secunet

Parallelizing IPsec: switching SMP to 'On' is not even half the way

Steffen Klassert

secunet Security Networks AG

Dresden

June 11 2010

secunet

Table of contents

secunet

The IPsec protocols

Every IPsec implementation must support two protocols.

The IPsec protocols

Every IPsec implementation must support two protocols.

- *IP - Authentication Header (AH)*
  - AH builds a cryptographic checksum over the payload and parts of the header of a network packet.
  - This checksum is appended to the network packet and is used to ensure authenticity of this network packet.

The IPsec protocols

Every IPsec implementation must support two protocols.

- *IP - Authentication Header (AH)*
  - AH builds a cryptographic checksum over the payload and parts of the header of a network packet.
  - This checksum is appended to the network packet and is used to ensure authenticity of this network packet.
- *IP - Encapsulated Security Payload (ESP)*
  - ESP is primary used to encrypt the payload of network packets.
  - A cryptographic checksum can be used to ensure authenticity of the payload, similar to AH.

secunet

ESP modes

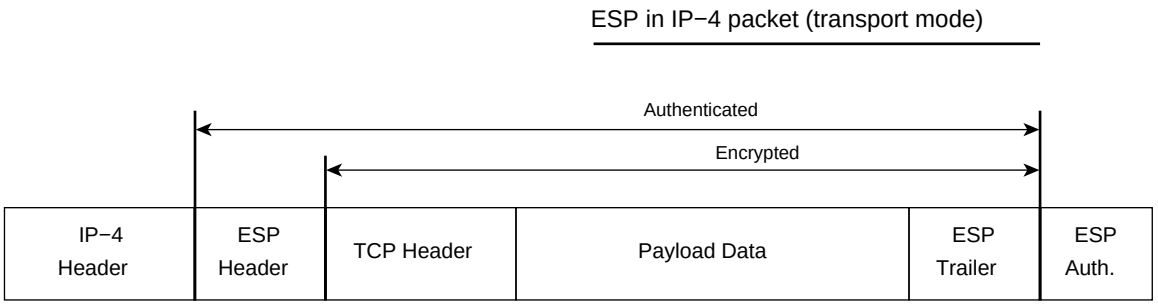
The ESP protocol can be used in several modes.

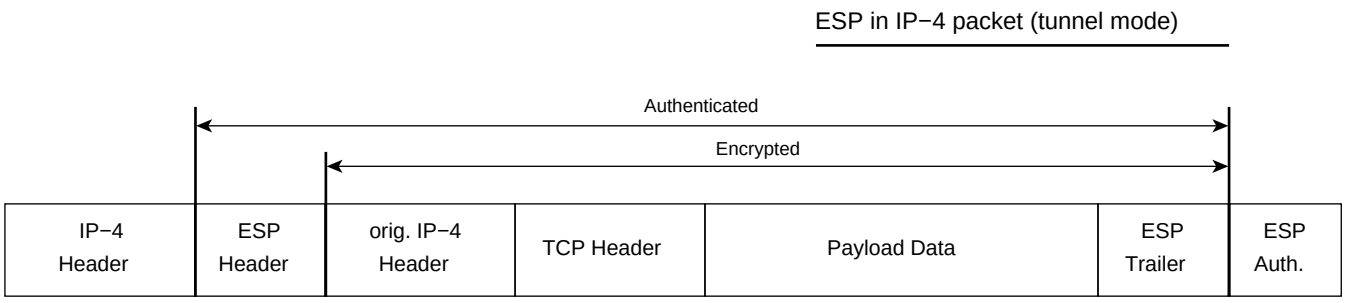
secunet

ESP modes

The ESP protocol can be used in several modes.

- *Transport mode* - Pure layer 4 payload encryption.
- *Tunnel mode* - Encryption for the whole IP packet (payload + IP header).

