Benchmarking Working Group M. Bustos
Internet-Draft IXIA
Expires: August 22, 2005 T. Van Herck
Cisco Systems
M. Kaeo
Double Shot Security
February 18, 2005

Terminology for Benchmarking IPsec Devices
draft-ietf-bmwg-ipsec-term-05

Status of this Memo

Copyright Notice

Abstract

This purpose of this document is to define terminology specific to measuring the performance of IPsec devices. It builds upon the

tenets set forth in [RFC1242], [RFC2544], [RFC2285] and other IETF Benchmarking Methodology Working Group (BMWG) documents used for benchmarking routers and switches. This document seeks to extend these efforts specific 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

Despite the need to secure communications over a public medium there is no standard method of performance measurement nor a standard in the terminology used to develop such hardware and software solutions. This results in varied implementations which challenge interoperability and direct performance comparisons. Standardized IPsec terminology and performance test methodologies will enable users to determine if the IPsec device they select will withstand loads of secured traffic that meet their requirements.

To appropriately define the parameters and scope of this document, this section will give a brief overview of the IPsec standard:

2. IPsec Fundamentals

IPsec is a framework of open standards that provides data confidentiality, data integrity, and data authenticity between participating peers. IPsec provides these security services at the IP layer. IPsec uses IKE to handle negotiation of protocols and algorithms based on local policy, and to generate the encryption and authentication keys to be used. IPsec can be used to protect one or more data flows between a pair of hosts, between a pair of security gateways, or between a security gateway and a host. The IPsec protocol suite set of standards is documented in RFC's [RFC2401] through [RFC2412] and [RFC2451]. The reader is assumed to be familiar with these documents. Some Internet Drafts supersede these RFC's and will be taken into consideration.

IPsec itself defines the following:

Authentication Header (AH): A security protocol, defined in [RFC2402], which provides data authentication and optional anti-replay services. AH ensures the integrity and data origin authentication of the IP datagram as well as the invariant fields in the outer IP header.

Encapsulating Security Payload (ESP): A security protocol, defined in [RFC2406], which provides confidentiality, data origin authentication, connectionless integrity, an anti-replay service and limited traffic flow confidentiality. The set of services provided depends on options selected at the time of Security Association (SA) establishment and on the location of the implementation in a network topology. ESP authenticates only headers and data after the IP header.

Internet Key Exchange (IKE): A hybrid protocol which implements Oakley [RFC2412] and SKEME [SKEME] key exchanges inside the ISAKMP

framework. While IKE can be used with other protocols, its initial implementation is with the IPsec protocol. IKE provides authentication of the IPsec peers, negotiates IPsec security associations, and establishes IPsec keys.

The AH and ESP protocols each support two modes of operation: transport mode and tunnel mode. In transport mode, two hosts provide protection primarily for upper-layer protocols. The cryptographic endpoints (where the encryption and decryption take place) are the source and destination of the data packet. In IPv4, a transport mode security protocol header appears immediately after the IP header and before any higher-layer protocols (such as TCP or UDP).

In the case of AH in transport mode, all upper-layer information is protected, and all fields in the IPv4 header excluding the fields typically are modified in transit. The fields of the IPv4 header that are not included are, therefore, set to 0 before applying the authentication algorithm. These fields are as follows:

* TOS
* TTL
* Header Checksum
* Offset
* Flags

In the case of ESP in transport mode, security services are provide only for the higher-layer protocols, not for the IP header.

A tunnel is a vehicle for encapsulating packets inside a protocol that is understood at the entry and exit points of a given network. These entry and exit points are defined as tunnel interfaces.

Both the AH and ESP protocols can be used in tunnel mode for data packet endpoints as well as by intermediate security gateways. In tunnel mode, there is an "outer" IP header that specifies the IPsec processing destination, plus an "inner" IP header that specifies the ultimate destination for the packet. The source address in the outer IP header is the initiating cryptographic endpoint; the source address in the inner header is the true source address of the packet. The security protocol header appears after the outer IP header and before the inner IP header.

If AH is employed in tunnel mode, portions of the outer IP header are given protection (those same fields as for transport mode, described earlier in this section), as well as all of the tunneled IP packet (that is, all of the inner IP header is protected as are the higher-layer protocols). If ESP is employed, the protection is afforded only to the tunneled packet, not to the outer header.

2.1 IPsec Operation

2.1.1 Security Associations

The concept of a Security Association (SA) is fundamental to IPsec. An SA is a relationship between two or more entities that describes how the entities will use security services to communicate. The SA includes: an encryption algorithm, an authentication algorithm and a shared session key.

Because an SA is unidirectional, two SA's (one in each direction) are required to secure typical, bidirectional communication between two entities. The security services associated with an SA can be used for AH or ESP, but not for both. If both AH and ESP protection is applied to a traffic stream, two (or more) SA's are created for each direction to protect the traffic stream.

The SA is uniquely identified by the Security Parameter Index (SPI) [RFC2406]. When a system sends a packet that requires IPsec protection, it looks up the SA in its database and applies the specified processing and security protocol (AH/ESP), inserting the SPI from the SA into the IPsec header. When the IPsec peer receives the packet, it looks up the SA in its database by destination address, protocol, and SPI and then processes the packet as required.

2.1.2 Key Management

IPsec uses cryptographic keys for authentication, integrity and encryption services. Both manual provisioning and automatic distribution of keys is supported. IKE is specified as the public-key-based approach for automatic key management.

IKE authenticates each peer involved in IPsec, negotiates the security policy, and handles the exchange of session keys. IKE is a hybrid protocol, combining parts of the following protocols to negotiate and derive keying material for SA's in a secure and authenticated manner:

1. ISAKMP [RFC2408] (Internet Security Association and Key Management Protocol), which provides a framework for authentication and key exchange but does not define them. ISAKMP is designed to be key exchange independent; it is designed to support many different key exchanges.

2. Oakley [RFC2412], which describes a series of key exchanges, called modes, and details the services provided by each (for example, perfect forward secrecy for keys, identity protection, and authentication).

3. [SKEME] (Secure Key Exchange Mechanism for Internet), which describes a versatile key exchange technique that provides anonymity, reputability, and quick key refreshment.

IKE creates an authenticated, secure tunnel between two entities and then negotiates the security association for IPsec. This is performed in two phases.

In Phase 1, the two unidirectional SA's establish a secure, authenticated channel with which to communicate. Phase 1 has two distinct modes; Main Mode and Aggressive Mode. Main Mode for Phase 1 provides identity protection. When identity protection is not needed, Aggressive Mode can be used. The completion of Phase 1 is called an IKE SA.

The following attributes are used by IKE and are negotiated as part of the IKE SA:

- Encryption algorithm.

- Hash algorithm.

- Authentication method (digital signature, public-key encryption or pre-shared key).

- Diffie-Hellman group information.

After the attributes are negotiated, both parties must be authenticated to each other. IKE supports multiple authentication methods. The following mechanisms are generally implemented:

- Pre-shared keys: The same key is pre-installed on each host. IKE peers authenticate each other by computing and sending a keyed hash of data that includes the pre-shared key. If the receiving peer can independently create the same hash using its preshared key, it knows that both parties must share the same secret, and thus the other party is authenticated.

- Public key cryptography: Each party generates a pseudo-random number (a nonce) and encrypts it and its ID using the other party's public key. The ability for each party to compute a keyed hash containing the other peer's nonce and ID, decrypted with the local private key, authenticates the parties to each other. This method does not provide nonrepudiation; either side of the exchange could plausibly deny that it took part in the exchange.

