

Benchmarking Working Group  
Internet-Draft  
Intended status: Informational  
Expires: January 29, 2010

M. Kaeo  
Double Shot Security  
T. Van Herck  
Cisco Systems  
July 28, 2009

**Methodology for Benchmarking IPsec Devices**  
**draft-ietf-bmwg-ipsec-meth-05**

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#). This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on January 29, 2010.

Copyright Notice

Copyright (c) 2009 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents in effect on the date of publication of this document (<http://trustee.ietf.org/license-info>). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

#### Abstract

The purpose of this draft is to describe methodology specific to the benchmarking of IPsec IP forwarding devices. It builds upon the tenets set forth in [[RFC2544](#)], [[RFC2432](#)] and other IETF Benchmarking Methodology Working Group (BMWG) efforts. This document seeks to extend these efforts to the IPsec paradigm.

The BMWG produces two major classes of documents: Benchmarking Terminology documents and Benchmarking Methodology documents. The Terminology documents present the benchmarks and other related terms. The Methodology documents define the procedures required to collect the benchmarks cited in the corresponding Terminology documents.



Table of Contents

- [1. Introduction . . . . .](#) [5](#)
- [2. Document Scope . . . . .](#) [5](#)
- [3. Methodology Format . . . . .](#) [5](#)
- [4. Key Words to Reflect Requirements . . . . .](#) [6](#)
- [5. Test Considerations . . . . .](#) [6](#)
- [6. Test Topologies . . . . .](#) [6](#)
- [7. Test Parameters . . . . .](#) [9](#)
  - [7.1. Frame Type . . . . .](#) [9](#)
    - [7.1.1. IP . . . . .](#) [9](#)
    - [7.1.2. UDP . . . . .](#) [9](#)
    - [7.1.3. TCP . . . . .](#) [9](#)
    - [7.1.4. NAT-Traversal . . . . .](#) [9](#)
  - [7.2. Frame Sizes . . . . .](#) [10](#)
  - [7.3. Fragmentation and Reassembly . . . . .](#) [10](#)
  - [7.4. Time To Live . . . . .](#) [11](#)
  - [7.5. Trial Duration . . . . .](#) [11](#)
  - [7.6. Security Context Parameters . . . . .](#) [11](#)
    - [7.6.1. IPsec Transform Sets . . . . .](#) [11](#)
    - [7.6.2. IPsec Topologies . . . . .](#) [13](#)
    - [7.6.3. IKE Keepalives . . . . .](#) [14](#)
    - [7.6.4. IKE DH-group . . . . .](#) [14](#)
    - [7.6.5. IKE SA / IPsec SA Lifetime . . . . .](#) [14](#)
    - [7.6.6. IPsec Selectors . . . . .](#) [15](#)
    - [7.6.7. NAT-Traversal . . . . .](#) [15](#)
- [8. Capacity . . . . .](#) [15](#)
  - [8.1. IPsec Tunnel Capacity . . . . .](#) [15](#)
  - [8.2. IPsec SA Capacity . . . . .](#) [16](#)
- [9. Throughput . . . . .](#) [17](#)
  - [9.1. Throughput baseline . . . . .](#) [17](#)
  - [9.2. IPsec Throughput . . . . .](#) [18](#)
  - [9.3. IPsec Encryption Throughput . . . . .](#) [19](#)
  - [9.4. IPsec Decryption Throughput . . . . .](#) [20](#)
- [10. Latency . . . . .](#) [20](#)
  - [10.1. Latency Baseline . . . . .](#) [21](#)
  - [10.2. IPsec Latency . . . . .](#) [22](#)
  - [10.3. IPsec Encryption Latency . . . . .](#) [23](#)
  - [10.4. IPsec Decryption Latency . . . . .](#) [24](#)
  - [10.5. Time To First Packet . . . . .](#) [24](#)
- [11. Frame Loss Rate . . . . .](#) [25](#)
  - [11.1. Frame Loss Baseline . . . . .](#) [25](#)
  - [11.2. IPsec Frame Loss . . . . .](#) [26](#)
  - [11.3. IPsec Encryption Frame Loss . . . . .](#) [27](#)
  - [11.4. IPsec Decryption Frame Loss . . . . .](#) [28](#)
  - [11.5. IKE Phase 2 Rekey Frame Loss . . . . .](#) [28](#)
- [12. IPsec Tunnel Setup Behavior . . . . .](#) [29](#)
  - [12.1. IPsec Tunnel Setup Rate . . . . .](#) [29](#)



- [12.2. IKE Phase 1 Setup Rate . . . . .](#) [30](#)
- [12.3. IKE Phase 2 Setup Rate . . . . .](#) [31](#)
- [13. IPsec Rekey Behavior . . . . .](#) [32](#)
- [13.1. IKE Phase 1 Rekey Rate . . . . .](#) [32](#)
- [13.2. IKE Phase 2 Rekey Rate . . . . .](#) [33](#)
- [14. IPsec Tunnel Failover Time . . . . .](#) [34](#)
- [15. DoS Attack Resiliency . . . . .](#) [36](#)
- [15.1. Phase 1 DoS Resiliency Rate . . . . .](#) [36](#)
- [15.2. Phase 2 Hash Mismatch DoS Resiliency Rate . . . . .](#) [37](#)
- [15.3. Phase 2 Anti Replay Attack DoS Resiliency Rate . . . . .](#) [37](#)
- [16. Security Considerations . . . . .](#) [39](#)
- [17. Acknowledgements . . . . .](#) [39](#)
- [18. References . . . . .](#) [39](#)
- [18.1. Normative References . . . . .](#) [39](#)
- [18.2. Informative References . . . . .](#) [41](#)
- [Authors' Addresses . . . . .](#) [41](#)



## **1. Introduction**

This document defines a specific set of tests that can be used to measure and report the performance characteristics of IPsec devices. It extends the methodology already defined for benchmarking network interconnecting devices in [[RFC2544](#)] to IPsec gateways and additionally introduces tests which can be used to measure end-host IPsec performance.

## **2. Document Scope**

The primary focus of this document is to establish a performance testing methodology for IPsec devices that support manual keying and IKEv1. A separate document will be written specifically to address testing using the updated IKEv2 specification. Both IPv4 and IPv6 addressing will be taken into consideration for all relevant test methodologies.

The testing will be constrained to:

- o Devices acting as IPsec gateways whose tests will pertain to both IPsec tunnel and transport mode.
- o Devices acting as IPsec end-hosts whose tests will pertain to both IPsec tunnel and transport mode.

What is specifically out of scope is any testing that pertains to considerations involving, L2TP [[RFC2661](#)], GRE [[RFC2784](#)], BGP/MPLS VPN's [[RFC2547](#)] and anything that does not specifically relate to the establishment and tearing down of IPsec tunnels.

## **3. Methodology Format**

The Methodology is described in the following format:

Objective: The reason for performing the test.

Topology: Physical test layout to be used as further clarified in [Section 6](#).

Procedure: Describes the method used for carrying out the test.

Reporting Format: Description of reporting of the test results.





#### **4. Key Words to Reflect Requirements**

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#). [RFC 2119](#) defines the use of these key words to help make the intent of standards track documents as clear as possible. While this document uses these keywords, this document is not a standards track document.

#### **5. Test Considerations**

Before any of the IPsec data plane benchmarking tests are carried out, a baseline MUST be established. (i.e. the particular test in question must first be executed to measure its performance without enabling IPsec). Once both the Baseline clear text performance and the performance using an IPsec enabled datapath have been measured, the difference between the two can be discerned.

This document explicitly assumes that you MUST follow logical performance test methodology that includes the pre-configuration or pre-population of routing protocols, ARP caches, IPv6 neighbor discovery and all other extraneous IPv4 and IPv6 parameters required to pass packets before the tester is ready to send IPsec protected packets. IPv6 nodes that implement Path MTU Discovery [[RFC1981](#)] MUST ensure that the PMTUD process has been completed before any of the tests have been run.

For every IPsec data plane benchmarking test, the SA database (SADB) MUST be created and populated with the appropriate SA's before any actual test traffic is sent, i.e. the DUT/SUT MUST have Active Tunnels. This may require manual commands to be executed on the DUT/SUT or the sending of appropriate learning frames to the DUT/SUT to trigger IKE negotiation. This is to ensure that none of the control plane parameters (such as IPsec Tunnel Setup Rates and IPsec Tunnel Rekey Rates) are factored into these tests.

For control plane benchmarking tests (i.e. IPsec Tunnel Setup Rate and IPsec Tunnel Rekey Rates), the authentication mechanism(s) used for the authenticated Diffie-Hellman exchange MUST be reported.