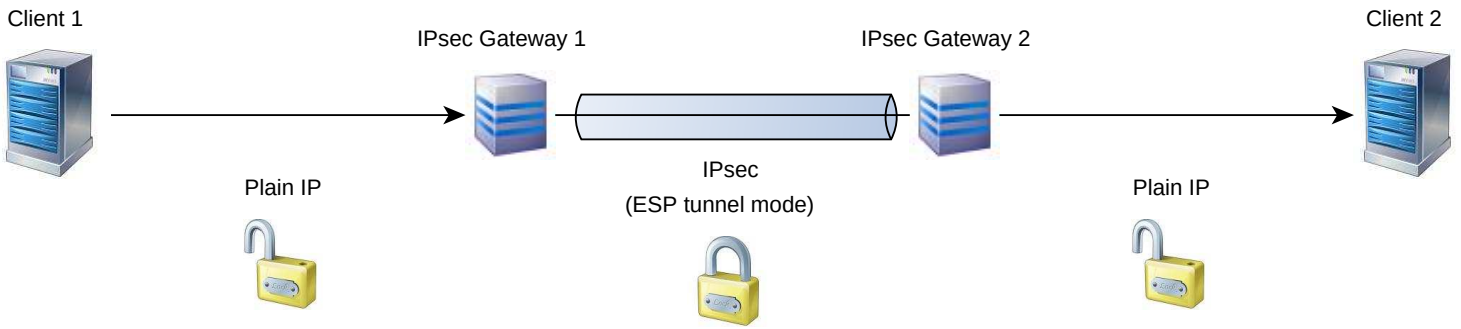




secunet

The Hardware setup and IPsec scenario

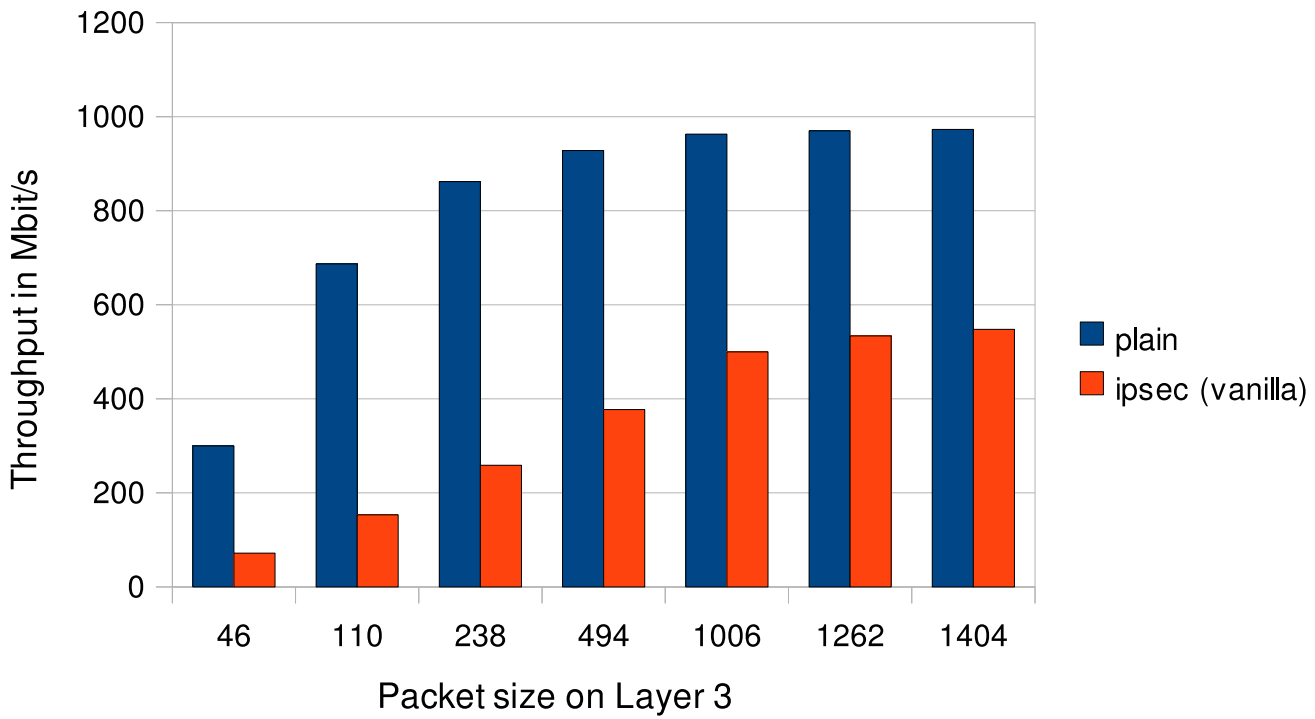
The hardware setup is the simplest possible IPsec VPN scenario, consisting of two IPsec gateways and two clients.



Packet forwarding from client 1 to client 2, unidirectional traffic, one packet flow.