- Digital signature: Each device digitally signs a set of data and sends it to the other party. This method is similar to the

public-key cryptography approach except that it provides nonrepudiation.

Note that both digital signature and public-key cryptography require the use of digital certificates to validate the public/private key mapping. IKE allows the certificate to be accessed independently or by having the two devices explicitly exchange certificates as part of IKE. Both parties must have a shared session key to encrypt the IKE tunnel. The Diffie-Hellman protocol is used to agree on a common session key.

In Phase 2 of IKE, SA's are negotiated for ESP and/or AH. These SA's will be called IPsec SA's. These IPsec SA's use a different shared key than that used for the IKE_SA. The IPsec SA shared key can be derived by using Diffie-Hellman again or by refreshing the shared key derived from the original Diffie-Hellman exchange that generated the IKE_SA by hashing it with nonces. Once the shared key is derived and additional communication parameters are negotiated, the IPsec SA's are established and traffic can be exchanged using the negotiated parameters.

3. Document Scope

The primary focus of this document is to establish useful performance testing terminology for IPsec devices that support IKEv1. We want to constrain the terminology specified in this document to meet the requirements of the Methodology for Benchmarking IPsec Devices documented test methodologies. The testing will be constrained to devices acting as IPsec gateways and will pertain to both IPsec tunnel and transport mode.

Any testing involving interoperability and/or conformance issues, L2TP [RFC2661], GRE [RFC2784], MPLS VPN's [RFC2547], multicast, and anything that does not specifically relate to the establishment and tearing down of IPsec tunnels is specifically out of scope. It is assumed that all relevant networking parameters that facilitate in the running of these tests are pre-configured (this includes at a minimum ARP caches and routing tables).

4. Definition Format

The definition format utilized by this document is described in [RFC1242], Section 2.

Term to be defined.

Definition:

The specific definition for the term.

Discussion:

A brief discussion of the term, its application, or other information that would build understanding.

Issues:

List of issues or conditions that affect this term. This field can present items the may impact the term's related methodology or otherwise restrict its measurement procedures.

[Measurement units:]

Units used to record measurements of this term. This field is mandatory where applicable. This field is optional in this document.

[See Also:]

List of other terms that are relevant to the discussion of this term. This field is optional in this document.

5. 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.

6. Existing Benchmark Definitions

It is recommended that readers consult [RFC1242], [RFC2544] and [RFC2285] before making use of this document. These and other IETF Benchmarking Methodology Working Group (BMWG) router and switch documents contain several existing terms relevant to benchmarking the performance of IPsec devices. The conceptual framework established in these earlier RFC's will be evident in this document.

This document also draws on existing terminology defined in other BMWG documents. Examples include, but are not limited to:

Throughput	[RFC 1242, section 3.17]
Latency	[RFC 1242, section 3.8]
Frame Loss Rate	[RFC 1242, section 3.6]
Forwarding Rates	[RFC 2285, section 3.6]
Loads	[RFC 2285, section 3.5]

7. Definitions

7.1 IPsec

Definition:

IPsec or IP Security protocol suite which comprises a set of standards used to provide security services at the IP layer.

Discussion:

IPsec is a framework of protocols that offer authentication, authenticity and encryption services to the IP and/or upper layer protocols. The major components of the protocol suite are IKE, used for key exchanges, and IPsec protocols such as AH and ESP, which use the exchanged keys to protect payload traffic.

Issues:

N/A

See Also:

IPsec Device, IKE, ISAKMP, ESP, AH

7.2 ISAKMP

Definition:

The Internet Security Association and Key Management Protocol, which provides a framework for authentication and key exchange but does not define them. ISAKMP is designed to be key exchange independent; it is designed to support many different key exchanges. ISAKMP is defined in [RFC2407].

Discussion:

Though ISAKMP is only a framework for the IPsec standard key management protocol, it is often misused and interchanged with the term 'IKE', which is an implementation of ISAKMP.

Issues:

When implementations refer to the term 'ISAKMP SA', it refers to an IKE Phase 1 SA.

See Also:

IKE, Security Association

7.3 IKE

Definition:

A hybrid key management protocol that allows secure negotiation of IPsec SA paramters.

Discussion:

A hybrid protocol, defined in [RFC2409], from the following 3 protocols:

* ISAKMP (Internet Security Association and Key Management Protocol), which provides a framework for authentication and key exchange but does not define them. ISAKMP is designed to be key exchange independent; it is designed to support many different key exchanges.

* Oakley, which describes a series of key exchanges, called modes, and details the services provided by each (for example, perfect forward secrecy for keys, identity protection, and authentication). [RFC2412]

* [SKEME] (Secure Key Exchange Mechanism for Internet), which describes a versatile key exchange technique that provides anonymity, reputability, and quick key refreshment.

Note that IKE is an optional protocol within the IPsec framework. Tunnels may also be manually configured i.e. the network administrator will provide keys that will be associated with the Phase 2 SA's as long as the IPsec Tunnel is configured. This method is the most basic mechanism to establish an IPsec tunnel between two IPsec devices but it also reduces the level of protection since the keys are static and as a result are more prone to various attacks. When IKE is employed as a key management protocol, the keys will change on a regular basis (time and/or traffic volume based) as part of the IKE rekeying mechanism.

Issues:

During the first IPsec deployment experiences, ambiguities were found in the IKEv1 specification, which lead to interoperability problems. To resolve these issues, IKEv1 is being updated by IKEv2.

See Also:

ISAKMP, IPsec, Security Association

7.3.1 IKE Phase 1

Definition:

The shared policy and key(s) used by negotiating peers to establish a secure authenticated "control channel" for further IKE communications.

Discussion:

The IPsec framework mandates that SPI's are used to secure payload traffic. If IKE is employed all SPI information will be exchanged between the IPsec devices. This has to be done in a secure fashion and for that reason IKE will set up a secure "control channel" over which it can exchange this information.

Note that IKE is an optional protocol within the IPsec framework and keys can also be manually configured.

Issues:

In some documents often referenced as ISAKMP SA or IKE SA.

See Also:

IKE, ISAKMP

7.3.2 IKE Phase 1 Main Mode

Definition:

Main Mode is an instantiation of the ISAKMP Identity Protect Exchange, defined in [RFC2409]. Upon successful completion it results in the establishment of an IKE Phase 1 SA.

Discussion:

IKE Main Mode use 3 distinct message pairs, for a total of 6 messages. The first two messages negotiate policy; the next two represent Diffie-Hellman public values and ancillary data (e.g. nonces); and the last two messages authenticate the Diffie-Hellman Exchange. The authentication method negotiated as part of the initial IKE Phase 1 influence the composition of the payloads but not their purpose.

Issues:

N/A

See Also:

ISAKMP, IKE, IKE Phase 1, Phase 1 Aggressive Mode

7.3.3 IKE Phase 1 Aggressive Mode

Definition:

Aggressive Mode is an instantiation of the ISAKMP Aggressive Exchange, defined in [RFC2409]. Upon successful completion it results in the establishment of an IKE Phase 1 SA.

Discussion:

IKE Aggressive Mode uses 3 messages. The first two messages negotiate policy, exchange Diffie-Hellman public values and ancillary data necessary for the exchange, and identities. In addition the second message authenticates the Responder. The third message authenticates the Initiator and provides proof of participation in the exchange.