#### **6. Test Topologies**

The tests can be performed as a DUT or SUT. When the tests are performed as a DUT, the Tester itself must be an IPsec peer. This scenario is shown in Figure 1. When testing an IPsec Device as a DUT, one considerations that needs to be take into account is that



the Tester can introduce interoperability issues potentially limiting the scope of the tests that can be executed. On the other hand, this method has the advantage that IPsec client side testing can be performed as well as it is able to identify abnormalities and assymetry between the encryption and decryption behavior.

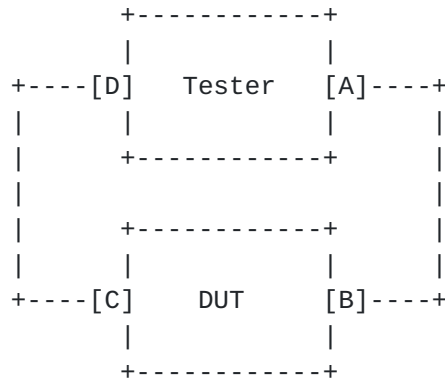


Figure 1: Device Under Test Topology

The SUT scenario is depicted in Figure 2. Two identical DUTs are used in this test set up which more accurately simulate the use of IPsec gateways. IPsec SA (i.e. AH/ESP transport or tunnel mode) configurations can be tested using this set-up where the tester is only required to send and receive cleartext traffic.

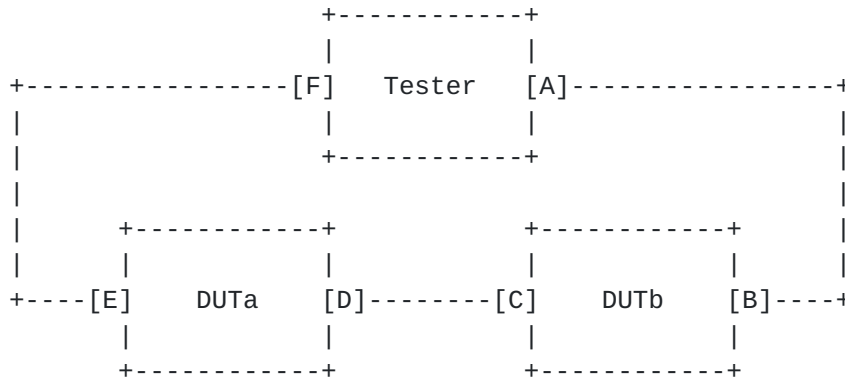


Figure 2: System Under Test Topology

When an IPsec DUT needs to be tested in a chassis failover topology, a second DUT needs to be used as shown in figure 3. This is the high-availability equivalent of the topology as depicted in Figure 1. Note that in this topology the Tester MUST be an IPsec peer.



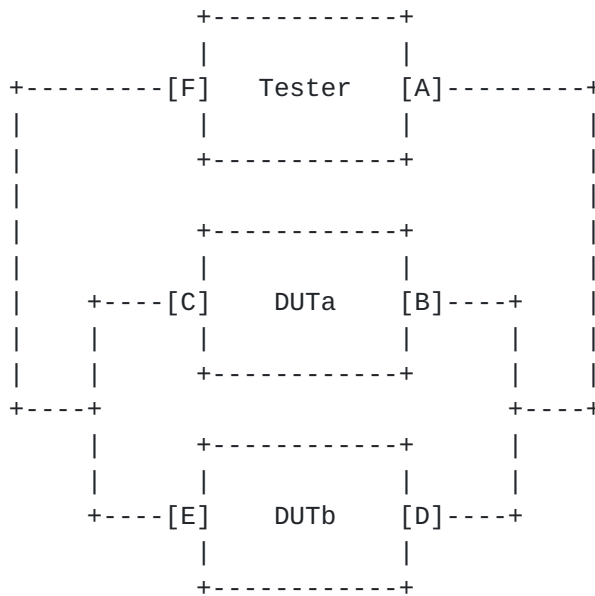


Figure 3: Redundant Device Under Test Topology

When no IPsec enabled Tester is available and an IPsec failover scenario needs to be tested, the topology as shown in Figure 4 can be used. In this case, either the high availability pair of IPsec devices can be used as an Initiator or as a Responder. The remaining chassis will take the opposite role.

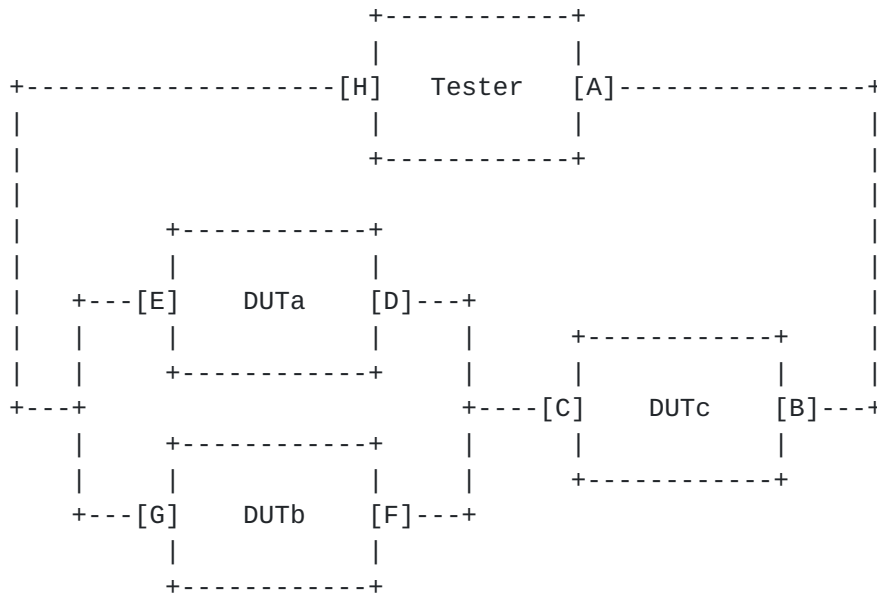


Figure 4: Redundant System Under Test Topology



## **7. Test Parameters**

For each individual test performed, all of the following parameters MUST be explicitly reported in any test results.

### **7.1. Frame Type**

#### **7.1.1. IP**

Both IPv4 and IPv6 frames MUST be used. The basic IPv4 header is 20 bytes long (which may be increased by the use of an options field). The basic IPv6 header is a fixed 40 bytes and uses an extension field for additional headers. Only the basic headers plus the IPsec AH and/or ESP headers MUST be present.

It is RECOMMENDED that IPv4 and IPv6 frames be tested separately to ascertain performance parameters for either IPv4 or IPv6 traffic. If both IPv4 and IPv6 traffic are to be tested, the device SHOULD be pre-configured for a dual-stack environment to handle both traffic types.

It is RECOMMENDED that a test payload field is added in the payload of each packet that allows flow identification and timestamping of a received packet.

#### **7.1.2. UDP**

It is also RECOMMENDED that the test is executed using UDP as the L4 protocol. When using UDP, instrumentation data SHOULD be present in the payload of the packet. It is OPTIONAL to have application payload.

#### **7.1.3. TCP**

It is OPTIONAL to perform the tests with TCP as the L4 protocol but in case this is considered, the TCP traffic is RECOMMENDED to be stateful. With a TCP as a L4 header it is possible that there will not be enough room to add all instrumentation data to identify the packets within the DUT/SUT.

#### **7.1.4. NAT-Traversal**

It is RECOMMENDED to test the scenario where IPsec protected traffic must traverse network address translation (NAT) gateways. This is commonly referred to as Nat-Traversal and requires UDP encapsulation.





## **7.2. Frame Sizes**

Each test MUST be run with different frame sizes. It is RECOMMENDED to use the following cleartext layer 2 frame sizes for IPv4 tests over Ethernet media: 64, 128, 256, 512, 1024, 1280, and 1518 bytes, per [RFC2544 section 9](#) [[RFC2544](#)]. The four CRC bytes are included in the frame size specified.

For GigabitEthernet, supporting jumboframes, the cleartext layer 2 framesizes used are 64, 128, 256, 512, 1024, 1280, 1518, 2048, 3072, 4096, 5120, 6144, 7168, 8192, 9234 bytes

For SONET these are: 47, 67, 128, 256, 512, 1024, 1280, 1518, 2048, 4096 bytes

To accommodate IEEE 802.1q and IEEE 802.3as it is RECOMMENDED to respectively include 1522 and 2000 byte framesizes in all tests.

Since IPv6 requires that every link has an MTU of 1280 octets or greater, it is MANDATORY to execute tests with cleartext layer 2 frame sizes that include 1280 and 1518 bytes. It is RECOMMENDED that additional frame sizes are included in the IPv6 test execution, including the maximum supported datagram size for the linktype used.