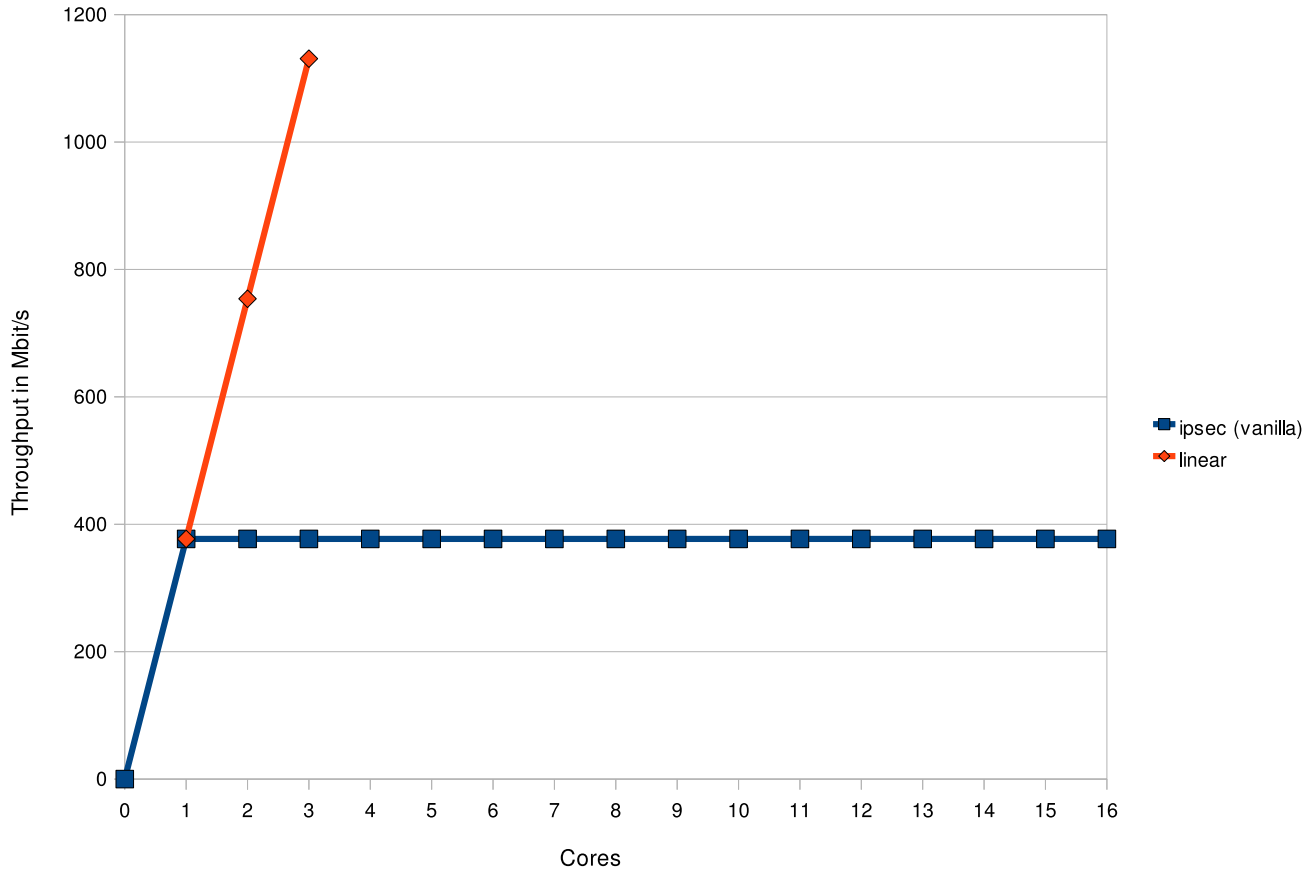
secunet

Plain packet forwarding vs. tunnel mode ESP with cbc-aes192 / hmac-sha1 on a Gbit network.



secunet

IPsec throughput: scaling with the number of cpus



494 byte packets (L3) cbc-aes-192 / hmac-sha1

secunet

IPsec throughput: scaling with the number of cpus

The forward packet path is strictly serialized. I.e. the cpu that drives the interrupt of the receiving NIC does all the work!

secunet

IPsec throughput: scaling with the number of cpus

The forward packet path is strictly serialized. I.e. the cpu that drives the interrupt of the receiving NIC does all the work!

Why?

secunet

IPsec throughput: scaling with the number of cpus

The forward packet path is strictly serialized. I.e. the cpu that drives the interrupt of the receiving NIC does all the work!

Why?

- The upper layer (L4) protocols rely on a certain packet order. The packets must be received in the same order they where sent.
- IPsec adds a sequence number to each packet to notify packet replay attacks.

IPsec throughput: scaling with the number of cpus

The forward packet path is strictly serialized. I.e. the cpu that drives the interrupt of the receiving NIC does all the work!

Why?

- The upper layer (L4) protocols rely on a certain packet order. The packets must be received in the same order they where sent.
- IPsec adds a sequence number to each packet to notify packet replay attacks.

Distributing the received network packets to multiple cpus leads to packet reordering!

secunet

Network parallelization approaches

Due to packet reorder problems, parallelization of the network stack is a highly nontrivial task. Several software, as well as hardware based approaches came up during the last years.

- Multiqueue network devices.
- Receive packet steering.

These techniques do flow based parallelization, i.e. distributing packet flows across the cpus.

No parallelization within the flows to preserve the packet order!

