size.

7.6.3.3. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole test duration.

a. The number of failed application transactions (receiving any HTTP response code other than 200 OK) **MUST** be less than 0.001% (1 out of 100,000 transactions) of the attempted transactions.
b. The number of terminated TCP connections due to unexpected TCP RST sent by the DUT/SUT **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated TCP connections. If HTTP/3 is used, the number of terminated QUIC connections due to unexpected errors **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated QUIC connections.
c. During the sustain phase, traffic **MUST** be forwarded at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).
d. The concurrent TCP connections generation rate **MUST** be constant during steady state, and any deviation of concurrent TCP connections **MUST** be less than 10%. If HTTP/3 is used, the concurrent QUIC connections generation rate **MUST** be constant during steady state, and any deviation of concurrent QUIC connections **MUST** be less than 10%. This confirms the DUT opens and closes connections at approximately the same rate.

7.6.3.4. Measurement

If HTTP 1.1 or HTTP/2 is used, TCP connections per second **MUST** be reported for each test iteration (for each object size).

If HTTP/3 is used, QUIC connections per second **MUST** be measured and reported for each test iteration (for each object size).

The KPI metric TLS handshake rate can be measured in the test using 1 KB object size.

7.6.4. Test Procedures and Expected Results

The test procedure is designed to measure the DUT/SUT's rate of TCP or QUIC connections per second during the sustaining period of the traffic load profile. The test procedure consists of three major steps. Step 1 ensures the DUT/SUT is able to reach the performance value (Initial connections per second) and meets the test results validation criteria when it was very minimally utilized. Step 2 determines whether the DUT/SUT is able to reach the target performance value within the test results validation criteria. Step 3 determines the maximum achievable performance value within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IPv4 and IPv6 traffic distributions.

7.6.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure the traffic load profile of the test equipment to establish "Initial connections per second", as defined in Section 7.6.3.2. The traffic load profile **MUST** be defined as described in Section 4.3.4.

The DUT/SUT **MUST** reach the "Initial connections per second" before the sustain phase. The measured KPIs during the sustain phase **MUST** meet all the test results validation criteria defined in Section 7.6.3.3.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

7.6.4.2. Step 2: Test Run with Target Objective

Configure test equipment to establish "Target connections per second", as defined in Section 7.6.3.2. The test equipment **MUST** follow the traffic load profile definition described in Section 4.3.4.

During the ramp up and sustain phases, other KPIs, such as inspected throughput, concurrent TCP or QUIC connections, and application transactions per second, **MUST NOT** reach the maximum value the DUT/SUT can support. The test results for the specific test iteration **MUST NOT** be reported as valid results if the abovementioned KPI (especially inspected throughput) reaches the maximum value. (For example, if the test iteration with 64 KB of HTTPS response object size reached the maximum inspected throughput limitation of the DUT, the test iteration **MAY** be interrupted, and the result for 64 KB should not be reported).

The test equipment **MUST** start to measure and record all specified KPIs. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT is expected to reach the desired value of the target objective ("Target connections per second") in the sustain phase. Follow Step 3 if the measured value does not meet the target value or does not fulfill the test results validation criteria.

7.6.4.3. Step 3: Test Iteration

Determine the achievable connections per second within the test results validation criteria.

7.7. HTTPS Throughput

7.7.1. Objective

Determine the sustainable inspected throughput of the DUT/SUT for HTTPS transactions by varying the HTTPS response object size.

Test iterations **MUST** include common cipher suites and key strengths, as well as forward-looking stronger keys. Specific test iterations **MUST** include the ciphers and keys defined in Section 7.7.3.2.

7.7.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any specific testbed configuration changes (number of interfaces, interface type, etc.) **MUST** be documented.

7.7.3. Test Parameters

In this section, benchmarking-test-specific parameters are defined.

7.7.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented.