## **7.3. Fragmentation and Reassembly**

IPsec devices can and must fragment packets in specific scenarios. Depending on whether the fragmentation is performed in software or using specialized custom hardware, there may be a significant impact on performance.

In IPv4, unless the DF (don't fragment) bit is set by the packet source, the sender cannot guarantee that some intermediary device on the way will not fragment an IPsec packet. For transport mode IPsec, the peers must be able to fragment and reassemble IPsec packets. Reassembly of fragmented packets is especially important if an IPv4 port selector (or IPv6 transport protocol selector) is configured. For tunnel mode IPsec, it is not a requirement. Note that fragmentation is handled differently in IPv6 than in IPv4. In IPv6 networks, fragmentation is no longer done by intermediate routers in the networks, but by the source node that originates the packet. The path MTU discovery (PMTUD) mechanism is recommended for every IPv6 node to avoid fragmentation.

Packets generated by hosts that do not support PMTUD, and have not set the DF bit in the IP header, will undergo fragmentation before IPsec encapsulation. Packets generated by hosts that do support PMTUD will use it locally to match the statically configured MTU on



the tunnel. If you manually set the MTU on the tunnel, you must set it low enough to allow packets to pass through the smallest link on the path. Otherwise, the packets that are too large to fit will be dropped.

Fragmentation can occur due to encryption overhead and is closely linked to the choice of transform used. Since each test SHOULD be run with a maximum cleartext frame size (as per the previous section) it will cause fragmentation to occur since the maximum frame size will be exceeded. All tests MUST be run with the DF bit not set. It is also recommended that all tests be run with the DF bit set.

#### **7.4. Time To Live**

The source frames should have a TTL value large enough to accommodate the DUT/SUT. A Minimum TTL of 64 is RECOMMENDED.

#### **7.5. Trial Duration**

The duration of the test portion of each trial SHOULD be at least 60 seconds. In the case of IPsec tunnel rekeying tests, the test duration must be at least two times the IPsec tunnel rekey time to ensure a reasonable worst case scenario test.

#### **7.6. Security Context Parameters**

All of the security context parameters listed in [section 7.13](#) of the IPsec Benchmarking Terminology document MUST be reported. When merely discussing the behavior of traffic flows through IPsec devices, an IPsec context MUST be provided. In the cases where IKE is configured (as opposed to using manually keyed tunnels), both an IPsec and an IKE context MUST be provided. Additional considerations for reporting security context parameters are detailed below. These all MUST be reported.

##### **7.6.1. IPsec Transform Sets**

All tests should be done on different IPsec transform set combinations. An IPsec transform specifies a single IPsec security protocol (either AH or ESP) with its corresponding security algorithms and mode. A transform set is a combination of individual IPsec transforms designed to enact a specific security policy for protecting a particular traffic flow. At minimum, the transform set must include one AH algorithm and a mode or one ESP algorithm and a mode.



ESP Transform	Encryption Algorithm	Authentication Algorithm	Mode
1	NULL	HMAC-SHA1-96	Transport
2	NULL	HMAC-SHA1-96	Tunnel
3	3DES-CBC	HMAC-SHA1-96	Transport
4	3DES-CBC	HMAC-SHA1-96	Tunnel
5	AES-CBC-128	HMAC-SHA1-96	Transport
6	AES-CBC-128	HMAC-SHA1-96	Tunnel
7	NULL	AES-XCBC-MAC-96	Transport
8	NULL	AES-XCBC-MAC-96	Tunnel
9	3DES-CBC	AES-XCBC-MAC-96	Transport
10	3DES-CBC	AES-XCBC-MAC-96	Tunnel
11	AES-CBC-128	AES-XCBC-MAC-96	Transport
12	AES-CBC-128	AES-XCBC-MAC-96	Tunnel

Table 1

Testing of ESP Transforms 1-4 MUST be supported. Testing of ESP Transforms 5-12 SHOULD be supported.

AH Transform	Authentication Algorithm	Mode
1	HMAC-SHA1-96	Transport
2	HMAC-SHA1-96	Tunnel
3	AES-XBC-MAC-96	Transport
4	AES-XBC-MAC-96	Tunnel

Table 2

If AH is supported by the DUT/SUT testing of AH Transforms 1 and 2 MUST be supported. Testing of AH Transforms 3 And 4 SHOULD be supported.

Note that this these tables are derived from the Cryptographic Algorithms for AH and ESP requirements as described in [\[RFC4305\]](#). Optionally, other AH and/or ESP transforms MAY be supported.



```

+-----+-----+
| Transform Combination | AH | ESP |
+-----+-----+
|           1          | 1 | 1 |
|           2          | 2 | 2 |
|           3          | 1 | 3 |
|           4          | 2 | 4 |
+-----+-----+

```

Table 3

It is RECOMMENDED that the transforms shown in Table 3 be supported for IPV6 traffic selectors since AH may be used with ESP in these environments. Since AH will provide the overall authentication and integrity, the ESP Authentication algorithm MUST be Null for these tests. Optionally, other combined AH/ESP transform sets MAY be supported.

### 7.6.2. IPsec Topologies

All tests should be done at various IPsec topology configurations and the IPsec topology used MUST be reported. Since IPV6 requires the implementation of manual keys for IPsec, both manual keying and IKE configurations MUST be tested.

For manual keying tests, the IPsec SA's used should vary from 1 to 101, increasing in increments of 50. Although it is not expected that manual keying (i.e. manually configuring the IPsec SA) will be deployed in any operational setting with the exception of very small controlled environments (i.e. less than 10 nodes), it is prudent to test for potentially larger scale deployments.

For IKE specific tests, the following IPsec topologies MUST be tested:

- o 1 IKE SA & 2 IPsec SA (i.e. 1 IPsec Tunnel)
- o 1 IKE SA & {max} IPsec SA's
- o {max} IKE SA's & {max} IPsec SA's

It is RECOMMENDED to also test with the following IPsec topologies in order to gain more datapoints:

- o {max/2} IKE SA's & {(max/2) IKE SA's} IPsec SA's
- o {max} IKE SA's & {(max) IKE SA's} IPsec SA's





### **7.6.3. IKE Keepalives**

IKE keepalives track reachability of peers by sending hello packets between peers. During the typical life of an IKE Phase 1 SA, packets are only exchanged over this IKE Phase 1 SA when an IPsec IKE Quick Mode (QM) negotiation is required at the expiration of the IPsec Tunnel SA's. There is no standards-based mechanism for either type of SA to detect the loss of a peer, except when the QM negotiation fails. Most IPsec implementations use the Dead Peer Detection (i.e. Keepalive) mechanism to determine whether connectivity has been lost with a peer before the expiration of the IPsec Tunnel SA's.

All tests using IKEv1 MUST use the same IKE keepalive parameters.

### **7.6.4. IKE DH-group**

There are 3 Diffie-Hellman groups which can be supported by IPsec standards compliant devices:

- o DH-group 1: 768 bits
- o DH-group 2: 1024 bits
- o DH-group 14: 2048 bits

DH-group 2 MUST be tested, to support the new IKEv1 algorithm requirements listed in [[RFC4109](#)]. It is recommended that the same DH-group be used for both IKE Phase 1 and IKE phase 2. All test methodologies using IKE MUST report which DH-group was configured to be used for IKE Phase 1 and IKE Phase 2 negotiations.

### **7.6.5. IKE SA / IPsec SA Lifetime**

An IKE SA or IPsec SA is retained by each peer until the Tunnel lifetime expires. IKE SA's and IPsec SA's have individual lifetime parameters. In many real-world environments, the IPsec SA's will be configured with shorter lifetimes than that of the IKE SA's. This will force a rekey to happen more often for IPsec SA's.

When the initiator begins an IKE negotiation between itself and a remote peer (the responder), an IKE policy can be selected only if the lifetime of the responder's policy is shorter than or equal to the lifetime of the initiator's policy. If the lifetimes are not the same, the shorter lifetime will be used.

To avoid any incompatibilities in data plane benchmark testing, all devices MUST have the same IKE SA lifetime as well as an identical IPsec SA lifetime configured. Both SHALL be configured to a time



which exceeds the test duration timeframe and the total number of bytes to be transmitted during the test.

Note that the IPsec SA lifetime MUST be equal to or less than the IKE SA lifetime. Both the IKE SA lifetime and the IPsec SA lifetime used MUST be reported. This parameter SHOULD be variable when testing IKE rekeying performance.