secunet

Flow based parallelization on IPsec

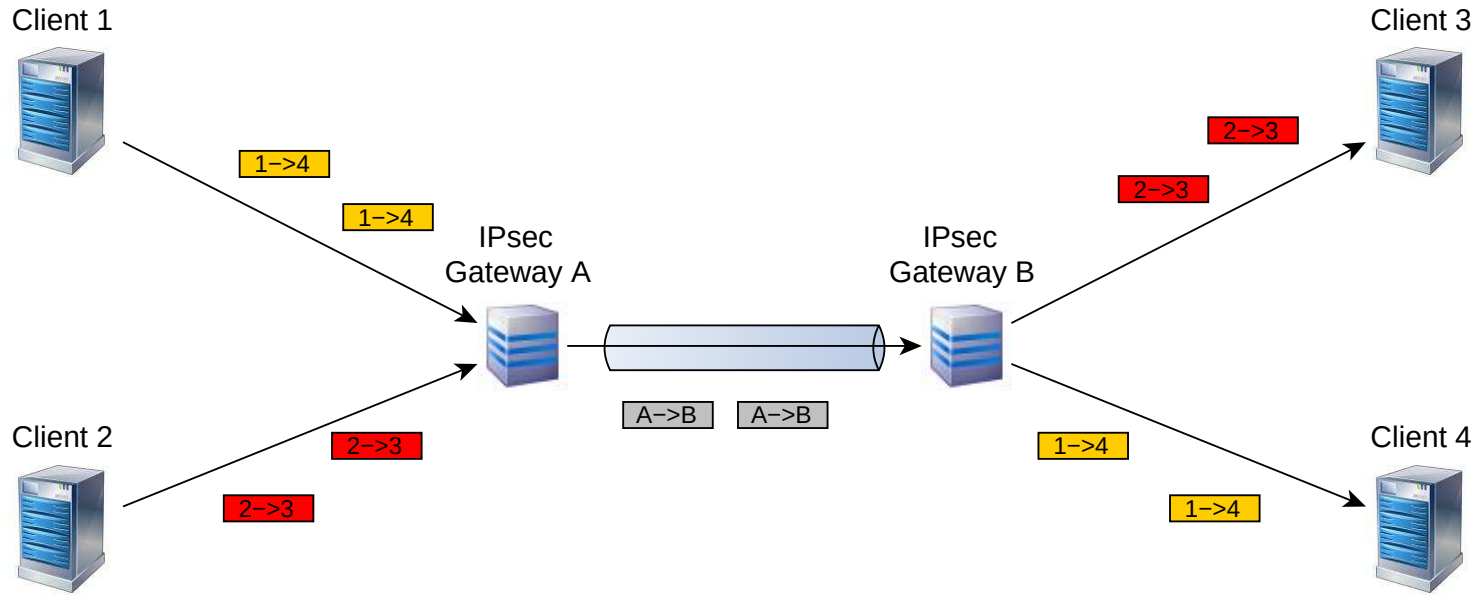
secunet

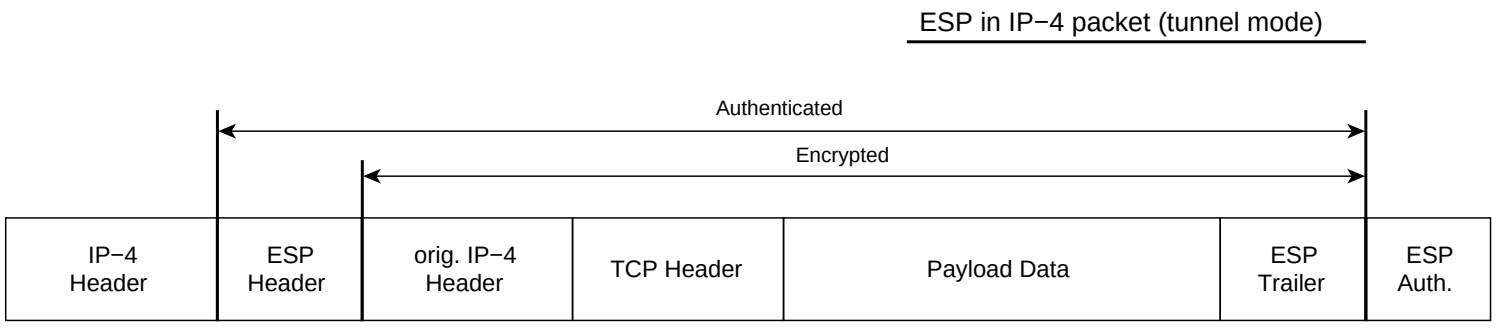
Flow based parallelization on IPsec

Flow based parallelization is only limited useful for tunnel mode ESP.

secunet

Flow based parallelization on IPsec (tunnel mode ESP)





secunet

Requirements of an IPsec parallelization

- **R1:** *It should be possible to distribute cpu intensive codepaths to a given set of cpus.*
- **R2:** *It should be possible to parallelize even within a flow.*
- **R3:** *The parallelization framework must preserve the order of the parallelized network packets. E.g. the packets must leave the parallel codepath in the same order as they entered.*

secunet

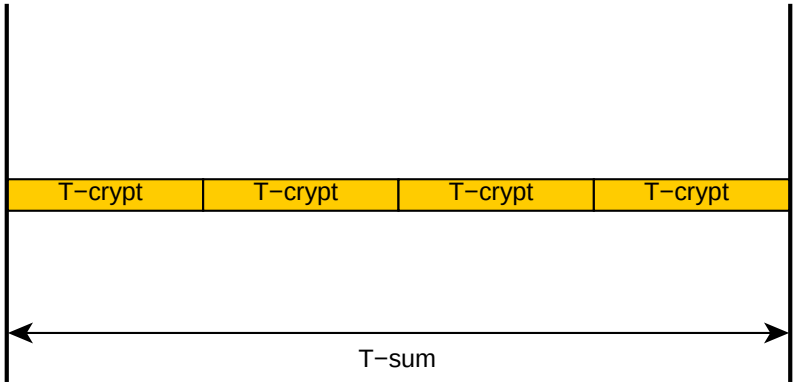
A parallel crypto layer

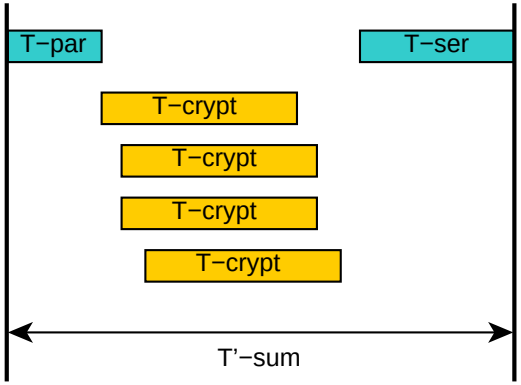
Advantages of a parallel crypto layer:

- The crypto operations are by far the most cpu intensive codepath (R1).
- The crypto layer does not know about the crypto user (ESP), no need to care about the order of the requests within the crypto layer (R2).
  We just have to ensure that the crypto requests leave the crypto layer in the same order as they entered (R3).

secunet

The gain of a crypto layer parallelization





secunet

The gain of a crypto layer parallelization

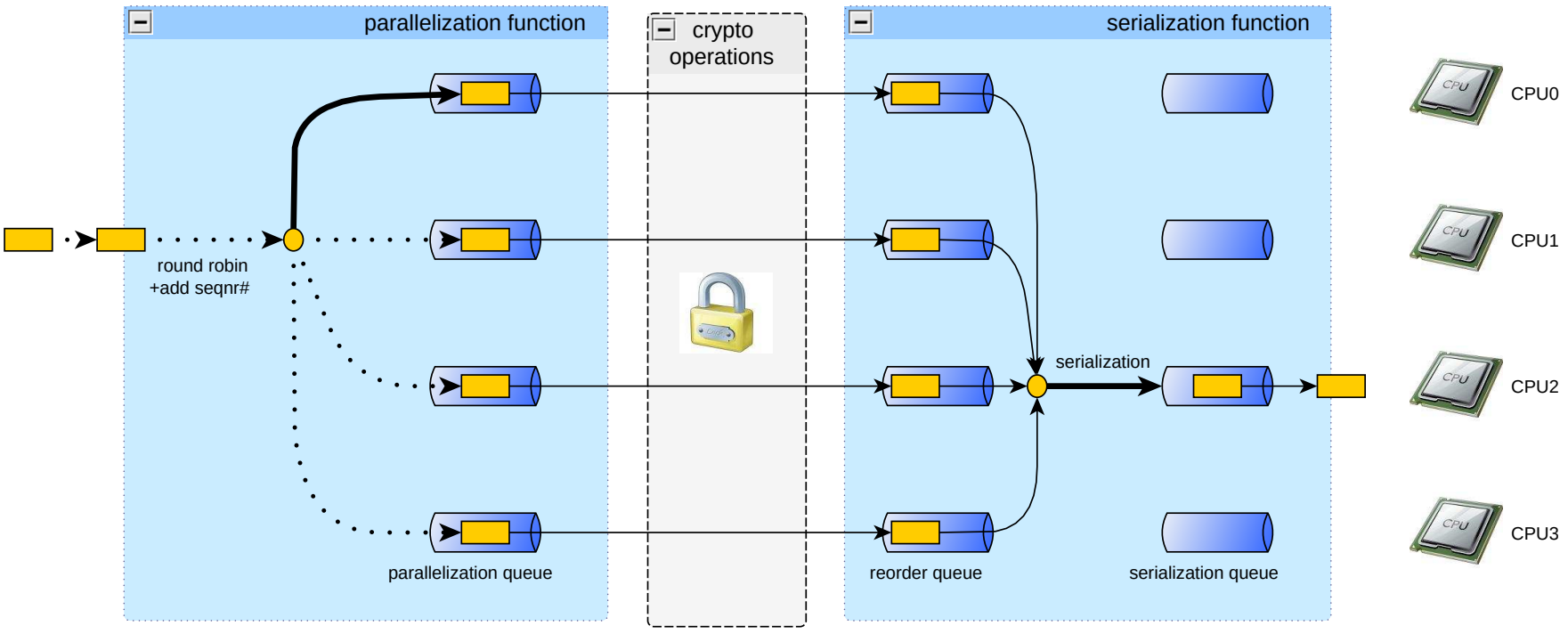
- Large crypto requests (e.g. big network packets) benefit well.
- Very cpu intensive crypto algorithms benefit well.