7.7.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be documented for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3
- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- Target inspected throughput: Aggregated line rate of one or more interfaces used in the DUT/SUT or the value defined based on the requirement for a specific deployment scenario
- Initial throughput: 10% of "Target inspected throughput"

  Note: Initial throughput is not a KPI to report. This value is configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.7.4.
- Number of HTTPS response object requests (transactions) per connection: 10

- **RECOMMENDED** ciphers and keys defined in Section 4.3.1.4
- **RECOMMENDED** HTTPS response object size: 1, 16, 64, and 256 KB and mixed objects defined in Table 5 of Section 7.3.3.2

7.7.3.3. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole sustain phase of the traffic load profile.

a. The number of failed application transactions (receiving any HTTP response code other than 200 OK) **MUST** be less than 0.001% (1 out of 100,000 transactions) of the attempted transactions.
b. Traffic **MUST** be generated at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).
c. The concurrent generated TCP connections **MUST** be constant during steady state, and any deviation of concurrent TCP connections **MUST** be less than 10%. If HTTP/3 is used, the concurrent generated QUIC connections **MUST** be constant during steady state, and any deviation of concurrent QUIC connections **MUST** be less than 10%. This confirms the DUT opens and closes connections at approximately the same rate.

7.7.3.4. Measurement

Inspected throughput and HTTPS transactions per second **MUST** be reported for each object size.

7.7.4. Test Procedures and Expected Results

The test procedure consists of three major steps. Step 1 ensures the DUT/SUT is able to reach the performance value (initial throughput) and meets the test results validation criteria when it was very minimally utilized. Step 2 determines whether the DUT/SUT is able to reach the target performance value within the test results validation criteria. Step 3 determines the maximum achievable performance value within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IPv4 and IPv6 traffic distributions and HTTPS response object sizes.

7.7.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure the traffic load profile of the test equipment to establish "initial throughput", as defined in Section 7.7.3.2.

The traffic load profile **MUST** be defined as described in Section 4.3.4. The DUT/SUT **MUST** reach the "initial throughput" during the sustain phase. Measure all KPIs, as defined in Section 7.7.3.4.

The measured KPIs during the sustain phase **MUST** meet the test results validation criteria "a" defined in Section 7.7.3.3. The test results validation criteria "b" and "c" are **OPTIONAL** for Step 1.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

7.7.4.2. Step 2: Test Run with Target Objective

Configure test equipment to establish the target objective ("Target inspected throughput") defined in Section 7.7.3.2. The test equipment **MUST** start to measure and record all specified KPIs. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT is expected to reach the desired value of the target objective in the sustain phase. Follow Step 3 if the measured value does not meet the target value or does not fulfill the test results validation criteria.

7.7.4.3. Step 3: Test Iteration

Determine the achievable average inspected throughput within the test results validation criteria. The final test iteration **MUST** be performed for the test duration defined in Section 4.3.4.

7.8. HTTPS Transaction Latency

7.8.1. Objective

Using HTTPS traffic, determine the HTTPS transaction latency when the DUT/SUT is running with sustainable HTTPS transactions per second supported by the DUT/SUT under different HTTPS response object sizes.

Scenario 1: The client **MUST** negotiate HTTPS and close the connection immediately after the completion of a single transaction (GET and RESPONSE).

Scenario 2: The client **MUST** negotiate HTTPS and close the connection immediately after the completion of 10 transactions (GET and RESPONSE) within a single TCP or QUIC connection.

7.8.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any specific testbed configuration changes (number of interfaces, interface type, etc.) **MUST** be documented.

7.8.3. Test Parameters

In this section, benchmarking-test-specific parameters are defined.

7.8.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented.

7.8.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be documented for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3

- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- **RECOMMENDED** cipher suites and key sizes defined in Section 4.3.1.4
- Target objective for scenario 1: 50% of the connections per second measured in the benchmarking test TCP or QUIC connections per second with HTTPS traffic (Section 7.6)
- Target objective for scenario 2: 50% of the inspected throughput measured in the benchmarking test HTTPS throughput (Section 7.7)
- Initial objective for scenario 1: 10% of "Target objective for scenario 1"
- Initial objective for scenario 2: 10% of "Target objective for scenario 2"

  Note: The initial objectives are not KPIs to report. These values are configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.8.4.