#### **7.6.6. IPsec Selectors**

All tests MUST be performed using standard IPsec selectors as described in [\[RFC2401\] section 4.4.2.](#)

#### **7.6.7. NAT-Traversal**

For any tests that include network address translation considerations, the use of NAT-T in the test environment MUST be recorded.

### **8. Capacity**

#### **8.1. IPsec Tunnel Capacity**

Objective: Measure the maximum number of IPsec Tunnels or Active Tunnels that can be sustained on an IPsec Device.

Topology If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1.

Procedure: The IPsec Device under test initially MUST NOT have any Active IPsec Tunnels. The Initiator (either a tester or an IPsec peer) will start the negotiation of an IPsec Tunnel (a single Phase 1 SA and a pair Phase 2 SA's).

After it is detected that the tunnel is established, a limited number (50 packets RECOMMENDED) SHALL be sent through the tunnel. If all packet are received by the Responder (i.e. the DUT), a new IPsec Tunnel may be attempted.

This proces will be repeated until no more IPsec Tunnels can be established.



At the end of the test, a traffic pattern is sent to the initiator that will be distributed over all Established Tunnels, where each tunnel will need to propagate a fixed number of packets at a minimum rate of e.g. 5 pps. The aggregate rate of all Active Tunnels SHALL NOT exceed the IPsec Throughput. When all packets sent by the Initiator are being received by the Responder, the test has successfully determined the IKE SA Capacity. If however this final check fails, the test needs to be re-executed with a lower number of Active IPsec Tunnels. There MAY be a need to enforce a lower number of Active IPsec Tunnels i.e. an upper limit of Active IPsec Tunnel SHOULD be defined in the test.

During the entire duration of the test rekeying of Tunnels SHALL NOT be permitted. If a rekey event occurs, the test is invalid and MUST be restarted.

Reporting Format: The reporting format should reflect the maximum number of IPsec Tunnels that can be established when all packets by the initiator are received by the responder. In addition the Security Context parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of capacity.

## **8.2. IPsec SA Capacity**

Objective: Measure the maximum number of IPsec SA's that can be sustained on an IPsec Device.

Topology If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1.

Procedure: The IPsec Device under test initially MUST NOT have any Active IPsec Tunnels. The Initiator (either a tester or an IPsec peer) will start the negotiation of an IPsec Tunnel (a single Phase 1 SA and a pair Phase 2 SA's).

After it is detected that the tunnel is established, a limited number (50 packets RECOMMENDED) SHALL be sent through the tunnel. If all packets are received by the Responder (i.e. the DUT), a new pair of IPsec SA's may be attempted. This will be achieved by offering a specific traffic pattern to the Initiator that matches a given selector and therefore triggering the negotiation of a new pair of IPsec SA's.



This process will be repeated until no more IPsec SA' can be established.

At the end of the test, a traffic pattern is sent to the initiator that will be distributed over all IPsec SA's, where each SA will need to propagate a fixed number of packets at a minimum rate of 5 pps. When all packets sent by the Initiator are being received by the Responder, the test has successfully determined the IPsec SA Capacity. If however this final check fails, the test needs to be re-executed with a lower number of IPsec SA's. There MAY be a need to enforce a lower number IPsec SA's i.e. an upper limit of IPsec SA's SHOULD be defined in the test.

During the entire duration of the test rekeying of Tunnels SHALL NOT be permitted. If a rekey event occurs, the test is invalid and MUST be restarted.

Reporting Format: The reporting format SHOULD be the same as listed in [Section 8.1](#) for the maximum number of IPsec SAs.

## **9. Throughput**

This section contains the description of the tests that are related to the characterization of the packet forwarding of a DUT/SUT in an IPsec environment. Some metrics extend the concept of throughput presented in [[RFC1242](#)]. The notion of Forwarding Rate is cited in [[RFC2285](#)].

A separate test SHOULD be performed for Throughput tests using IPv4/UDP, IPv6/UDP, IPv4/TCP and IPv6/TCP traffic.

### **9.1. Throughput baseline**

Objective: Measure the intrinsic cleartext throughput of a device without the use of IPsec. The throughput baseline methodology and reporting format is derived from [[RFC2544](#)].

Topology If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1.

Procedure: Send a specific number of frames that matches the IPsec SA selector(s) to be tested at a specific rate through the DUT and then count the frames that are transmitted by the DUT. If the count of offered frames is equal to the count of received frames,





the rate of the offered stream is increased and the test is rerun. If fewer frames are received than were transmitted, the rate of the offered stream is reduced and the test is rerun.

The throughput is the fastest rate at which the count of test frames transmitted by the DUT is equal to the number of test frames sent to it by the test equipment.

Note that the IPsec SA selectors refer to the IP addresses and port numbers. So eventhough this is a test of only cleartext traffic, the same type of traffic should be sent for the baseline test as for tests utilizing IPsec.

Reporting Format: The results of the throughput test SHOULD be reported in the form of a graph. If it is, the x coordinate SHOULD be the frame size, the y coordinate SHOULD be the frame rate. There SHOULD be at least two lines on the graph. There SHOULD be one line showing the theoretical frame rate for the media at the various frame sizes. The second line SHOULD be the plot of the test results. Additional lines MAY be used on the graph to report the results for each type of data stream tested. Text accompanying the graph SHOULD indicate the protocol, data stream format, and type of media used in the tests.

Any values for throughput rate MUST be expressed in packets per second. The rate MAY also be expressed in bits (or bytes) per second if the vendor so desires. The statement of performance MUST include:

- \* Measured maximum frame rate
- \* Size of the frame used
- \* Theoretical limit of the media for that frame size
- \* Type of protocol used in the test

## **9.2. IPsec Throughput**

Objective: Measure the intrinsic throughput of a device utilizing IPsec.

Topology If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1.



Procedure: Send a specific number of cleartext frames that match the IPsec SA selector(s) at a specific rate through the DUT/SUT. DUTa will encrypt the traffic and forward to DUTb which will in turn decrypt the traffic and forward to the testing device. The testing device counts the frames that are transmitted by the DUTb. If the count of offered frames is equal to the count of received frames, the rate of the offered stream is increased and the test is rerun. If fewer frames are received than were transmitted, the rate of the offered stream is reduced and the test is rerun.

The IPsec Throughput is the fastest rate at which the count of test frames transmitted by the DUT/SUT is equal to the number of test frames sent to it by the test equipment.

For tests using multiple IPsec SA's, the test traffic associated with the individual traffic selectors defined for each IPsec SA MUST be sent in a round robin type fashion to keep the test balanced so as not to overload any single IPsec SA.

Reporting format: The reporting format SHALL be the same as listed in [Section 9.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

### **9.3. IPsec Encryption Throughput**

Objective: Measure the intrinsic DUT vendor specific IPsec Encryption Throughput.

Topology The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Send a specific number of cleartext frames that match the IPsec SA selector(s) at a specific rate to the DUT. The DUT will receive the cleartext frames, perform IPsec operations and then send the IPsec protected frame to the tester. Upon receipt of the encrypted packet, the testing device will timestamp the packet(s) and record the result. If the count of offered frames is equal to the count of received frames, the rate of the offered stream is increased and the test is rerun. If fewer frames are received than were transmitted, the rate of the offered stream is reduced and the test is rerun.

The IPsec Encryption Throughput is the fastest rate at which the count of test frames transmitted by the DUT is equal to the number of test frames sent to it by the test equipment.



For tests using multiple IPsec SA's, the test traffic associated with the individual traffic selectors defined for each IPsec SA MUST be sent in a round robin type fashion to keep the test balanced so as not to overload any single IPsec SA.

Reporting format: The reporting format SHALL be the same as listed in [Section 9.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

#### **9.4. IPsec Decryption Throughput**

Objective: Measure the intrinsic DUT vendor specific IPsec Decryption Throughput.

Topology The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Send a specific number of IPsec protected frames that match the IPsec SA selector(s) at a specific rate to the DUT. The DUT will receive the IPsec protected frames, perform IPsec operations and then send the cleartext frame to the tester. Upon receipt of the cleartext packet, the testing device will timestamp the packet(s) and record the result. If the count of offered frames is equal to the count of received frames, the rate of the offered stream is increased and the test is rerun. If fewer frames are received than were transmitted, the rate of the offered stream is reduced and the test is rerun.

The IPsec Decryption Throughput is the fastest rate at which the count of test frames transmitted by the DUT is equal to the number of test frames sent to it by the test equipment.

For tests using multiple IPsec SA's, the test traffic associated with the individual traffic selectors defined for each IPsec SA MUST be sent in a round robin type fashion to keep the test balanced so as not to overload any single IPsec SA.