secunet

The padata/pcrypt framework

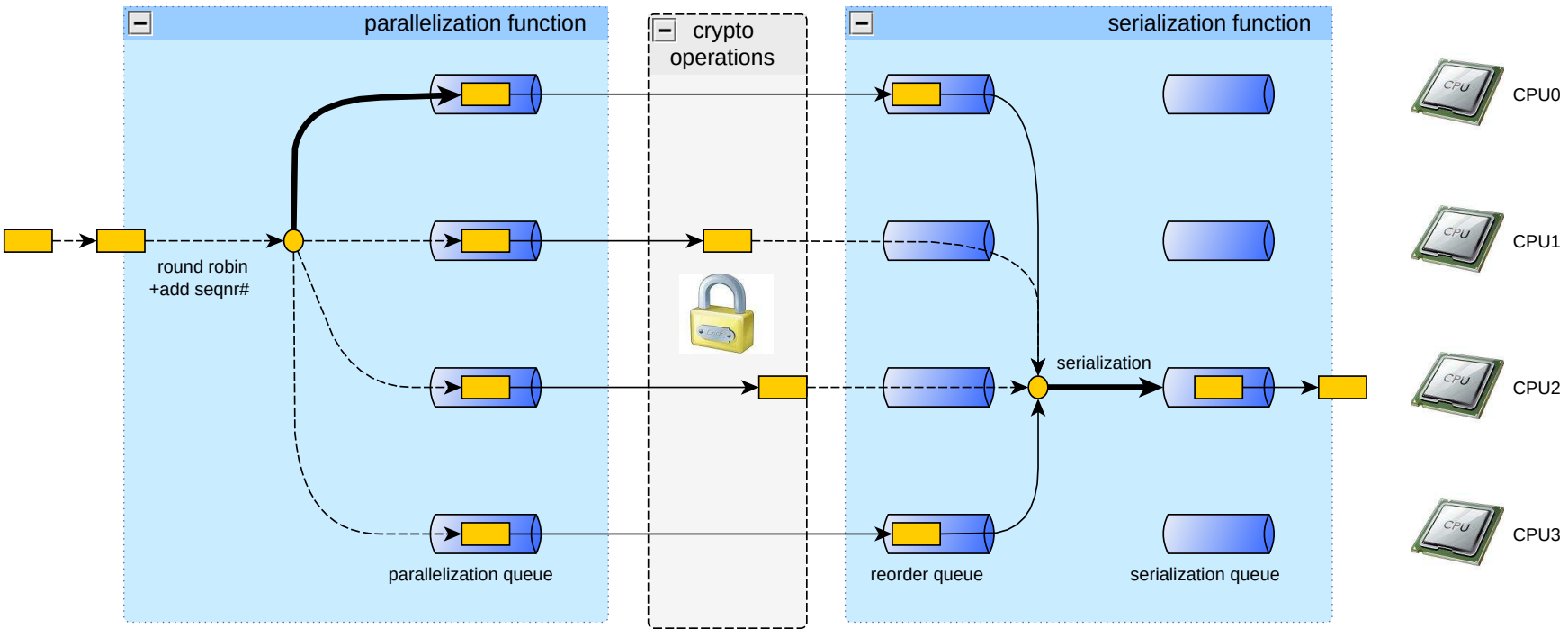
secunet

The padata/pcrypt framework



secunet

The padata/pcrypt framework



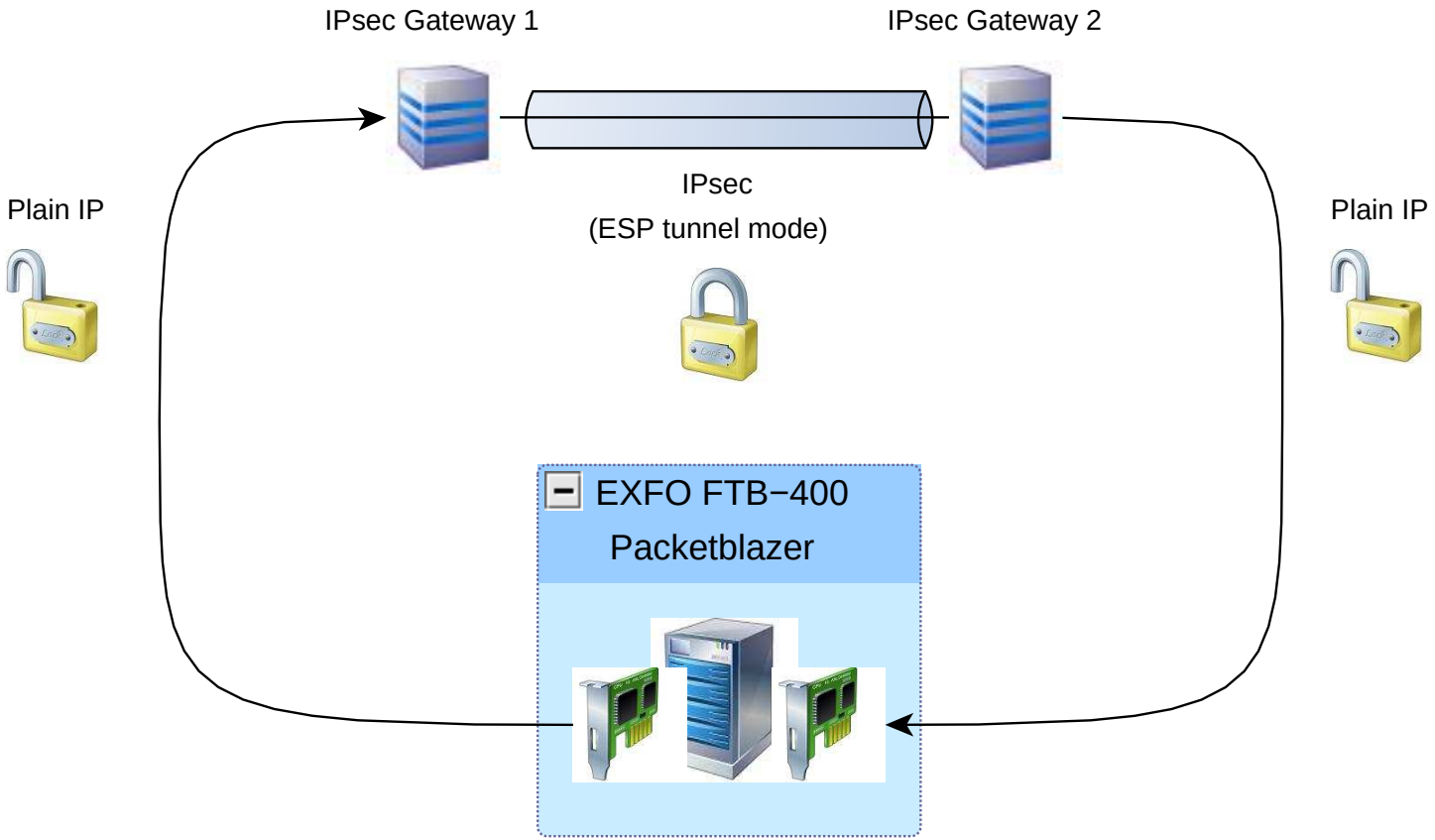
secunet

The software test setup

- Kernel: linux-2.6.33-rc7 with two additional patches (padata/pcrypt) picked from the cryptodev-2.6 tree.
- IPsec: Tunnel mode ESP on IPv4.
- Encryption/Decryption: cbc-aes-192 (x86_64 optimized version of AES).
- Authentication: hmac-sha1 (generic C version).