- HTTPS transaction per TCP or QUIC connection: Test scenario 1 with a single transaction and scenario 2 with 10 transactions
- HTTPS with GET request requesting a single object: The **RECOMMENDED** object sizes are 1, 16, and 64 KB. For each test iteration, the client **MUST** request a single HTTPS response object size.

7.8.3.3. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole sustain phase of the traffic load profile.

a. The number of failed application transactions (receiving any HTTP response code other than 200 OK) **MUST** be less than 0.001% (1 out of 100,000 transactions) of the total attempted transactions.

b. The number of terminated TCP connections due to unexpected TCP RST sent by the DUT/SUT **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated TCP connections. If HTTP/3 is used, the number of terminated QUIC connections due to unexpected errors **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated QUIC connections.

c. During the sustain phase, traffic **MUST** be forwarded at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).

d. Concurrent TCP or QUIC connections **MUST** be constant during steady state, and any deviation of concurrent TCP connections **MUST** be less than 10%. If HTTP/3 is used, the concurrent generated QUIC connections **MUST** be constant during steady state, and any deviation of concurrent QUIC connections **MUST** be less than 10%. This confirms the DUT opens and closes connections at approximately the same rate.

e. After ramp up, the DUT/SUT **MUST** achieve the target objectives defined in the parameters in Section 7.8.3.2 and remain in that state for the entire test duration (sustain phase).

7.8.3.4. Measurement

The TTFB (minimum, average, and maximum) and TTLB (minimum, average, and maximum) **MUST** be reported for each object size.

7.8.4. Test Procedures and Expected Results

The test procedure is designed to measure the TTFB or TTLB when the DUT/SUT is operating close to 50% of its maximum achievable connections per second or inspected throughput. The test procedure consists of two major steps. Step 1 ensures the DUT/SUT is able to reach the initial performance values and meets the test results validation criteria when it is very minimally utilized. Step 2 measures the latency values within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IP types (IPv4 only, IPv6 only, and IPv4 and IPv6 mixed traffic distribution), HTTPS response object sizes, and single and multiple transactions per connection scenarios.

7.8.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure the traffic load profile of the test equipment to establish the initial objectives, as defined in Section 7.8.3.2. The traffic load profile **MUST** be defined as described in Section 4.3.4.

The DUT/SUT **MUST** reach the initial objectives before the sustain phase. The measured KPIs during the sustain phase **MUST** meet all the test results validation criteria defined in Section 7.8.3.3.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

7.8.4.2. Step 2: Test Run with Target Objective

Configure test equipment to establish the target objectives defined in Section 7.8.3.2. The test equipment **MUST** follow the traffic load profile definition described in Section 4.3.4.

The test equipment **MUST** start to measure and record all specified KPIs. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT **MUST** reach the desired value of the target objective in the sustain phase.

Measure the minimum, average, and maximum values of the TTFB and TTLB.

7.9. Concurrent TCP or QUIC Connection Capacity with HTTPS Traffic

7.9.1. Objective

Determine the number of concurrent TCP or QUIC connections the DUT/SUT sustains when using HTTPS traffic.

7.9.2. Test Setup

The testbed setup **MUST** be configured as defined in Section 4. Any specific testbed configuration changes (number of interfaces, interface type, etc.) **MUST** be documented.

7.9.3. Test Parameters

In this section, benchmarking-test-specific parameters are defined.

7.9.3.1. DUT/SUT Configuration Parameters

DUT/SUT parameters **MUST** conform to the requirements defined in Section 4.2. Any configuration changes for this specific benchmarking test **MUST** be documented.

7.9.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The following parameters **MUST** be documented for this benchmarking test:

- Client IP address ranges defined in Section 4.3.1.3
- Server IP address ranges defined in Section 4.3.2.3
- Traffic distribution ratio between IPv4 and IPv6 defined in Section 4.3.1.3
- **RECOMMENDED** cipher suites and key sizes defined in Section 4.3.1.4
- Target concurrent connections: Initial value from the product datasheet or the value defined based on the requirement for a specific deployment scenario
- Initial concurrent connections: 10% of "Target concurrent connections"

  Note: Initial concurrent connections is not a KPI to report. This value is configured on the traffic generator and used to perform Step 1 (Test Initialization and Qualification) described in Section 7.9.4.