Issues:

For IKEv1 the standard specifies that all implementations use both main and agressive mode, however, it is common to use only main mode.

See Also:

ISAKMP, IKE, IKE Phase 1, Phase 1 Main Mode

7.3.4 IKE Phase 2

Definition:

ISAKMP phase which upon successful completion establishes the shared keys used by the negotiating peers to set up a secure "data channel" for IPsec.

Discussion:

The main purpose of Phase 2 is to produce the key for the IPsec tunnel. Phase 2 is also used to regenerate the key being used for IPsec (called "rekeying"), as well as for exchanging informational messages.

Issues:

In other documents also referenced as IPsec SA.

See Also:

IKE Phase 1, ISAKMP, IKE

7.3.5 Phase 2 Quick Mode

Definition:

Quick Mode is an instanciation of IKE Phase 2. After successful completion it will result in one or typically two or more IPsec SA's

Discussion:

Quick Mode is used to negotiate the SA's and keys that will be used to protect the user data, i.e. the IPsec SA. Three different messages are exchanged, which are protected by the security parameters negotiated by the IKE phase 1 exchange. An additional Diffie-Hellman exchange may be performed if PFS (Perfect Forward Secrecy) is enabled.

Issues:

N/A

See Also:

ISAKMP, IKE, IKE Phase 2

7.4 Security Association (SA)

Definition:

A set of policy and key(s) used to protect traffic flows that require authentication and/or encryption services. It is a negotiation agreement between two IPsec devices, specifically the Initiator and Responder.

Discussion:

A simplex (unidirectional) logical connection that links a traffic flow to a set of security parameters. All traffic traversing an SA is provided the same security processing and will be subjected to a common set of encryption and/or authentication algorithms. In IPsec, an SA is an Internet layer abstraction implemented through the use of AH or ESP as defined in [RFC2401].

Issues:

N/A

See Also:

Initiator, Responder

7.5 Selectors

Definition:

A mechanism used for the classification of traffic flows that require authentication and/or encryption services.

Discussion:

The selectors are a set of fields that will be extracted from the network and transport layer headers that provide the ability to classify the traffic flow and associate it with an SA.

After classification, a decision can be made if the traffic needs to be encrypted/decrypted and how this should be done depending on the SA linked to the traffic flow. Simply put, selectors classify IP packets that require IPsec processing and those packets that must be passed along without any intervention of the IPsec

framework.

Selectors are flexible objects that can match on ranges of source and destination addresses and ranges of source and destination ports.

Issues:

Both sides must agree exactly on both the networks being protected, and they both must agree on how to describe the networks (range, subnet, addresses). This is a common point of non-interoperability.

7.6 IPsec Device

Definition:

Any implementation that has the ability to process data flows according to the IPsec protocol suite specifications.

Discussion:

Implementations can be grouped by 'external' properties (e.g. software vs. hardware implementations) but more important is the subtle differences that implementations may have with relation to the IPsec Protocol Suite. Not all implementations will cover all RFC's that encompass the IPsec Protocol Suite, but the majority will support a large subset of features described in the suite, nor will all implementations utilize all of the cryptographic functions listed in the RFC's.

In that context, any implementation, that supports basic IP layer security services as described in the IPsec protocol suite shall be called an IPsec Device.

Issues:

Due to the fragmented nature of the IPsec Protocol Suite RFC's, it is possible that IPsec implementations will not be able to interoperate. Therefore it is important to know which features and options are implemented in the IPsec Device.

See Also:

IPsec

7.6.1 Initiator

Definition:

An IPsec device which starts the negotiation of IKE Phase 1 and Phase 2 tunnels.

Discussion:

When a traffic flow is offered at an IPsec device and it is determined that the flow must be protected, but there is no IPsec tunnel to send the traffic through, it is the responsibility of the IPsec device to start a negotiation process that will instantiate the IPsec tunnel. This process will establish an IKE Phase 1 SA and one, or more likely, a pair IKE phase 2 SA's, eventually resulting in secured data transport. The device that takes the action to start this negotiation process will be called an Initiator.

Issues:

IPsec devices/implementations can be both an initiator as well as a responder. The distinction is useful from a test perspective.

See Also:

Responder, IKE, IPsec

7.6.2 Responder

Definition:

An IPsec devices which replies to incoming IKE Phase 1 and Phase 2 tunnel requests and process these messages in order to establish a tunnel.

Discussion:

When an initiator attempts to establish SA's with another IPsec device, this peer will need to evaluate the proposals made by the initiator and either accept or deny them. In the former case, the traffic flow will be decrypted according to the negotiated parameters. Such a device will be called a Responder.

Issues:

IPsec devices/implementations can usually be both an initiator as well as a responder. The distinction is useful from a test perspective.

See Also:

Initiator, IKE

7.6.3 IPsec Client

Definition:

IPsec Devices that will only act as an Initiator.

Discussion:

In some situations it is not needed or prefered to have an IPsec device respond to an inbound tunnel request. In the case of e.g. road warriors or home office scenarios the only property needed from the IPsec device is the ability to securely connect to a remote private network. The IPsec Client will set up one or more Tunnels to an IPsec Server on the network that needs to be accessed and to provide the required security services. An IPsec client will silently drop and ignore any inbound tunnel requests. IPsec clients are generally used to connect remote users in a secure fashion over the Internet to a private network.

Issues:

N/A

See Also:

IPsec device, IPsec Server, Initiator, Responder

7.6.4 IPsec Server

Definition:

IPsec Devices that can both act as an Initiator as well as a Responder.

Discussion:

IPsec Servers are mostly positioned at private network edges and provide several functions :

Responds to tunnel setup request from IPsec Clients.

Responds to tunnel setup request from other IPsec devices (Initiators).

Initiate tunnels to other IPsec servers inside or outside the private network.

Issues:

IPsec Servers are also sometimes referred to as 'VPN Concentrators'.

See Also:

IPsec Device, IPsec Client, Initiator, Responder

7.7 Tunnels

The term "tunnel" is often used in a variety of contexts. To avoid any discrepancies, in this document, the following distinctions have been defined :

7.7.1 IKE Tunnel

Definition:

A single Phase 1 IKE SA.

Discussion:

An IKE Tunnel between IPsec devices facilitates a mechanism for secure negotiation of Phase 1 properties and Phase 2 SA's needed for protected data transport. The initiator may choose which mode to start the negotiation of the IKE Tunnel in. This can be either main mode or aggressive mode.

Issues:

Also referred to as an ISAKMP SA or IKE SA or Phase 1 Tunnel.

See Also:

Tunnel, IPsec Tunnel, Security Association, IKE, IKE Phase 1

7.7.2 IPsec Tunnel

Definition:

One or more Phase 2 SA's that are negotiated in conjunction with an IKE Tunnel.

Discussion:

In the case of simplex communication, a single phase 2 SA.

In the more likely case where bidirectional communication is needed it is a pair (2) Phase 2 SA's. The two SA's are used to secure inbound and outbound traffic.

Not in all situations is it required to have an existing IKE Tunnel in order to negotiate IPsec Tunnel properties and parameters. Manually keyed tunnels allow the set up of IPsec Tunnels without the need of the IKE protocol.

Issues:

If not explicitly specified it SHALL be assumed that an IPsec Tunnel is a pair (2) Phase 2 SA's.

Also referred to as a Phase 2 Tunnel or a Phase 2 SA (may be multiple).