secunet

The hardware test setup



secunet

The hardware test setup

IPsec gateway 1 (Apligo Nexom NSA7110):

- 2 x XEON DP E5540 2.53GHz (2 x quad-core)
- 2 x 1024 DDR3 ECC
- 8 x Intel Corporation 82575EB Gbit NIC
- Intel 5520 and ICH10R Chipset

IPsec gateway 2 (SIE XL-1.0):

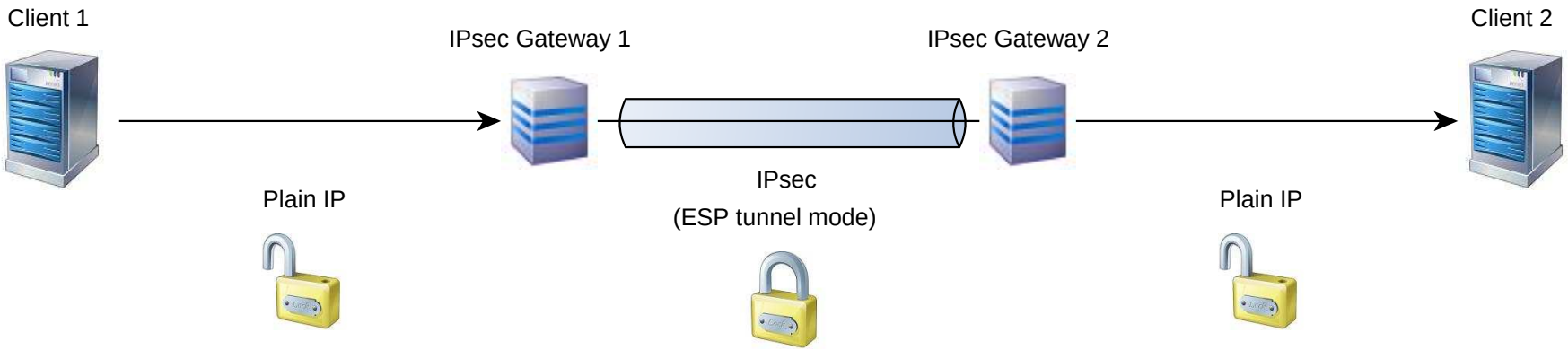
- 2 x Intel Xeon X5550 2,66GHz (2 x quad-core)
- 4 x 1024 DDR3 ECC
- Intel 4Port Gbit NIC EXPI9404PTL
- Intel 5520 and ICH10R Chipset

Hyperthreading was enabled on both IPsec gateways on all tests, so we had 16 logical cores (8 on each socket) for parallel processing.

RFC 2544 Benchmarking Methodology

- Test duration: 60 sec.
- Throughput test results: Maximal throughput rate without packet loss (60 sec.).
- latency test results: Latency at Maximal throughput rate without packet loss.
- Packet sizes RFC 2544 (Layer 2): 64, 128, 256, 512, 1024, 1280, 1518 byte.
- Used packet sizes (Layer 2): 64, 128, 256, 512, 1024, 1280, 1420 byte.
- Used packet sizes (Layer 3): 46, 110, 238, 494, 1006, 1262, 1402 byte.

Maximum theoretical throughput on Layer 3

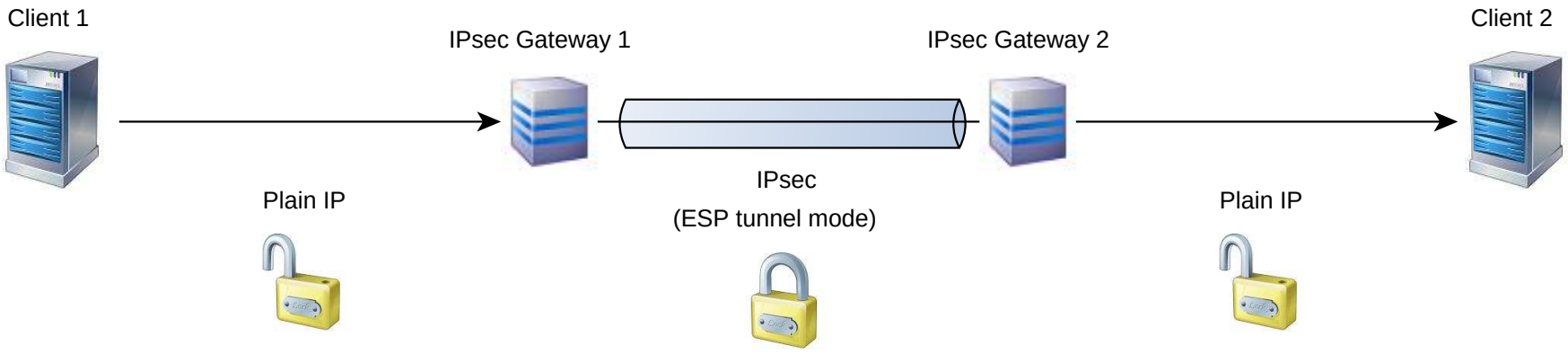


Wire speed at layer 1: 1000 Mbit/s.