- Connections per second during ramp up phase: 50% of maximum connections per second measured in the benchmarking test TCP or QUIC connections per second with HTTPS traffic (Section 7.6)
- Ramp up time (in traffic load profile for "Target concurrent connections"): "Target concurrent connections" / "Maximum connections per second during ramp up phase"
- Ramp up time (in traffic load profile for "Initial concurrent connections"): "Initial concurrent connections" / "Maximum connections per second during ramp up phase"

The client **MUST** perform HTTPS transactions with persistence, and each client can open multiple concurrent connections per server endpoint IP.

Each client sends 10 GET requests requesting 1 KB HTTPS response objects in the same TCP or QUIC connections (10 transactions/connections), and the delay (think time) between each transaction **MUST** be X seconds, where X is as follows.

X = ("Ramp up time" + "steady state time") / 10

The established connections **MUST** remain open until the ramp down phase of the test. During the ramp down phase, all connections **MUST** be successfully closed with FIN.

7.9.3.3. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole sustain phase of the traffic load profile.

a. The number of failed application transactions (receiving any HTTP response code other than 200 OK) **MUST** be less than 0.001% (1 out of 100,000 transactions) of the total attempted transactions.
b. The number of terminated TCP connections due to unexpected TCP RSTs sent by the DUT/SUT **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated TCP connections. If HTTP/3 is used, the number of terminated QUIC connections due to unexpected errors **MUST** be less than 0.001% (1 out of 100,000 connections) of the total initiated QUIC connections.
c. During the sustain phase, traffic **MUST** be forwarded at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).

7.9.3.4. Measurement

Average concurrent TCP or QUIC connections **MUST** be reported for this benchmarking test.

7.9.4. Test Procedures and Expected Results

The test procedure is designed to measure the concurrent TCP connection capacity of the DUT/SUT at the sustaining period of the traffic load profile. The test procedure consists of three major steps. Step 1 ensures the DUT/SUT is able to reach the performance value (Initial concurrent connection) and meets the test results validation criteria when it was very minimally utilized. Step 2 determines whether the DUT/SUT is able to reach the target performance value within the test results validation criteria. Step 3 determines the maximum achievable performance value within the test results validation criteria.

This test procedure **MAY** be repeated multiple times with different IPv4 and IPv6 traffic distributions.

7.9.4.1. Step 1: Test Initialization and Qualification

Verify the link status of all connected physical interfaces. All interfaces are expected to be in "UP" status.

Configure test equipment to establish "Initial concurrent connections" defined in Section 7.9.3.2. Except ramp up time, the traffic load profile **MUST** be defined as described in Section 4.3.4.

During the sustain phase, the DUT/SUT **MUST** reach the "Initial concurrent connections". The measured KPIs during the sustain phase **MUST** meet the test results validation criteria "a" and "b" defined in Section 7.9.3.3.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

7.9.4.2. Step 2: Test Run with Target Objective

Configure test equipment to establish the target objective ("Target concurrent connections"). The test equipment **MUST** follow the traffic load profile definition (except ramp up time) described in Section 4.3.4.

During the ramp up and sustain phases, the other KPIs, such as inspected throughput, TCP or QUIC connections per second, and application transactions per second, **MUST NOT** reach the maximum value that the DUT/SUT can support.

The test equipment **MUST** start to measure and record KPIs defined in Section 7.9.3.4. Continue the test until all traffic profile phases are completed.

Within the test results validation criteria, the DUT/SUT is expected to reach the desired value of the target objective in the sustain phase. Follow Step 3 if the measured value does not meet the target value or does not fulfill the test results validation criteria.

7.9.4.3. Step 3: Test Iteration

Determine the achievable concurrent TCP or QUIC connections within the test results validation criteria.

8. IANA Considerations

This document makes no specific request of IANA.