See Also:

Tunnel, IKE Tunnel, Security Association, IKE, IKE Phase 2

7.7.3 Tunnel

Definition:

The combination of an IKE Tunnel and an IPsec Tunnel

Discussion:

In the majority of the cases IPsec is used to protect bidirectional traffic flows. Unless stated otherwise a 'Tunnel' will be defined as a single Phase 1 SA and two Phase 2 SA's that are associated with the Phase 1 SA.

Issues:

If other than a single Phase 2 SA, for each direction, have been negotiated through a single IKE Tunnel, then this specific ratio MUST be mentioned and the term 'Tunnel' MUST NOT be used in this context."

See Also:

IKE Tunnel, IPsec Tunnel

7.7.4 Configured Tunnel

Definition:

A tunnel that is present in the IPsec device's configuration but does not have any entries in the SADB (Security Association DataBase) i.e. SA's.

Discussion:

Several steps are required before a Tunnel can be used to actually transport traffic. The very first step is to configure the tunnel in the IPsec device. In that way packet classification can make a decision if it is required to start negotiating SA's. At this time there are no SA's associated with the Tunnel and no traffic is going through the IPsec device that matches the Selectors, which would instantiate the Tunnel.

A Configured Tunnel is also a tunnel that has relinquished all it's SA's and is not transmitting data anymore. To be more specific, when an Established or an Active Tunnel is terminated due to either an administrative action or an IKE event that deactivated the tunnel, the tunnel will be back in a configured state.

Issues:

N/A

See Also:

Tunnel, Established Tunnel, Active Tunnel

7.7.5 Established Tunnel

Definition:

A Tunnel that has completed Phase 1 and Phase 2 SA negotiations but is otherwise idle.

Discussion:

A second step needed to ensure that a Tunnel can transport data is to complete the Phase 1 and Phase 2 negotiations. After the packet classification process has asserted that a packet requires security services, the negotation is started to obtain both Phase 1 and Phase 2 SA's. After this is completed the tunnel is called 'Established'. Note that at this time there is still no traffic flowing through the Tunnel. Just enough packet(s) have been sent to the IPsec device that matched the selectors and triggered the Tunnel setup. This may also be acomplished by an administrative command to connect the Tunnel, in which case the Tunnel is not triggered by any positive packet classification.

Issues:

In the case of manually keyed tunnels, there is no distinction between a Configured Tunnel or an Established Tunnel since there is no negotiation required with these type of Tunnels and the Tunnel is Established at time of Configuration since all keying information is known at that point.

See Also:

Tunnel, Configured Tunnel, Active Tunnel

7.7.6 Active Tunnel

Definition:

A tunnel that has completed Phase 1 and Phase 2 SA negotiations and is forwarding data.

Discussion:

When a Tunnel is Established and it is transporting traffic, the tunnel is called 'Active'.

Issues:

The distinction between an Active Tunnel and Configured/Established Tunnel is made in the context of manual keyed Tunnels. In this case it would be possible to have an Established tunnel on an IPsec device which has no counterpart on it's corresponding peer. This will lead to encrypted traffic flows which will be discarded on the receiving peer. Only if both peers have an Established Tunnel that shows evidence of traffic transport, it may be called an Active Tunnel.

See Also:

Tunnel, Configured Tunnel, Established Tunnel

7.8 Iterated Tunnels

Iterated Tunnels are a bundle of transport and/or tunnel mode SA's. The bundles are divided into two major groups :

7.8.1 Nested Tunnels

Definition:

An SA bundle consisting of two or more 'tunnel mode' SA's.

Discussion:

The process of nesting tunnels can theoretically be repeated multiple times (for example, tunnels can be many levels deep), but for all practical purposes, most implementations limit the level of nesting. Nested tunnels can use a mix of AH and ESP encapsulated traffic.

```
[GW1] --- [GW2] ---- [IP CLOUD] ---- [GW3] --- [GW4]
  |         |                          |         |
  |         |                          |         |
  |         +----{SA1 (ESP tunnel)}----+         |
  |                                              |
  +--------------{SA2 (AH tunnel)}---------------+
```

In the IP Cloud a packet would have a format like this :
[IP{2,3}][ESP][IP{1,4}][AH][IP][PAYLOAD][ESP TRAILER][ESP AUTH]

Nested tunnels can be deployed to provide additional security on already secured traffic. A typical example of this would be that the inner gateways (GW2 and GW3) are securing traffic between two branch offices and the outer gateways (GW1 & GW4) add an additional layer of security between departments within those branch offices.

Issues:

N/A

See Also:

Transport Adjacency, IPsec Tunnel

7.8.2 **Transport Adjacency**

Definition:

An SA bundle consisting of two or more transport mode SA's.

Discussion:

Transport adjacency is a form of tunnel nesting. In this case two or more transport mode IPsec tunnels are set side by side to enhance applied security properties.

Transport adjacency can be used with a mix of AH and ESP tunnels although some combinations are not preferred. If AH and ESP are mixed, the ESP tunnel should always encapsulate the AH tunnel. The reverse combination is a valid combination but doesn't make cryptographical sense.

```
[GW1] --- [GW2] ---- [IP CLOUD] ---- [GW3] --- [GW4]
 | |                                  |         |
 | |                                  |         |
 | +-------{SA1 (ESP transport)}--------+         |
 |                                              |
 +-------------{SA2 (AH transport)}--------------+
```

In the IP Cloud a packet would have a format like this :
[IP][ESP][AH][PAYLOAD][ESP TRAILER][ESP AUTH]

Issues:

This is rarely used in the way it is depicted. It is more common, but still not likely, that SA's are established from different gateways as depicted in the Nested Tunnels figure. The packet format in the IP Cloud would remain unchanged.

See Also:

Nested Tunnels, IPsec Tunnel

7.9 Transform protocols

Definition:

Encryption and authentication algorithms that provide cryptograhical services to the IPsec Protocols.

Discussion:

Some algorithms run significantly slower than others. For example, TripleDES encryption is one third as fast as DES encryption.

Issues:

N/A

See Also:

Authentication protocols, Encryption protocols

7.9.1 Authentication Protocols

Definition:

Algorithms which provide data integrity and data source authentication.

Discussion:

Authentication protocols provide no confidentiality. Commonly used authentication algorithms/protocols are:

* MD5-HMAC
* SHA-HMAC

* AES-HMAC

Issues:

N/A

See Also:

Transform protocols, Encryption protocols

7.9.2 Encryption Protocols

Definition:

Algorithms which provide data confidentiality.

Discussion:

Encryption protocols provide no authentication. Commonly used encryption algorithms/protocols are:

* NULL encryption
* DES-CBC
* 3DES-CBC
* AES-CBC

Issues:

Null option is a valid encryption mechanism although it reverts to use of IPsec back to message authenticity but only for upper layer protocols.

DES has been officially deprecated by NIST, though it is still mandated by the IPsec framework and is still commonly implemented and used due to it's speed advantage over 3DES. AES will be the likely successor of 3DES due to its superior encryption and its single oparation nature which translates into a speed advantage.

See Also:

Transform protocols, Authentication protocols

7.10 IPsec Protocols

Definition:

A suite of protocols which provide a framework of open standards that provides data confidentiality, data integrity, and data authenticity between participating peers at the IP layer. The IPsec protocol suite set of standards is documented in [RFC2401] through [RFC2412] and [RFC2451].