Reporting format: The reporting format SHALL be the same as listed in [Section 9.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

## **10. Latency**

This section presents methodologies relating to the characterization of the forwarding latency of a DUT/SUT. It extends the concept of



latency characterization presented in [[RFC2544](#)] to an IPsec environment.

A separate tests SHOULD be performed for latency tests using IPv4/UDP, IPv6/UDP, IPv4/TCP and IPv6/TCP traffic.

In order to lessen the effect of packet buffering in the DUT/SUT, the latency tests MUST be run at the measured IPsec throughput level of the DUT/SUT; IPsec latency at other offered loads is optional.

Lastly, [[RFC1242](#)] and [[RFC2544](#)] draw distinction between two classes of devices: "store and forward" and "bit-forwarding". Each class impacts how latency is collected and subsequently presented. See the related RFC's for more information. In practice, much of the test equipment will collect the latency measurement for one class or the other, and, if needed, mathematically derive the reported value by the addition or subtraction of values accounting for medium propagation delay of the packet, bit times to the timestamp trigger within the packet, etc. Test equipment vendors SHOULD provide documentation regarding the composition and calculation latency values being reported. The user of this data SHOULD understand the nature of the latency values being reported, especially when comparing results collected from multiple test vendors (e.g., If test vendor A presents a "store and forward" latency result and test vendor B presents a "bit-forwarding" latency result, the user may erroneously conclude the DUT has two differing sets of latency values.).

### **10.1. Latency Baseline**

Objective: Measure the intrinsic latency (min/avg/max) introduced by a device without the use of IPsec.

Topology If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1.

Procedure: First determine the throughput for the DUT/SUT at each of the listed frame sizes. Send a stream of frames at a particular frame size through the DUT at the determined throughput rate using frames that match the IPsec SA selector(s) to be tested. The stream SHOULD be at least 120 seconds in duration. An identifying tag SHOULD be included in one frame after 60 seconds with the type of tag being implementation dependent. The time at which this frame is fully transmitted is recorded (timestamp A). The receiver logic in the test equipment MUST recognize the tag





information in the frame stream and record the time at which the tagged frame was received (timestamp B).

The latency is timestamp B minus timestamp A as per the relevant definition from [RFC 1242](#), namely latency as defined for store and forward devices or latency as defined for bit forwarding devices.

The test MUST be repeated at least 20 times with the reported value being the average of the recorded values.

**Reporting Format** The report MUST state which definition of latency (from [[RFC1242](#)]) was used for this test. The latency results SHOULD be reported in the format of a table with a row for each of the tested frame sizes. There SHOULD be columns for the frame size, the rate at which the latency test was run for that frame size, for the media types tested, and for the resultant latency values for each type of data stream tested.

## **[10.2.](#) IPsec Latency**

**Objective:** Measure the intrinsic IPsec Latency (min/avg/max) introduced by a device when using IPsec.

**Topology** If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1.

**Procedure:** First determine the throughput for the DUT/SUT at each of the listed frame sizes. Send a stream of cleartext frames at a particular frame size through the DUT/SUT at the determined throughput rate using frames that match the IPsec SA selector(s) to be tested. DUTa will encrypt the traffic and forward to DUTb which will in turn decrypt the traffic and forward to the testing device.

The stream SHOULD be at least 120 seconds in duration. An identifying tag SHOULD be included in one frame after 60 seconds with the type of tag being implementation dependent. The time at which this frame is fully transmitted is recorded (timestamp A). The receiver logic in the test equipment MUST recognize the tag information in the frame stream and record the time at which the tagged frame was received (timestamp B).



The IPsec Latency is timestamp B minus timestamp A as per the relevant definition from [[RFC1242](#)], namely latency as defined for store and forward devices or latency as defined for bit forwarding devices.

The test MUST be repeated at least 20 times with the reported value being the average of the recorded values.

Reporting format: The reporting format SHALL be the same as listed in [Section 10.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

### **[10.3](#). IPsec Encryption Latency**

Objective: Measure the DUT vendor specific IPsec Encryption Latency for IPsec protected traffic.

Topology The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Send a stream of cleartext frames at a particular frame size through the DUT/SUT at the determined throughput rate using frames that match the IPsec SA selector(s) to be tested.

The stream SHOULD be at least 120 seconds in duration. An identifying tag SHOULD be included in one frame after 60 seconds with the type of tag being implementation dependent. The time at which this frame is fully transmitted is recorded (timestamp A). The DUT will receive the cleartext frames, perform IPsec operations and then send the IPsec protected frames to the tester. Upon receipt of the encrypted frames, the receiver logic in the test equipment MUST recognize the tag information in the frame stream and record the time at which the tagged frame was received (timestamp B).

The IPsec Encryption Latency is timestamp B minus timestamp A as per the relevant definition from [[RFC1242](#)], namely latency as defined for store and forward devices or latency as defined for bit forwarding devices.

The test MUST be repeated at least 20 times with the reported value being the average of the recorded values.

Reporting format: The reporting format SHALL be the same as listed in [Section 10.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.



#### **10.4. IPsec Decryption Latency**

Objective: Measure the DUT Vendor Specific IPsec Decryption Latency for IPsec protected traffic.

Topology The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Send a stream of IPsec protected frames at a particular frame size through the DUT/SUT at the determined throughput rate using frames that match the IPsec SA selector(s) to be tested.

The stream SHOULD be at least 120 seconds in duration. An identifying tag SHOULD be included in one frame after 60 seconds with the type of tag being implementation dependent. The time at which this frame is fully transmitted is recorded (timestamp A). The DUT will receive the IPsec protected frames, perform IPsec operations and then send the cleartext frames to the tester. Upon receipt of the decrypted frames, the receiver logic in the test equipment MUST recognize the tag information in the frame stream and record the time at which the tagged frame was received (timestamp B).

The IPsec Decryption Latency is timestamp B minus timestamp A as per the relevant definition from [[RFC1242](#)], namely latency as defined for store and forward devices or latency as defined for bit forwarding devices.

The test MUST be repeated at least 20 times with the reported value being the average of the recorded values.

Reporting format: The reporting format SHALL be the same as listed in [Section 10.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

#### **10.5. Time To First Packet**

Objective: Measure the time it takes to transmit a packet when no SA's have been established.

Topology If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1.



Procedure: Determine the IPsec throughput for the DUT/SUT at each of the listed frame sizes. Start with a DUT/SUT with Configured Tunnels. Send a stream of cleartext frames at a particular frame size through the DUT/SUT at the determined throughput rate using frames that match the IPsec SA selector(s) to be tested.

The time at which the first frame is fully transmitted from the testing device is recorded as timestamp A. The time at which the testing device receives its first frame from the DUT/SUT is recorded as timestamp B. The Time To First Packet is the difference between Timestamp B and Timestamp A.

Note that it is possible that packets can be lost during IPsec Tunnel establishment and that timestamp A & B are not required to be associated with a unique packet.

Reporting format: The Time To First Packet results SHOULD be reported in the format of a table with a row for each of the tested frame sizes. There SHOULD be columns for the frame size, the rate at which the TTFP test was run for that frame size, for the media types tested, and for the resultant TTFP values for each type of data stream tested. The Security Context Parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance.

## **11. Frame Loss Rate**

This section presents methodologies relating to the characterization of frame loss rate, as defined in [[RFC1242](#)], in an IPsec environment.

### **11.1. Frame Loss Baseline**

Objective: To determine the frame loss rate, as defined in [[RFC1242](#)], of a DUT/SUT throughout the entire range of input data rates and frame sizes without the use of IPsec.

Topology If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1.

Procedure: Send a specific number of frames at a specific rate through the DUT/SUT to be tested using frames that match the IPsec SA selector(s) to be tested and count the frames that are transmitted by the DUT/SUT. The frame loss rate at each point is calculated using the following equation:





$$\left( \left( \text{input\_count} - \text{output\_count} \right) * 100 \right) / \text{input\_count}$$

The first trial SHOULD be run for the frame rate that corresponds to 100% of the maximum rate for the nominal device throughput, which is the throughput that is actually supported on an interface for a specific packet size and may not be the theoretical maximum. Repeat the procedure for the rate that corresponds to 90% of the maximum rate used and then for 80% of this rate. This sequence SHOULD be continued (at reduced 10% intervals) until there are two successive trials in which no frames are lost. The maximum granularity of the trials MUST be 10% of the maximum rate, a finer granularity is encouraged.

Reporting Format: The results of the frame loss rate test SHOULD be plotted as a graph. If this is done then the X axis MUST be the input frame rate as a percent of the theoretical rate for the media at the specific frame size. The Y axis MUST be the percent loss at the particular input rate. The left end of the X axis and the bottom of the Y axis MUST be 0 percent; the right end of the X axis and the top of the Y axis MUST be 100 percent. Multiple lines on the graph MAY be used to report the frame loss rate for different frame sizes, protocols, and types of data streams.