IANA has assigned IPv4 and IPv6 address blocks in [RFC6890] that have been registered for special purposes. The IPv6 address block 2001:2::/48 has been allocated for the purpose of IPv6 benchmarking [RFC5180], and the IPv4 address block 198.18.0.0/15 has been allocated for the purpose of IPv4 benchmarking [RFC2544]. This assignment was made to minimize the chance of conflict in case a testing device were to be accidentally connected to the part of the Internet.

9. Security Considerations

The primary goal of this document is to provide benchmarking terminology and methodology for next-generation network security devices for use in a laboratory-isolated test environment. However, readers should be aware that there is some overlap between performance and security issues. Specifically, the optimal configuration for network security device performance may not be the most secure, and vice versa. Testing security platforms with working exploits and malware carries risks. Ensure proper access controls are implemented to prevent unintended exposure to vulnerable networks or systems. The cipher suites recommended in this document are for test purposes only. The cipher suite recommendation for a real deployment is outside the scope of this document.

Security assessment of an NGFW/NGIPS product could also include an analysis whether any type of uncommon traffic characteristics would have a significant impact on performance. Such performance impacts would allow an attacker to use such specifically crafted traffic as a DoS attack to reduce the remaining performance available to other traffic through the NGFW/NGIPS. Such uncommon traffic characteristics might include, for example, IP-fragmented traffic, a specific type of application traffic, or uncommonly high HTTP transaction rate traffic.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

10.2. Informative References

[CVE] CVE, "Current CVSS Score Distribution For All Vulnerabilities", <https://www.cvedetails.com/>.

[fastly] Oku, K. and J. Iyengar, "QUIC vs TCP: Which is Better?", April 2020, <https://www.fastly.com/blog/measuring-quic-vs-tcp-computational-efficiency>.

[RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, DOI 10.17487/RFC2544, March 1999, <https://www.rfc-editor.org/info/rfc2544>.

[RFC2647] Newman, D., "Benchmarking Terminology for Firewall Performance", RFC 2647, DOI 10.17487/RFC2647, August 1999, <https://www.rfc-editor.org/info/rfc2647>.

[RFC3511] Hickman, B., Newman, D., Tadjudin, S., and T. Martin, "Benchmarking Methodology for Firewall Performance", RFC 3511, DOI 10.17487/RFC3511, April 2003, <https://www.rfc-editor.org/info/rfc3511>.

[RFC5180] Popoviciu, C., Hamza, A., Van de Velde, G., and D. Dugatkin, "IPv6 Benchmarking Methodology for Network Interconnect Devices", RFC 5180, DOI 10.17487/RFC5180, May 2008, <https://www.rfc-editor.org/info/rfc5180>.

[RFC6815] Bradner, S., Dubray, K., McQuaid, J., and A. Morton, "Applicability Statement for RFC 2544: Use on Production Networks Considered Harmful", RFC 6815, DOI 10.17487/RFC6815, November 2012, <https://www.rfc-editor.org/info/rfc6815>.

[RFC6890] Cotton, M., Vegoda, L., Bonica, R., Ed., and B. Haberman, "Special-Purpose IP Address Registries", BCP 153, RFC 6890, DOI 10.17487/RFC6890, April 2013, <https://www.rfc-editor.org/info/rfc6890>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

[RFC9001] Thomson, M., Ed. and S. Turner, Ed., "Using TLS to Secure QUIC", RFC 9001, DOI 10.17487/RFC9001, May 2021, <https://www.rfc-editor.org/info/rfc9001>.

[RFC9002] Iyengar, J., Ed. and I. Swett, Ed., "QUIC Loss Detection and Congestion Control", RFC 9002, DOI 10.17487/RFC9002, May 2021, <https://www.rfc-editor.org/info/rfc9002>.

[RFC9113] Thomson, M., Ed. and C. Benfield, Ed., "HTTP/2", RFC 9113, DOI 10.17487/RFC9113, June 2022, <https://www.rfc-editor.org/info/rfc9113>.

[RFC9114] Bishop, M., Ed., "HTTP/3", RFC 9114, DOI 10.17487/RFC9114, June 2022, <https://www.rfc-editor.org/info/rfc9114>.