Discussion:

The IPsec Protocol suite is modular and forward compatible. The protocols that comprise the IPsec protocol suite can be replaced with new versions of those protocols as the older versions become obsolete. For example, IKEv2 will soon replace IKEv1.

Issues:

N/A

See Also:

AH, ESP

7.10.1 Authentication Header (AH)

Definition:

Provides authentication and data integrity (including replay protection) security services [RFC2402].

Discussion:

The AH protocol supports both modes of operation; tunnel mode and transport mode. If AH is employed in tunnel mode, portions of the outer IP header are given protection, as well as all of the tunneled IP packet (that is, all of the inner IP header is protected as are the higher-layer protocols). In the case of AH in transport mode, all upper-layer information is protected, and all fields in the IPv4 header excluding the fields typically are modified in transit.

```
Original IPv4 packet :
[IP ORIG][L4 HDR][PAYLOAD]
In transport mode :
[IP ORIG][AH][L4 HDR][PAYLOAD]
In tunnel mode :
[IP NEW][AH][IP ORIG][L4 HDR][PAYLOAD]
```

Issues:

AH is rarely used to secure traffic over the Internet.

See Also:

Transform protocols, IPsec protocols, Encapsulated Security Payload

7.10.2 Encapsulated Security Payload (ESP)

Definition:

Provides three essential components needed for secure data exchange: authentication, integrity (including replay protection) and confidentiality as defined in [RFC2406].

Discussion:

The ESP protocol supports both modes of operation i.e. tunnel mode and transport mode. If ESP is employed in tunnel mode, the protection is afforded only to the tunneled packet, not to the outer header. In the case of ESP in transport mode, security services are provided only for the higher-layer protocols, not for the IP header.

```
Original IPv4 packet :
[IP ORIG][L4 HDR][PAYLOAD]
In transport mode :
[IP ORIG][ESP][L4 HDR][PAYLOAD][ESP TRAILER][ESP AUTH]
In tunnel mode :
[IP NEW][ESP][IP ORIG][L4 HDR][PAYLOAD][ESP TRAILER][ESP AUTH]
```

Issues:

N/A

See Also:

Transform protocols, IPsec protocols, Authentication Header

7.11 NAT Traversal (NAT-T)

Definition:

The capability to support IPsec functionality in the presence of NAT devices.

Discussion:

NAT-Traversal requires some modifications to IKE as defined in [I-D.ietf-ipsec-nat-t-ike]. Specifically, in phase 1, it requires detecting if the other end supports NAT-Traversal, and detecting if there are one or more NAT instances along the path from host to host. In IKE Quick Mode, there is a need to negotiate the use of UDP encapsulated IPsec packets.

NAT-T also describes how to transmit the original source and destination addresses to the corresponding IPsec Device. The original source and destination addresses are used in transport mode to incrementally update the TCP/IP checksums so that they will match after the NAT transform (The NAT cannot do this, because the TCP/IP checksum is inside the UDP encapsulated IPsec packet).

Issues:

N/A

See Also:

IKE, ISAKMP, IPsec Device

7.12 IP Compression

Definition:

A mechanism as defined in [RFC2393] that reduces the size of the payload that needs to be encrypted.

Discussion:

IP payload compression is a protocol to reduce the size of IP datagrams. This protocol will increase the overall communication performance between a pair of communicating hosts/gateways ("nodes") by compressing the datagrams, provided the nodes have sufficient computation power, through either CPU capacity or a compression coprocessor, and the communication is over slow or congested links.

IP payload compression is especially useful when encryption is applied to IP datagrams. Encrypting the IP datagram causes the data to be random in nature, rendering compression at lower protocol layers (e.g., PPP Compression Control Protocol [RFC1962]) ineffective. If both compression and encryption are required, compression must be applied before encryption.

Issues:

N/A

See Also:

IKE, ISAKMP, IPsec Device

7.13 Security Context

Definition:

A security context is a collection of security parameters that describe the characteristics of the path that a tunnel will take, all of the tunnel parameters and the effects it has on the underlying protected traffic. Security Context encompasses protocol suite and security policy.

Discussion:

In order to fairly compare multiple IPsec devices it is imperative that an accurate overview is given of all security parameters that were used to establish tunnels and to secure the traffic between protected networks. Security Context is not a metric; it is included to accurately reflect the test environment variables when reporting the methodology results. To avoid listing too much information when reporting metrics, we have divided the security context into an IKE context and an IPsec context.

When merely discussing the behavior of traffic flows through IPsec devices, an IPsec context MUST be provided. In other cases the scope of a discussion or report may focus on a more broad set of behavioral characteristics of the IPsec device, the both and IPsec and an IKE context MUST be provided.

The IPsec context MUST consist of the following elements:

* Number of IPsec tunnels

* IPsec tunnels per IKE tunnel (IKE/IPsec tunnel ratio)

* IPsec protocol

* IPsec mode (tunnel or transport)

* Authentication protocol used by IPsec

* Encryption protocol used by IPsec (if applicable)

* IPsec SA lifetime (traffic and time based)

The IPsec Context MAY also list:

* Selectors

* Fragmentation handling

The IKE Context MUST consist of the following elements:

* Number of IKE tunnels.

* Authentication protocol used by IKE

* Encryption protocol used by IKE

* Key exchange mechanism (pre-shared key, certificate authority, etc ...)

* Key size (if applicable)

* Diffie-Hellman group

* IKE SA lifetime (time based)

* Keepalive or DPD values as defined in [I-D.ietf-ipsec-dpd]

* IP Compression [RFC2393]

* PFS Diffie-Hellman group

The IKE context MAY also list:

* Phase 1 mode (main or aggressive)

* Available bandwidth and latency to Certificate Authority server (if applicable)

Issues:

A Security Context will be an important element in describing the environment where protected traffic is traveling through.

See Also:

IPsec Protocols, Transform Protocols, IKE Phase 1, IKE phase 2, Selectors, IPsec Tunnel

8. Framesizes

8.1 Layer3 clear framesize

Definition:

The total size of the unencrypted L3 PDU.

Discussion:

In relation to IPsec this is the size of the IP header and its payload. It SHALL NOT include any encapsulations that MAY be applied before the PDU is processed for encryption.

For example: 46 bytes PDU = 20 bytes IP header + 26 bytes payload.

Measurement Units:

Bytes

Issues:

N/A

See Also:

Layer3 Encrypted Framesize, Layer2 Clear Framesize, Layer2 Encrypted Framesize.

8.2 Layer3 encrypted framesize

Definition:

The total size of the encrypted L3 PDU.

Discussion:

The size of the IP packet and itÆs payload after encapsulations MAY be applied and the PDU is being processed by the transform.

For example, after a tunnel mode ESP 3DES/SHA1 transform has been applied an unencrypted or clear layer3 framesize of 46 bytes Becomes 96 bytes:

```
20 bytes outer IP header (tunnel mode)
4 bytes SPI (ESP header)
4 bytes Sequence (ESP Header)
8 bytes IV (IOS ESP-3DES)
46 bytes payload
0 bytes pad (ESP-3DES 64 bit)
1 byte Pad length (ESP Trailer)
1 byte Next Header (ESP Trailer)
12 bytes ESP-HMAC SHA1 96 digest
```

Measurement Units:

Bytes

Issues:

N/A

See Also:

Layer3 Clear Framesize, Layer2 Clear Framesize, Layer2 Encrypted Framesize.

8.3 Layer2 clear framesize

Definition:

The total size of the unencrypted L2 PDU.