### **11.2. IPsec Frame Loss**

Objective: To measure the frame loss rate of a device when using IPsec to protect the data flow.

Topology When an IPsec aware tester is available the test MUST be executed using a Device Under Test Topology as depicted in Figure 1. If no IPsec aware tester is available the test MUST be conducted using a System Under Test Topology as depicted in Figure 2. In this scenario, it is common practice to use an asymmetric topology, where a less powerful (lower throughput) DUT is used in conjunction with a much more powerful IPsec device. This topology variant can in many cases produce more accurate results than the symmetric variant depicted in the figure, since all bottlenecks are expected to be on the less performant device.

Procedure: Ensure that the DUT/SUT is in active tunnel mode. Send a specific number of cleartext frames that match the IPsec SA selector(s) to be tested at a specific rate through the DUT/SUT. DUTa will encrypt the traffic and forward to DUTb which will in turn decrypt the traffic and forward to the testing device. The testing device counts the frames that are transmitted by the DUTb. The frame loss rate at each point is calculated using the following equation:



$$( ( \text{input\_count} - \text{output\_count} ) * 100 ) / \text{input\_count}$$

The first trial SHOULD be run for the frame rate that corresponds to 100% of the maximum rate for the nominal device throughput, which is the throughput that is actually supported on an interface for a specific packet size and may not be the theoretical maximum. Repeat the procedure for the rate that corresponds to 90% of the maximum rate used and then for 80% of this rate. This sequence SHOULD be continued (at reducing 10% intervals) until there are two successive trials in which no frames are lost. The maximum granularity of the trials MUST be 10% of the maximum rate, a finer granularity is encouraged.

Reporting Format: The reporting format SHALL be the same as listed in [Section 11.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

### **[11.3](#). IPsec Encryption Frame Loss**

Objective: To measure the effect of IPsec encryption on the frame loss rate of a device.

Topology The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Send a specific number of cleartext frames that match the IPsec SA selector(s) at a specific rate to the DUT. The DUT will receive the cleartext frames, perform IPsec operations and then send the IPsec protected frame to the tester. The testing device counts the encrypted frames that are transmitted by the DUT. The frame loss rate at each point is calculated using the following equation:

$$( ( \text{input\_count} - \text{output\_count} ) * 100 ) / \text{input\_count}$$

The first trial SHOULD be run for the frame rate that corresponds to 100% of the maximum rate for the nominal device throughput, which is the throughput that is actually supported on an interface for a specific packet size and may not be the theoretical maximum. Repeat the procedure for the rate that corresponds to 90% of the maximum rate used and then for 80% of this rate. This sequence SHOULD be continued (at reducing 10% intervals) until there are two successive trials in which no frames are lost. The maximum granularity of the trials MUST be 10% of the maximum rate, a finer granularity is encouraged.



Reporting Format: The reporting format SHALL be the same as listed in [Section 11.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

#### **11.4. IPsec Decryption Frame Loss**

Objective: To measure the effects of IPsec encryption on the frame loss rate of a device.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Send a specific number of IPsec protected frames that match the IPsec SA selector(s) at a specific rate to the DUT. The DUT will receive the IPsec protected frames, perform IPsec operations and then send the cleartext frames to the tester. The testing device counts the cleartext frames that are transmitted by the DUT. The frame loss rate at each point is calculated using the following equation:

$$\left( \left( \text{input\_count} - \text{output\_count} \right) * 100 \right) / \text{input\_count}$$

The first trial SHOULD be run for the frame rate that corresponds to 100% of the maximum rate for the nominal device throughput, which is the throughput that is actually supported on an interface for a specific packet size and may not be the theoretical maximum. Repeat the procedure for the rate that corresponds to 90% of the maximum rate used and then for 80% of this rate. This sequence SHOULD be continued (at reducing 10% intervals) until there are two successive trials in which no frames are lost. The maximum granularity of the trials MUST be 10% of the maximum rate, a finer granularity is encouraged.

Reporting format: The reporting format SHALL be the same as listed in [Section 11.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

#### **11.5. IKE Phase 2 Rekey Frame Loss**

Objective: To measure the frame loss due to an IKE Phase 2 (i.e. IPsec SA) Rekey event.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.



Procedure: The procedure is the same as in [Section 11.2](#) with the exception that the IPsec SA lifetime MUST be configured to be one-third of the trial test duration or one-third of the total number of bytes to be transmitted during the trial duration.

Reporting format: The reporting format SHALL be the same as listed in [Section 11.1](#) with the additional requirement that the Security Context Parameters, as defined in [Section 7.6](#), utilized for this test MUST be included in any statement of performance.

## **[12.](#) IPsec Tunnel Setup Behavior**

### **[12.1.](#) IPsec Tunnel Setup Rate**

Objective: Determine the rate at which IPsec Tunnels can be established.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Configure the Responder (where the Responder is the DUT) with  $n$  IKE Phase 1 and corresponding IKE Phase 2 policies. Ensure that no SA's are established and that the Responder has Established Tunnels for all  $n$  policies. Send a stream of cleartext frames at a particular frame size to the Responder at the determined throughput rate using frames with selectors matching the first IKE Phase 1 policy. As soon as the testing device receives its first frame from the Responder, it knows that the IPsec Tunnel is established and starts sending the next stream of cleartext frames using the same frame size and throughput rate but this time using selectors matching the second IKE Phase 1 policy. This process is repeated until all configured IPsec Tunnels have been established.

Some devices may support policy configurations where you do not need a one-to-one correspondence between an IKE Phase 1 policy and a specific IKE SA. In this case, the number of IKE Phase 1 policies configured should be sufficient so that the transmitted (i.e. offered) test traffic will create ' $n$ ' IKE SAs.

The IPsec Tunnel Setup Rate is measured in Tunnels Per Second (TPS) and is determined by the following formula:

$$\text{Tunnel Setup Rate} = n / [\text{Duration of Test} - (n * \text{frame\_transmit\_time})] \text{ TPS}$$





The IKE SA lifetime and the IPsec SA lifetime MUST be configured to exceed the duration of the test time. It is RECOMMENDED that n=100 IPsec Tunnels are tested at a minimum to get a large enough sample size to depict some real-world behavior.

Reporting Format: The Tunnel Setup Rate results SHOULD be reported in the format of a table with a row for each of the tested frame sizes. There SHOULD be columns for:

The throughput rate at which the test was run for the specified frame size

The media type used for the test

The resultant Tunnel Setup Rate values, in TPS, for the particular data stream tested for that frame size

The Security Context Parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance.

## **12.2. IKE Phase 1 Setup Rate**

Objective: Determine the rate of IKE SA's that can be established.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Configure the Responder with n IKE Phase 1 and corresponding IKE Phase 2 policies. Ensure that no SA's are established and that the Responder has Configured Tunnels for all n policies. Send a stream of cleartext frames at a particular frame size through the Responder at the determined throughput rate using frames with selectors matching the first IKE Phase 1 policy. As soon as the Phase 1 SA is established, the testing device starts sending the next stream of cleartext frames using the same frame size and throughput rate but this time using selectors matching the second IKE Phase 1 policy. This process is repeated until all configured IKE SA's have been established.

Some devices may support policy configurations where you do not need a one-to-one correspondence between an IKE Phase 1 policy and a specific IKE SA. In this case, the number of IKE Phase 1 policies configured should be sufficient so that the transmitted (i.e. offered) test traffic will create 'n' IKE SAs.



The IKE SA Setup Rate is determined by the following formula:

$$\text{IKE SA Setup Rate} = n / [\text{Duration of Test} - (n * \text{frame\_transmit\_time})] \text{ IKE SAs per second}$$

The IKE SA lifetime and the IPsec SA lifetime MUST be configured to exceed the duration of the test time. It is RECOMMENDED that n=100 IKE SA's are tested at a minimum to get a large enough sample size to depict some real-world behavior.

Reporting Format: The IKE Phase 1 Setup Rate results SHOULD be reported in the format of a table with a row for each of the tested frame sizes. There SHOULD be columns for the frame size, the rate at which the test was run for that frame size, for the media types tested, and for the resultant IKE Phase 1 Setup Rate values for each type of data stream tested. The Security Context Parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance.

### **12.3. IKE Phase 2 Setup Rate**

Objective: Determine the rate of IPsec SA's that can be established.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Configure the Responder (where the Responder is the DUT) with a single IKE Phase 1 policy and n corresponding IKE Phase 2 policies. Ensure that no SA's are established and that the Responder has Configured Tunnels for all policies. Send a stream of cleartext frames at a particular frame size through the Responder at the determined throughput rate using frames with selectors matching the first IPsec SA policy.