[RFC9204] Krasic, C., Bishop, M., and A. Frindell, Ed., "QPACK: Field Compression for HTTP/3", RFC 9204, DOI 10.17487/RFC9204, June 2022, <https://www.rfc-editor.org/info/rfc9204>.

[RFC9293] Eddy, W., Ed., "Transmission Control Protocol (TCP)", STD 7, RFC 9293, DOI 10.17487/RFC9293, August 2022, <https://www.rfc-editor.org/info/rfc9293>.

[Undertow] undertow, "An in depth overview of HTTP/2", <https://undertow.io/blog/2015/04/27/An-in-depth-overview-of-HTTP2.html>.

[Wiki-NGFW] Wikipedia, "Next-generation firewall", January 2023, <https://en.wikipedia.org/w/index.php?title=Next-generation_firewall&oldid=1133673904>.

Appendix A. Test Methodology - Security Effectiveness Evaluation

A.1. Test Objective

This test methodology verifies the DUT/SUT is able to detect, prevent, and report the vulnerabilities.

In this test, background test traffic will be generated to utilize the DUT/SUT. In parallel, some malicious traffic will be sent to the DUT/SUT as encrypted and cleartext payload formats using a traffic generator. Section 4.2.1 defines the selection of the malicious traffic from the Common Vulnerabilities and Exposures (CVEs) list for testing.

The following KPIs are measured in this test:

- Number of blocked CVEs
- Number of bypassed (non-blocked) CVEs
- Background traffic performance (verify if the background traffic is impacted while sending CVEs toward the DUT/SUT)
- Accuracy of DUT/SUT statistics in terms of vulnerabilities reporting

A.2. Testbed Setup

The same testbed **MUST** be used for security effectiveness tests and for benchmarking test cases defined in Section 7.

A.3. Test Parameters

In this section, the benchmarking-test-specific parameters are defined.

A.3.1. DUT/SUT Configuration Parameters

DUT/SUT configuration parameters **MUST** conform to the requirements defined in Section 4.2. The same DUT configuration **MUST** be used for the security effectiveness test and for benchmarking test cases defined in Section 7. The DUT/SUT **MUST** be configured in "Inline" mode, all detected attack traffic **MUST** be dropped, and the session **MUST** be reset

A.3.2. Test Equipment Configuration Parameters

Test equipment configuration parameters **MUST** conform to the requirements defined in Section 4.3. The same client and server IP ranges **MUST** be configured as used in the benchmarking test cases. In addition, the following parameters **MUST** be documented for this benchmarking test:

- Background Traffic: 45% of maximum HTTP throughput and 45% of maximum HTTPS throughput supported by the DUT/SUT (measured with object size 64 KB in the benchmarking tests HTTP(S) Throughput defined in Sections 7.3 and 7.7)
- **RECOMMENDED** CVE traffic transmission Rate: 10 CVEs per second
- It is **RECOMMENDED** to generate each CVE multiple times (sequentially) at 10 CVEs per second.
- Ciphers and keys for the encrypted CVE traffic **MUST** use the same cipher configured for HTTPS-traffic-related benchmarking tests (Sections 7.6-7.9)

A.4. Test Results Validation Criteria

The following criteria are the test results validation criteria. The test results validation criteria **MUST** be monitored during the whole test duration.

a. The number of failed application transactions in the background traffic **MUST** be less than 0.01% of the attempted transactions.
b. The number of terminated TCP or QUIC connections of the background traffic (due to unexpected errors) **MUST** be less than 0.01% of the total initiated TCP connections in the background traffic.
c. During the sustain phase, traffic **MUST** be forwarded at a constant rate (it is considered as a constant rate if any deviation of the traffic forwarding rate is less than 5%).
d. A false positive **MUST NOT** occur in the background traffic.