Discussion:

This is the Layer 3 clear framesize plus all the layer2 overhead. In the case of Ethernet this would be 18 bytes.

For example, a 46 byte Layer3 clear framesize packet would become 64 Bytes after Ethernet Layer2 overhead is added:

    6 bytes destination mac address
    6 bytes source mac address
    2 bytes length/type field
    46 bytes layer3 (IP) payload
    4 bytes FCS

Measurement Units:

Bytes

Issues:

If it is not mentioned explicitly what kind of framesize is used, the layer2 clear framesize will be the default.

See Also:

Layer3 clear framesize, Layer2 encrypted framesize, Layer2 encrypted framesize.

8.4 Layer2 encrypted framesize

Definition:

The total size of the encrypted L2 PDU.

Discussion:

This is the Layer 3 encrypted framesize plus all the layer2 overhead. In the case of Ethernet this would be 18 bytes.

For example, a 96 byte Layer3 encrypted framesize packet would become 114 bytes after Ethernet Layer2 overhead is added:

    6 bytes destination mac address
    6 bytes source mac address
    2 bytes length/type field
    96 bytes layer3 (IPsec) payload
    4 bytes FCS

Measurement Units:

Bytes

Issues:

N/A

See Also:

Layer3 Clear Framesize, Layer3 Encrypted Framesize, Layer2 Clear Framesize

9. Performance Metrics

9.1 Tunnels Per Second (TPS)

Definition:

The measurement unit for the Tunnel Setup Rate tests. The rate that Tunnels are established per second.

Discussion:

According to [RFC2401] two tunnels cannot be established between the same gateways with the same selectors. This is to prevent overlapping tunnels. If overlapping tunnels are attempted, the error will take longer than if the tunnel setup was successful. For this reason, a unique pair of selector sets are required for TPS testing.

Issues:

A unique pair of selector sets are required for TPS testing.

See Also:

Tunnel Setup Rate Behavior, Tunnel Setup Rate, IKE Setup Rate, IPsec Setup Rate

9.2 Tunnel Rekeys Per Seconds (TRPS)

Definition:

A metric that quantifies the number of IKE or IPsec Tunnel rekey's per seconds a DUT can correctly process.

Discussion:

This metric will be will be primary used with Tunnel Rekey behavior tests.

TRPS will provide a metric used to see system behavior under stressful conditions where large volumes of tunnels are being rekeyed at the same time or in a short timespan.

Issues:

N/A

See Also:

Tunnel Rekey; Phase 1 Rekey Rate, Phase 2 Rekey Rate

9.3 Tunnel Attempts Per Second (TAPS)

Definition:

A metric that quantifies the number of successful and unsuccessful tunnel (both Phase 1 or Phase 2) establishment requests per second.

Discussion:

This metric can be used to measure IKE DOS Resilience behavior test.

TAPS provides an important metric to validate the stability of a platform, if stressed with valid (large number of IPsec tunnel establishments per seconds or TPS) or invalid (IKE DOS attacks of any style) tunnel establishment requests.

Issues:

If the TAPS increases, the TPS usually decreases, due to burdening of the DUT with the DOS attack traffic.

10. Test Definitions

10.1 Throughput

10.1.1 Tunnel Throughput

Definition:

The maximum rate through an IPsec tunnel at which none of the offered frames are dropped by the device under test.

Discussion:

The IPsec Tunnel Throughput is almost identically defined as Throughput in [RFC1242], section 3.17. The only difference is that the throughput is measured with a traffic flow getting encrypted and decrypted by an IPsec device. IPsec Tunnel Throughput is an end-to-end measurement.

The metric can be represented in two variantions depending on where measurement is taken in the SUT. One can look at throughput from a cleartext point of view i.e. find the maximum rate where clearpackets no longer get dropped. This resulting rate can be recalculated with an encrypted framesize to represent the encryption throughput rate. The latter is the preferred method of representation.

Measurement Units:

Packets per seconds (pps), Mbps

Issues:

N/A

See Also:

IPsec Encryption Throughput, IPsec Decryption Throughput

10.1.2 IPsec Encryption Throughput

Definition:

The maximum encryption rate through an IPsec tunnel at which none of the offered cleartext frames are dropped by the device under test.

Discussion:

Since encryption throughput is not necessarily equal to the decryption throughput, both of the forwarding rates must be measured independently. The independent forwarding rates have to measured with the help of an IPsec aware test device that can originate and terminate IPsec and IKE tunnels. As defined in [RFC1242], measurements should be taken with an assortment of frame sizes.

Measurement Units:

Packets per seconds (pps), Mbps

Issues:

N/A

See Also:

IPsec Tunnel Throughput, IPsec Decryption Throughput

10.1.3 IPsec Decryption Throughput

Definition:

The maximum decryption rate through an IPsec tunnel at which none of the offered encrypted frames are dropped by the device under test.

Discussion:

Since encryption throughput is not necessarily equal to the decryption throughput, both of the forwarding rates must be measured independently.

The independent forwarding rates have to be measured with the help of an IPsec aware test device that can originate and terminate IPsec and IKE tunnels. As defined in [RFC1242], measurements should be taken with an assortment of frame sizes.

Measurement Units:

Packets per seconds (pps), Mbps

Issues:

Recommended test frame sizes will be addressed in future methodology document.

See Also:

IPsec Tunnel Throughput, IPsec Encryption Throughput

10.2 Latency

10.2.1 Tunnel Latency

Definition:

Time required to propagate a cleartext frame from the input interface of an initiator, through an IPsec Tunnel, to the output interface of the responder.

Discussion:

The Tunnel Latency is the time interval starting when the end of the first bit of the cleartext frame reaches the input interface of the initiator and ending when the start of the first bit of the same cleartext frame is detected on the output interface of the responder. The frame has passed through an IPsec Tunnel between an initiator and a responder and has been through an encryption and decryption cycle.

Measurement Units:

Time units with enough precision to reflect latency measurement.

Issues:

N/A

See Also:

IPsec Tunnel Encryption Latency, IPsec Tunnel Decryption Latency

10.2.2 IPsec Tunnel Encryption Latency

Definition:

The IPsec Tunnel Encryption Latency is the time interval starting when the end of the first bit of the cleartext frame reaches the input interface, through an IPsec tunnel, and ending when the start of the first bit of the encrypted output frame is seen on the output interface.

Discussion:

IPsec Tunnel Encryption latency is the latency introduced when encrypting traffic through an IPsec tunnel.

Like encryption/decryption throughput, it is not always the case that encryption latency equals the decryption latency. Therefore a distinction between the two has to be made in order to get a more accurate view of where the latency is the most pronounced.

The independent encryption/decryption latencies have to be measured with the help of an IPsec aware test device that can originate and terminate IPsec and IKE tunnels. As defined in [RFC1242], measurements should be taken with an assortment of frame sizes.

Measurement Units:

Time units with enough precision to reflect latency measurement.

Issues:

N/A

See Also:

IPsec Tunnel Latency, IPsec Tunnel Decryption Latency

10.2.3 IPsec Tunnel Decryption Latency

Definition:

The IPsec Tunnel decryption Latency is the time interval starting when the end of the first bit of the encrypted frame reaches the input interface, through an IPsec tunnel, and ending when the start of the first bit of the decrypted output frame is seen on the output interface.