The time at which the IKE SA is established is recorded as timestamp\_A. As soon as the Phase 1 SA is established, the IPsec SA negotiation will be initiated. Once the first IPsec SA has been established, start sending the next stream of cleartext frames using the same frame size and throughput rate but this time using selectors matching the second IKE Phase 2 policy. This process is repeated until all configured IPsec SA's have been established.

The IPsec SA Setup Rate is determined by the following formula, where test\_duration and frame\_transmit\_times are expressed in units of seconds:



IPsec SA Setup Rate =  $n / [\text{test\_duration} - \{\text{timestamp\_A} + ((n-1) * \text{frame\_transmit\_time})\}]$  IPsec SA's per Second

The IKE SA lifetime and the IPsec SA lifetime MUST be configured to exceed the duration of the test time. It is RECOMMENDED that n=100 IPsec SA's are tested at a minimum to get a large enough sample size to depict some real-world behavior.

Reporting Format: The IKE Phase 2 Setup Rate results SHOULD be reported in the format of a table with a row for each of the tested frame sizes. There SHOULD be columns for:

The throughput rate at which the test was run for the specified frame size

The media type used for the test

The resultant IKE Phase 2 Setup Rate values, in IPsec SA's per second, for the particular data stream tested for that frame size

The Security Context parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance.

### **13. IPsec Rekey Behavior**

The IPsec Rekey Behavior test all need to be executed by an IPsec aware test device since the test needs to be closely linked with the IKE FSM (Finite State Machine) and cannot be done by offering specific traffic pattern at either the Initiator or the Responder.

#### **13.1. IKE Phase 1 Rekey Rate**

Objective: Determine the maximum rate at which an IPsec Device can rekey IKE SA's.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: The IPsec Device under test should initially be set up with the determined IPsec Tunnel Capacity number of Active IPsec Tunnels.



The IPsec aware tester should then perform a binary search where it initiates an IKE Phase 1 SA rekey for all Active IPsec Tunnels. The tester MUST timestamp for each IKE SA when it initiated the rekey (timestamp\_A) and MUST timestamp once more once the FSM declares the rekey is completed (timestamp\_B). The rekey time for a specific SA equals timestamp\_B - timestamp\_A. Once the iteration is complete the tester now has a table of rekey times for each IKE SA. The reciprocal of the average of this table is the IKE Phase 1 Rekey Rate.

It is expected that all IKE SA were able to rekey successfully. If this is not the case, the IPsec Tunnels are all re-established and the binary search goes to the next value of IKE SA's it will rekey. The process will repeat itself until a rate is determined at which all SA's in that timeframe rekey correctly.

Reporting Format: The IKE Phase 1 Rekey Rate results SHOULD be reported in the format of a table with a row for each of the tested frame sizes. There SHOULD be columns for the frame size, the rate at which the test was run for that frame size, for the media types tested, and for the resultant IKE Phase 1 Rekey Rate values for each type of data stream tested. The Security Context Parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance.

### **13.2. IKE Phase 2 Rekey Rate**

Objective: Determine the maximum rate at which an IPsec Device can rekey IPsec SA's.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: The IPsec Device under test should initially be set up with the determined IPsec Tunnel Capacity number of Active IPsec Tunnels.

The IPsec aware tester should then perform a binary search where it initiates an IKE Phase 2 SA rekey for all IPsec SA's. The tester MUST timestamp for each IPsec SA when it initiated the rekey (timestamp\_A) and MUST timestamp once more once the FSM declares the rekey is completed (timestamp\_B). The rekey time for a specific IPsec SA is timestamp\_B - timestamp\_A. Once the iteration is complete the tester now has a table of rekey times for each IPsec SA. The reciprocal of the average of this table is the IKE Phase 2 Rekey Rate.





It is expected that all IPsec SA's were able to rekey successfully. If this is not the case, the IPsec Tunnels are all re-established and the binary search goes to the next value of IPsec SA's it will rekey. The process will repeat itself until a rate is determined at which a all SA's in that timeframe rekey correctly.

Reporting Format: The IKE Phase 2 Rekey Rate results SHOULD be reported in the format of a table with a row for each of the tested frame sizes. There SHOULD be columns for the frame size, the rate at which the test was run for that frame size, for the media types tested, and for the resultant IKE Phase 2 Rekey Rate values for each type of data stream tested. The Security Context Parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance.

#### **14. IPsec Tunnel Failover Time**

This section presents methodologies relating to the characterization of the failover behavior of a DUT/SUT in a IPsec environment.

In order to lessen the effect of packet buffering in the DUT/SUT, the Tunnel Failover Time tests MUST be run at the measured IPsec Throughput level of the DUT. Tunnel Failover Time tests at other offered constant loads are OPTIONAL.

Tunnel Failovers can be achieved in various ways, for example:

- o Failover between two Software Instances of an IPsec stack.
- o Failover between two IPsec devices.
- o Failover between two Hardware IPsec Engines within a single IPsec Device.
- o Fallback to Software IPsec from Hardware IPsec within a single IPsec Device.

In all of the above cases there shall be at least one active IPsec device and a standby device. In some cases the standby device is not present and two or more IPsec devices are backing eachother up in case of a catastrophic device or stack failure. The standby (or potential other active) IPsec Devices can back up the active IPsec Device in either a stateless or statefull method. In the former case, Phase 1 SA's as well as Phase 2 SA's will need to be re-established in order to guarantuee packet forwarding. In the latter case, the SPD and SADB of the active IPsec Device is synchronized to the standby IPsec Device to ensure immediate packet path recovery.



**Objective:** Determine the time required to fail over all Active Tunnels from an active IPsec Device to its standby device.

**Topology:** If no IPsec aware tester is available, the test MUST be conducted using a Redundant System Under Test Topology as depicted in Figure 4. When an IPsec aware tester is available the test MUST be executed using a Redundant Unit Under Test Topology as depicted in Figure 3. If the failover is being tested withing a single DUT e.g. crypto engine based failovers, a Device Under Test Topology as depicted in Figure 1 MAY be used as well.

**Procedure:** Before a failover can be triggered, the IPsec Device has to be in a state where the active stack/engine/node has a the maximum supported number of Active Tunnels. The Tunnels will be transporting bidirectional traffic at the determined IPsec Throughput rate for the smallest framesize that the stack/engine/node is capable of forwarding (In most cases, this will be 64 Bytes). The traffic should traverse in a round robin fashion through all Active Tunnels.

When traffic is flowing through all Active Tunnels in steady state, a failover shall be triggered.

Both receiver sides of the testers will now look at sequence counters in the instrumented packets that are being forwarded through the Tunnels. Each Tunnel MUST have its own counter to keep track of packetloss on a per SA basis.

If the tester observes no sequence number drops on any of the Tunnels in both directions then the Failover Time MUST be listed as 'null', indicating that the failover was immediate and without any packetloss.

In all other cases where the tester observes a gap in the sequence numbers of the instrumented payload of the packets, the tester will monitor all SA's and look for any Tunnels that are still not receiving packets after the Failover. These will be marked as 'pending' Tunnels. Active Tunnels that are forwarding packets again without any additional packet loss shall be marked as 'recovered' Tunnels. In background the tester will keep monitoring all SA's to make sure that no packets are dropped. If this is the case then the Tunnel in question will be placed back in 'pending' state.

Note that reordered packets can naturally occur after en/decryption. This is not a valid reason to place a Tunnel back in 'pending' state.



The tester will wait until all Tunnel are marked as 'recovered'. Then it will find the SA with the largest gap in sequence number. Given the fact that the framesize is fixed and the time of that framesize can easily be calculated for the initiator links, a simple multiplication of the framesize time \* largest packetloss gap will yield the Tunnel Failover Time.

This test MUST be repeated for the single tunnel, maximum throughput failover case. It is RECOMMENDED that the test is repeated for various number of Active Tunnels as well as for different framesizes and framerates.

Reporting Format: The results shall be represented in a tabular format, where the first column will list the number of Active Tunnels, the second column the Framesize, the third column the Framerate and the fourth column the Tunnel Failover Time in milliseconds.

## **15. DoS Attack Resiliency**

### **15.1. Phase 1 DoS Resiliency Rate**

Objective: Determine how many invalid IKE phase 1 sessions can be directed at a DUT before the Responder ignores or rejects valid IKE SA attempts.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: Configure the Responder with n IKE Phase 1 and corresponding IKE Phase 2 policies, where n is equal to the IPsec Tunnel Capacity. Ensure that no SA's are established and that the Responder has Configured Tunnels for all n policies. Start with 95% of the offered test traffic containing an IKE Phase 1 policy mismatch (either a mismatched pre-shared-key or an invalid certificate).