A.5. Measurement

The following KPI metrics **MUST** be reported for this test scenario:

Mandatory KPIs:

- Blocked CVEs: They **MUST** be represented in the following ways:
  - Number of blocked CVEs out of total CVEs
  - Percentage of blocked CVEs
- Unblocked CVEs: They **MUST** be represented in the following ways:
  - Number of unblocked CVEs out of total CVEs
  - Percentage of unblocked CVEs
- Background traffic behavior: It **MUST** be represented in one of the followings ways:
  - No impact: Considered as "no impact" if any deviation of the traffic forwarding rate is less than or equal to 5% (constant rate)
  - Minor impact: Considered as "minor impact" if any deviation of the traffic forwarding rate is greater than 5% and less than or equal to 10% (i.e., small spikes)
  - Heavy impact: Considered as "heavy impact" if any deviation of the traffic forwarding rate is greater than 10% (i.e., large spikes) or reduced the background HTTP(S) throughput greater than 10%
- DUT/SUT reporting accuracy: The DUT/SUT **MUST** report all detected vulnerabilities.

Optional KPIs:

- List of unblocked CVEs

A.6. Test Procedures and Expected Results

The test procedure is designed to measure the security effectiveness of the DUT/SUT at the sustaining period of the traffic load profile. The test procedure consists of two major steps. This test procedure **MAY** be repeated multiple times with different IPv4 and IPv6 traffic distributions.

A.6.1. Step 1: Background Traffic

Generate background traffic at the transmission rate defined in Appendix A.3.2.

The DUT/SUT **MUST** reach the target objective (HTTP(S) throughput) in the sustain phase. The measured KPIs during the sustain phase **MUST** meet all the test results validation criteria defined in Appendix A.4.

If the KPI metrics do not meet the test results validation criteria, the test procedure **MUST NOT** be continued to Step 2.

A.6.2. Step 2: CVE Emulation

While generating background traffic (in the sustain phase), send the CVE traffic, as defined in the parameter section (Appendix A.3.2).

The test equipment **MUST** start to measure and record all specified KPIs. Continue the test until all CVEs are sent.

The measured KPIs **MUST** meet all the test results validation criteria defined in Appendix A.4.

In addition, the DUT/SUT should report the detected vulnerabilities in the log correctly, or there **MUST** be reference material available that will allow for verification that the correct vulnerability was detected if, for example, a different naming convention is used. This reference material **MUST** be cited in the report.

Appendix B. DUT/SUT Classification

This document aims to classify the DUT/SUT into four different categories based on its maximum-supported firewall throughput performance number defined in the vendor datasheet. This classification **MAY** help users to determine specific configuration scales (e.g., number of ACL entries), traffic profiles, and attack traffic profiles, scaling those proportionally to the DUT/SUT sizing category.

The four different categories are Extra Small (XS), Small (S), Medium (M), and Large (L). The **RECOMMENDED** throughput values for the following categories are:

Extra Small (XS) - Supported throughput less than or equal to 1 Gbit/s

Small (S) - Supported throughput greater than 1 Gbit/s and less than or equal to 5Gbit/s

Medium (M) - Supported throughput greater than 5 Gbit/s and less than or equal to 10Gbit/s

Large (L) - Supported throughput greater than 10 Gbit/s

Acknowledgements

The authors wish to acknowledge the members of NetSecOPEN for their participation in the creation of this document. Additionally, the following members need to be acknowledged:

Anand Vijayan, Chris Marshall, Jay Lindenauer, Michael Shannon, Mike Deichman, Ryan Riese, and Toulnay Orkun.

Contributors

The following individuals contributed significantly to the creation of this document:

Alex Samonte, Amritam Putatunda, Aria Eslambolchizadeh, Chao Guo, Chris Brown, Cory Ford, David DeSanto, Jurrie Van Den Breekel, Michelle Rhines, Mike Jack, Ryan Liles, Samaresh Nair, Stephen Goudreault, Tim Carlin, and Tim Otto.

Authors' Addresses

Balamuhunthan Balarajah
Berlin
Germany
Email: bm.balarajah@gmail.com

Carsten Rossenhoevel
EANTC AG
Salzufer 14
10587 Berlin
Germany
Email: cross@eantc.de

Brian Monkman
NetSecOPEN
417 Independence Court
Mechanicsburg, PA 17050
United States of America
Email: bmonkman@netsecopen.org