Discussion:

IPsec Tunnel decryption latency is the latency introduced when decrypting traffic through an IPsec tunnel. Like encryption/decryption throughput, it is not always the case that encryption latency equals the decryption latency. Therefore a distinction between the two has to be made in order to get a more accurate view of where the latency is the most pronounced.

The independent encryption/decryption latencies have to be measured with the help of an IPsec aware test device that can originate and terminate IPsec and IKE tunnels. As defined in [RFC1242], measurements should be taken with an assortment of frame sizes.

Measurement Units:

Time units with enough precision to reflect latency measurement.

Issues:

N/A

See Also:

IPsec Tunnel Latency, IPsec Tunnel Encryption Latency

10.2.4 Time To First Packet

Definition:

The Time To First Packet (TTFP) is the time required process an cleartext packet when no tunnel is present.

Discussion:

The TTFP addresses the issue of responsiveness of an IPsec device by looking how long it take to transmit a packet over a not yet established tunnel path. The TTFP MUST include the time to set up the tunnel, triggered by the traffic flow (both phase 1 and phase 2 setup times are included) and the time it takes to encrypt and decrypt the packet on a corresponding peer. In short it is the tunnel setup time plus the propagation delay of the packet through the Tunnel.

It must be noted that it is highly unlikely that the first packet of the traffic flow will be the packet that will be used to measure the TTFP. There MAY be several protocol layers in the stack before the tunnel is formed and the traffic is forwarded, hence several packets COULD be lost during negotiation, for example, ARP and/or IKE.

Measurement Units:

Time units with enough precision to reflect a TTFP measurement.

Issues:

N/A

10.3 Frame Loss

10.3.1 IPsec Tunnel Frame Loss

Definition:

Percentage of cleartext frames that should have been forwarded through a Tunnel under steady state (constant) load but were dropped before encryption or after decryption.

Discussion:

The IPsec Tunnel Frame Loss is almost identically defined as Frame Loss Rate in [RFC1242], section 3.6. The only difference is that the IPsec Tunnel Frame Loss Rate is measured with a traffic flow getting encrypted and decrypted by an IPsec device. IPsec Tunnel Frame Loss Rate is an end-to-end measurement.

Measurement Units:

Percent (%)

Issues:

N/A

See Also:

IPsec Tunnel Encryption Frame Loss, IPsec Tunnel Decryption Frame Loss

10.3.2 IPsec Tunnel Encryption Frame Loss

Definition:

Percentage of cleartext frames that should have been encrypted through an IPsec tunnel under steady state (constant) load but were dropped.

Discussion:

DUT's will always have an inherent forwarding limitation. This will be more pronounced when IPsec is employed on the DUT. The moment that a Tunnel is established and traffic is offered at a given rate that will flow through that tunnel, there is a possibility that the offered traffic rate at the tunnel is too high to be transported through the IPsec tunnel and not all

cleartext packets will get encrypted. In that case, some percentage of the cleartext traffic will be dropped. This drop percentage is called the IPsec Tunnel Encryption Frame Loss.

Measurement Units:

Percent (%)

Issues:

N/A

See Also:

IPsec Tunnel Frame Loss, IPsec Tunnel Decryption Frame Loss

10.3.3 IPsec Tunnel Decryption Frame Loss

Definition:

Percentage of encrypted frames that should have been decrypted through an IPsec tunnel under steady state (constant) load but were dropped.

Discussion:

A DUT will also have an inherent forwarding limitation when decrypting packets. When established tunnel encrypted traffic is offered at a costant load, there might be a possibility that the IPsec Device that needs to decrypt the traffic will not be able to perfom this action on all of the packets due to limitations of the decryption performance. The percentage of encrypted frames that would get dropped under these conditions is called the IPsec Tunnel Decryption Frame Loss.

Measurement Units:

Percent (%)

Issues:

N/A

See Also:

IPsec Tunnel Frame Loss, IPsec Tunnel Encryption Frame Loss

10.3.4 Phase 2 Rekey Frame Loss

Definition:

Number of frames dropped as a result of an inefficient Phase 2 rekey.

Discussion:

Normal operation of an IPsec device would require that a rekey does not create temporary Frame Loss of a traffic stream that is protected by the Phase 2 SA's. Nevertheless there can be situations where Frame Loss occurs during the rekey process.

This metric should be ideally zero but this may not be the case on IPsec devices where IPsec funtionality is not a core feature.

Measurement Units:

Number of N-octet frames

Issues:

N/A

See Also:

Phase 2 Rekey Rate

10.4 Back-to-back Frames

10.4.1 Tunnel Back-to-back Frames

Definition:

A burst of cleartext frames, offered at a constant load that can be sent through an IPsec tunnel without losing a single cleartext frame after decryption.

Discussion:

The Tunnel Back-to-back Frames is almost identically defined as Back-to-back in [RFC1242], section 3.1. The only difference is that the Tunnel Back-to-back Frames is measured with a traffic

flow getting encrypted and decrypted by an IPsec device. Tunnel Back-to-back Frames is an end-to-end measurement.

Measurement Units:

Number of N-octet frames in burst.

Issues:

Recommended test frame sizes will be addressed in future methodology document.

See Also:

Encryption Back-to-back frames, Decryption Back-to-back frames

10.4.2 Encryption Back-to-back Frames

Definition:

A burst of cleartext frames, offered at a constant load that can be sent through an IPsec tunnel without losing a single encrypted frame.

Discussion:

Encryption back-to-back frames is the measure of the maximum burst size that a device can handle for encrypting traffic that it receives as plaintext. Since it is not necessarily the case that the maximum burst size a DUT can handle for encryption is equal to the maximum burst size a DUT can handle for decryption, both of these capabilities must be measured independently. The encryption back-to-back frame measurement has to be measured with the help of an IPsec aware test device that can decrypt the traffic to determine the validity of the encrypted frames.

Measurement Units:

Number of N-octet frames in burst.

Issues:

Recommended test frame sizes will be addressed in future methodology document.

See Also:

Tunnel Back-to-back frames, Decryption Back-to-back frames

10.4.3 Decryption Back-to-back Frames

Definition:

The number of encrypted frames, offered at a constant load, that can be sent through an IPsec tunnel without losing a single cleartext frame.

Discussion:

Decryption back-to-back frames is the measure of the maximum burst size that a device can handle for decrypting traffic that it receives as encrypted traffic. Since it is not necessarily the case that the maximum burst size a DUT can handle for decryption is equal to the maximum burst size a DUT can handle for encryption, both of these capabilities must be measured independently. The decryption back-to-back frame measurement has to be measured with the help of an IPsec aware test device that can determine the validity of the decrypted frames.

Measurement Units:

Number of N-octet frames in burst.

Issues:

Recommended test frame sizes will be addressed in future methodology document.

See Also:

Tunnel Back-to-back frames, Encryption back-to-back frames

10.5 Tunnel Setup Rate Behavior

10.5.1 Tunnel Setup Rate

Definition:

The maximum number of tunnels (1 IKE SA + 2 IPsec SA's) per second that an IPsec device can successfully establish.

Discussion:

The tunnel setup rate SHOULD be measured at varying number of tunnels on the DUT. Several factors may influence Tunnel Setup Rate, such as: TAPS rate, Background cleartext traffic load on the secure interface, Already established tunnels, Authentication method such as pre-shared keys, RSA-encryption, RSA-signature, DSS Key sizes used (when using RSA/DSS).