Send a burst of cleartext frames at a particular frame size through the Responder at the determined throughput rate using frames with selectors matching all n policies. Once the test completes, check whether all 5% of the correct IKE Phase 1 SAs have been established. If not, keep repeating the test by decrementing the number of mismatched IKE Phase 1 policies configured by 5% until all correct IKE Phase 1 SAs have been established. Between each retest, ensure that the DUT is reset and cleared of all previous state information.



The IKE SA lifetime and the IPsec SA lifetime MUST be configured to exceed the duration of the test time. It is RECOMMENDED that the test duration is  $2 \times (n \times \text{IKE SA set up rate})$  to ensure that there is enough time to establish the valid IKE Phase 1 SAs.

Some devices may support policy configurations where you do not need a one-to-one correspondence between an IKE Phase 1 policy and a specific IKE SA. In this case, the number of IKE Phase 1 policies configured should be sufficient so that the transmitted (i.e. offered) test traffic will create 'n' IKE SAs.

Reporting Format: The result shall be represented as the highest percentage of invalid IKE Phase1 messages that still allowed all the valid attempts to be completed. The Security Context Parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance.

### **[15.2.](#) Phase 2 Hash Mismatch DoS Resiliency Rate**

Objective: Determine the rate of Hash Mismatched packets at which a valid IPsec stream starts dropping frames.

Topology: The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

Procedure: A stream of IPsec traffic is offered to a DUT for decryption. This stream consists of two microflows. One valid microflow and one that contains altered IPsec packets with a Hash Mismatch. The aggregate rate of both microflows MUST be equal to the IPsec Throughput and should therefore be able to pass the DUT. A binary search will be applied to determine the ratio between the two microflows that causes packetloss on the valid microflow of traffic.

The test MUST be conducted with a single Active Tunnel. It MAY be repeated at various Tunnel scalability data points (e.g. 90%).

Reporting Format: The results shall be listed as PPS (of invalid traffic). The Security Context Parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance. The aggregate rate of both microflows which act as the offered testing load MUST also be reported.

### **[15.3.](#) Phase 2 Anti Replay Attack DoS Resiliency Rate**





**Objective:** Determine the rate of replayed packets at which a valid IPsec stream start dropping frames.

**Topology:** The test MUST be conducted using a Device Under Test Topology as depicted in Figure 1.

**Procedure:** A stream of IPsec traffic is offered to a DUT for decryption. This stream consists of two microflows. One valid microflow and one that contains replayed packets of the valid microflow. The aggregate rate of both microflows MUST be equal to the IPsec Throughput and should therefore be able to pass the DUT. A binary search will be applied to determine the ratio between the two microflows that causes packet loss on the valid microflow of traffic.

The replayed packets should always be offered within the window of which the original packet arrived i.e. it MUST be replayed directly after the original packet has been sent to the DUT. The binary search SHOULD start with a low anti-replay count where every few anti-replay windows, a single packet in the window is replayed. To increase this, one should obey the following sequence:

- \* Increase the replayed packets so every window contains a single replayed packet
- \* Increase the replayed packets so every packet within a window is replayed once
- \* Increase the replayed packets so packets within a single window are replayed multiple times following the same fill sequence

If the flow of replayed traffic equals the IPsec Throughput, the flow SHOULD be increased till the point where packetloss is observed on the replayed traffic flow.

The test MUST be conducted with a single Active Tunnel. It MAY be repeated at various Tunnel scalability data points. The test SHOULD also be repeated on all configurable anti-replay window sizes.

**Reporting Format:** PPS (of replayed traffic). The Security Context Parameters defined in [Section 7.6](#) and utilized for this test MUST be included in any statement of performance.



## **16. Security Considerations**

As this document is solely for the purpose of providing test benchmarking methodology and describes neither a protocol nor a protocol's implementation; there are no security considerations associated with this document.

## **17. Acknowledgements**

The authors would like to acknowledge the following individuals for their help and participation of the compilation and editing of this document: Michele Bustos, Paul Hoffman, Benno Overeinder, Scott Poretzky, Yaron Sheffer and Al Morton.

## **18. References**

### **18.1. Normative References**

- [RFC1242] Bradner, S., "Benchmarking terminology for network interconnection devices", [RFC 1242](#), July 1991.
- [RFC1981] McCann, J., Deering, S., and J. Mogul, "Path MTU Discovery for IP version 6", [RFC 1981](#), August 1996.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2285] Mandeville, R., "Benchmarking Terminology for LAN Switching Devices", [RFC 2285](#), February 1998.
- [RFC2393] Shacham, A., Monsour, R., Pereira, R., and M. Thomas, "IP Payload Compression Protocol (IPComp)", [RFC 2393](#), December 1998.
- [RFC2401] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", [RFC 2401](#), November 1998.
- [RFC2402] Kent, S. and R. Atkinson, "IP Authentication Header", [RFC 2402](#), November 1998.
- [RFC2403] Madson, C. and R. Glenn, "The Use of HMAC-MD5-96 within ESP and AH", [RFC 2403](#), November 1998.
- [RFC2404] Madson, C. and R. Glenn, "The Use of HMAC-SHA-1-96 within ESP and AH", [RFC 2404](#), November 1998.



- [RFC2405] Madson, C. and N. Doraswamy, "The ESP DES-CBC Cipher Algorithm With Explicit IV", [RFC 2405](#), November 1998.
- [RFC2406] Kent, S. and R. Atkinson, "IP Encapsulating Security Payload (ESP)", [RFC 2406](#), November 1998.
- [RFC2407] Piper, D., "The Internet IP Security Domain of Interpretation for ISAKMP", [RFC 2407](#), November 1998.
- [RFC2408] Maughan, D., Schneider, M., and M. Schertler, "Internet Security Association and Key Management Protocol (ISAKMP)", [RFC 2408](#), November 1998.
- [RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", [RFC 2409](#), November 1998.
- [RFC2410] Glenn, R. and S. Kent, "The NULL Encryption Algorithm and Its Use With IPsec", [RFC 2410](#), November 1998.
- [RFC2411] Thayer, R., Doraswamy, N., and R. Glenn, "IP Security Document Roadmap", [RFC 2411](#), November 1998.
- [RFC2412] Orman, H., "The OAKLEY Key Determination Protocol", [RFC 2412](#), November 1998.
- [RFC2432] Dubray, K., "Terminology for IP Multicast Benchmarking", [RFC 2432](#), October 1998.
- [RFC2451] Pereira, R. and R. Adams, "The ESP CBC-Mode Cipher Algorithms", [RFC 2451](#), November 1998.
- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", [RFC 2544](#), March 1999.
- [RFC2547] Rosen, E. and Y. Rekhter, "BGP/MPLS VPNs", [RFC 2547](#), March 1999.
- [RFC2661] Townsley, W., Valencia, A., Rubens, A., Pall, G., Zorn, G., and B. Palter, "Layer Two Tunneling Protocol "L2TP"", [RFC 2661](#), August 1999.
- [RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", [RFC 2784](#), March 2000.
- [RFC4109] Hoffman, P., "Algorithms for Internet Key Exchange version 1 (IKEv1)", [RFC 4109](#), May 2005.



- [RFC4305] Eastlake, D., "Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)", [RFC 4305](#), December 2005.
- [RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", [RFC 4306](#), December 2005.
- [RFC5180] Popoviciu, C., Hamza, A., Van de Velde, G., and D. Dugatkin, "IPv6 Benchmarking Methodology for Network Interconnect Devices", [RFC 5180](#), May 2008.
- [I-D.ietf-ipsec-properties]  
Krywaniuk, A., "Security Properties of the IPsec Protocol Suite", [draft-ietf-ipsec-properties-02](#) (work in progress), July 2002.

## **18.2. Informative References**

- [FIPS.186-1.1998]  
National Institute of Standards and Technology, "Digital Signature Standard", FIPS PUB 186-1, December 1998, <<http://csrc.nist.gov/fips/fips1861.pdf>>.

### Authors' Addresses

Merike Kaeo  
Double Shot Security  
3518 Fremont Ave N #363  
Seattle, WA 98103  
USA

Phone: +1(310)866-0165  
Email: [kaeo@merike.com](mailto:kaeo@merike.com)

Tim Van Herck  
Cisco Systems  
170 West Tasman Drive  
San Jose, CA 95134-1706  
USA

Phone: +1(408)853-2284  
Email: [herckt@cisco.com](mailto:herckt@cisco.com)