Measurement Units:

Tunnels Per Second (TPS)

Issues:

N/A

See Also:

Phase 1 Setup Rate, Phase 2 Setup Rate, Tunnel Rekey

10.5.2 Phase 1 Setup Rate

Definition:

The maximum number of IKE tunnels (1 IKE Phase 1 SA) per second that an IPsec device can be observed to successfully establish.

Discussion:

The Phase 1 Setup Rate is a portion of the Tunnel Setup Rate. In the process of establishing a Tunnel, it is interesting to know what the limiting factor of the IKE Finite State Machine is i.e. is it limited by the Phase 1 processing delays or rather by the Phase 2 processing delays.

Measurement Units:

Tunnels Per Second (TPS)

Issues:

N/A

See Also:

Tunnel Setup Rate, Phase 2 Setup Rate, Tunnel Rekey

10.5.3 Phase 2 Setup Rate

Definition:

The maximum number of IPsec tunnels (2 IKE Phase 2 SA's) per second that a IPsec device can be observed to successfully establish.

Discussion:

The Phase 2 Setup Rate is a portion of the Tunnel Setup Rate. For identical reasons why it is required to quantify the Phase 1 Setup Rate, it is a good practice to know the processing delays involved in setting up a Phase 2 SA for each direction of the protected traffic flow.

Note that once you have the Tunnel Setup Rate and either the Phase 1 or the Phase 2 Setup Rate data, you can extrapolate the unmeasured metric, although it is RECOMMENDED to measure all three metrics.

Measurement Units:

Tunnels Per Second (TPS)

Issues:

N/A

See Also:

Tunnel Setup Rate, Phase 1 Setup Rate, Tunnel Rekey

10.6 Tunnel Rekey

10.6.1 Phase 1 Rekey Rate

Definition:

The number of Phase 1 SA's that can be succesfully re-establish per second.

Discussion:

Although the Phase 1 Rekey Rate has less impact on the forwarding behavior of traffic that requires security services then the Phase 2 Rekey Rate, it can pose a large burden on the CPU or network processor of the IPsec Device. Due to the highly computational nature of a Phase 1 exchange, it may impact the stability of Active Tunnels in the network when the IPsec Device fails to properly rekey an IKE Tunnel.

Measurement Units:

Rekey's per second

Issues:

N/A

See Also:

Phase 2 Rekey Rate

10.6.2 Phase 2 Rekey Rate

Definition:

The number of Phase 2 SA's that can be succesfully re-negotiated per-second.

Discussion:

Although many implementations will usually derive new keying material before the old keys expire, there may still be a period of time where frames get dropped before the phase 2 tunnels are successfully re-established. There may also be some packetloss introduced when the handover of traffic is done from the expired SA to the newly negotiated SA. To measure the phase 2 rekey rate, the measurement will require an IPsec aware test device to act as a responder when negotiating the new phase 2 keying material.

The test methodology report must specify if PFS is enabled in reported security context.

Measurement Units:

Rekey's per second

Issues:

N/A

See Also:

Phase 1 Rekey Rate

10.7 Tunnel Failover Time (TFT)

Definition:

Time required to recover all tunnels on a stanby IPsec device, after a catastrophic failure occurs on the active IPsec device.

Discussion:

Recovery time required to re-establish all tunnels and reroute all traffic on a standby node or other failsafe system after a failure has occurred. Failure can include but are not limited to a catastrophic IPsec Device failure, a encryption engine failure, link outage. The recovery time is delta between the point of failure and the time the first packet is seen on the last restored tunnel on the backup device.

Measurement Units:

Time units with enough precision to reflect Tunnel Failover Time.

Issues:

N/A

10.8 IKE DOS Resilience Rate

Definition:

The IKE Denial Of Service (DOS) Resilience Rate provides a rate of invalid or mismatching IKE tunnels setup attempts at which it is no longer possible to set up a valid IKE tunnel.

Discussion:

The IKE DOS Resilience Rate will provide a metric to how robust and hardened an IPsec device is against malicious attempts to set up a tunnel.

IKE DOS attacks can pose themselves in various forms and do not necessarily have to have a malicious background. It is sufficient to make a typographical error in a shared secret in an IPsec aggregation device to be susceptible to a large number of IKE attempts that need to be turned down. Due to the intense computational nature of an IKE exchange every single IKE tunnel attempt that has to be denied will take a non-negligible time an a CPU in the IPsec device.

Depending on how many of these messages have to be processed, a system might end up in a state that it is only doing key exchanges and burdening the CPU for any other processes that might be running in the IPsec device. At this point it will be no longer possible to process a valid IKE tunnel setup request and thus IKE DOS is in effect.

Measurement Units:

Tunnel Attempts Per Seconds (TAPS)

Issues:

N/A

11. Security Considerations

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

12. Acknowledgements

The authors would like to acknowledge the following individual for their help and participation of the compilation and editing of this document: Debby Stopp, Ixia.

13. Contributors

The authors would like to acknowledge the following individual for their significant help, guidance, and contributions to this document: Paul Hoffman, VPNC, Sunil Kalidindi, Ixia, Brian Talbert, MCI.

14. References

14.1 Normative References

[RFC1242] Bradner, S., "Benchmarking terminology for network interconnection devices", RFC 1242, July 1991.

[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.

[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.

[I-D.ietf-ipsec-ikev2]
Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", Internet-Draft draft-ietf-ipsec-ikev2-17, October 2004.

[I-D.ietf-ipsec-dpd]
Huang, G., Beaulieu, S. and D. Rochefort, "A Traffic-Based Method of Detecting Dead IKE Peers", Internet-Draft draft-ietf-ipsec-dpd-04, October 2003.

[I-D.ietf-ipsec-nat-t-ike]
Kivinen, T., "Negotiation of NAT-Traversal in the IKE", Internet-Draft draft-ietf-ipsec-nat-t-ike-08, February 2004.

[I-D.ietf-ipsec-udp-encaps]
Huttunen, A., "UDP Encapsulation of IPsec Packets", Internet-Draft draft-ietf-ipsec-udp-encaps-09, May 2004.

[I-D.ietf-ipsec-nat-reqts]
Aboba, B. and W. Dixon, "IPsec-NAT Compatibility Requirements", Internet-Draft draft-ietf-ipsec-nat-reqts-06, October 2003.

[I-D.ietf-ipsec-properties]

Krywaniuk, A., "Security Properties of the IPsec Protocol Suite", Internet-Draft draft-ietf-ipsec-properties-02, July 2002.

[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>.

14.2 Informative References

[Designing Network Security]
Kaeo, M., "Designing Network Security", ISBN: 1578700434, Published: May 07, 1999; Copyright: 1999, 1999.

[SKEME] Krawczyk, H., "SKEME: A Versatile Secure Key Exchange Mechanism for Internet", from IEEE Proceedings of the 1996 Symposium on Network and Distributed Systems Security, URI http://www.research.ibm.com/security/skeme.ps, 1996.

Authors' Addresses

Michele Bustos
IXIA
26601 W. Agoura Rd.
Calabasas, CA 91302
US

Phone: +1 (818)444-3244
Email: mbustos@ixiacom.com

Tim Van Herck
Cisco Systems
170 West Tasman Dr.
San Jose, CA 95134-1706
US

Phone: +1 (408)527-2461
Email: herckt@cisco.com

Merike Kaeo
Double Shot Security
520 Washington Blvd #363
Marina Del Rey, CA 90292
US

Phone: +1 (310)866-0165
Email: merike@doubleshotsecurity.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP 11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

Full Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.