IP−4 Header	TCP Header	Payload Data

secunet

Maximum theoretical throughput on Layer 3

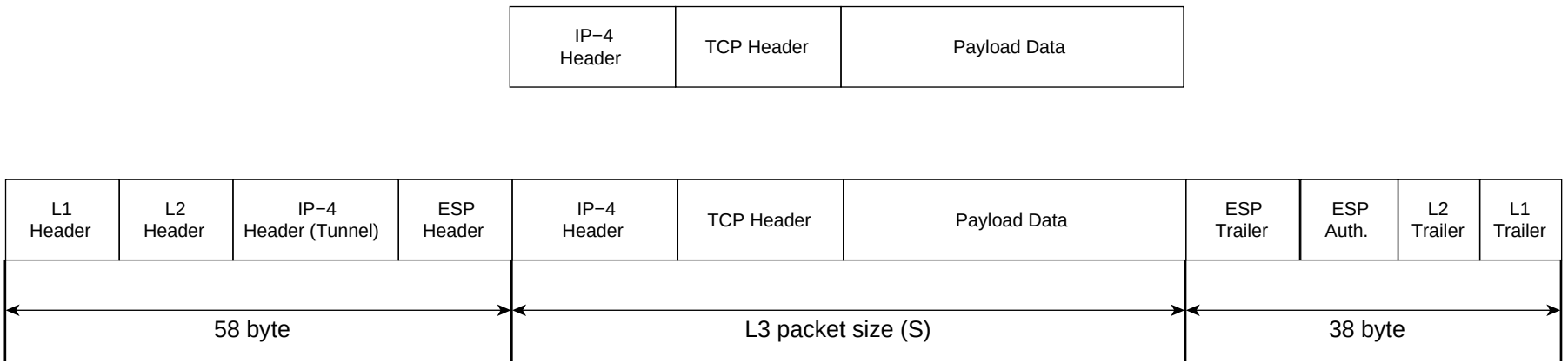


IP−4 Header	TCP Header	Payload Data

IP−4 Header (Tunnel)	ESP Header	IP−4 Header	TCP Header	Payload Data	ESP Trailer	ESP Auth.

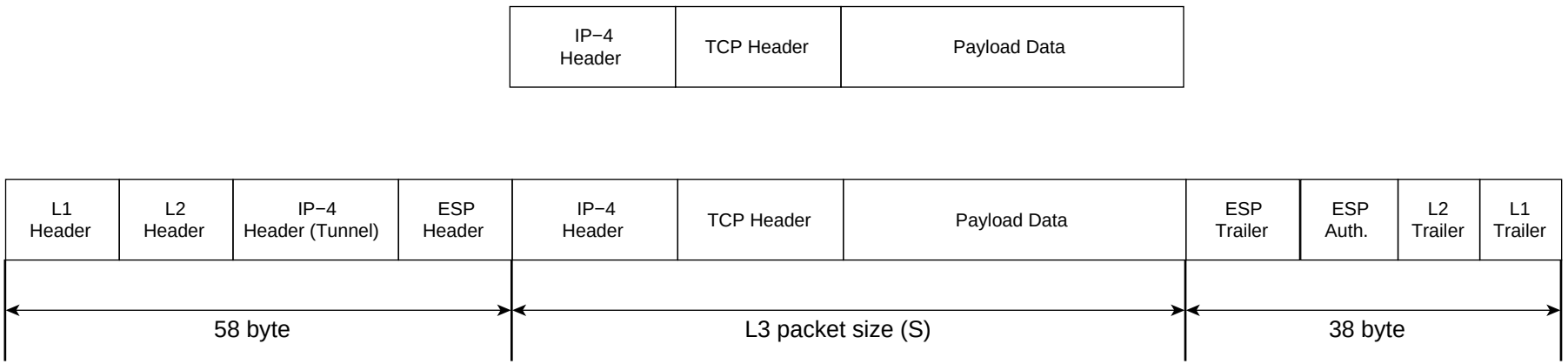
secunet

Maximum theoretical throughput on Layer 3



secunet

Maximum theoretical throughput on Layer 3



Maximum theoretical throughput on Layer 3:

$$MTT(S) = \frac{S}{S + 96} \times 1000\ \mathrm{Mbit/s}$$

secunet

Maximum theoretical throughput on Layer 3

$$MTT(46) = 324\ \text{Mbit/s} \quad (1)$$
$$MTT(110) = 534\ \text{Mbit/s} \quad (2)$$
$$MTT(238) = 712\ \text{Mbit/s} \quad (3)$$
$$MTT(494) = 837\ \text{Mbit/s} \quad (4)$$
$$MTT(1006) = 913\ \text{Mbit/s} \quad (5)$$
$$MTT(1262) = 929\ \text{Mbit/s} \quad (6)$$
$$MTT(1402) = 932\ \text{Mbit/s} \quad (7)$$

secunet

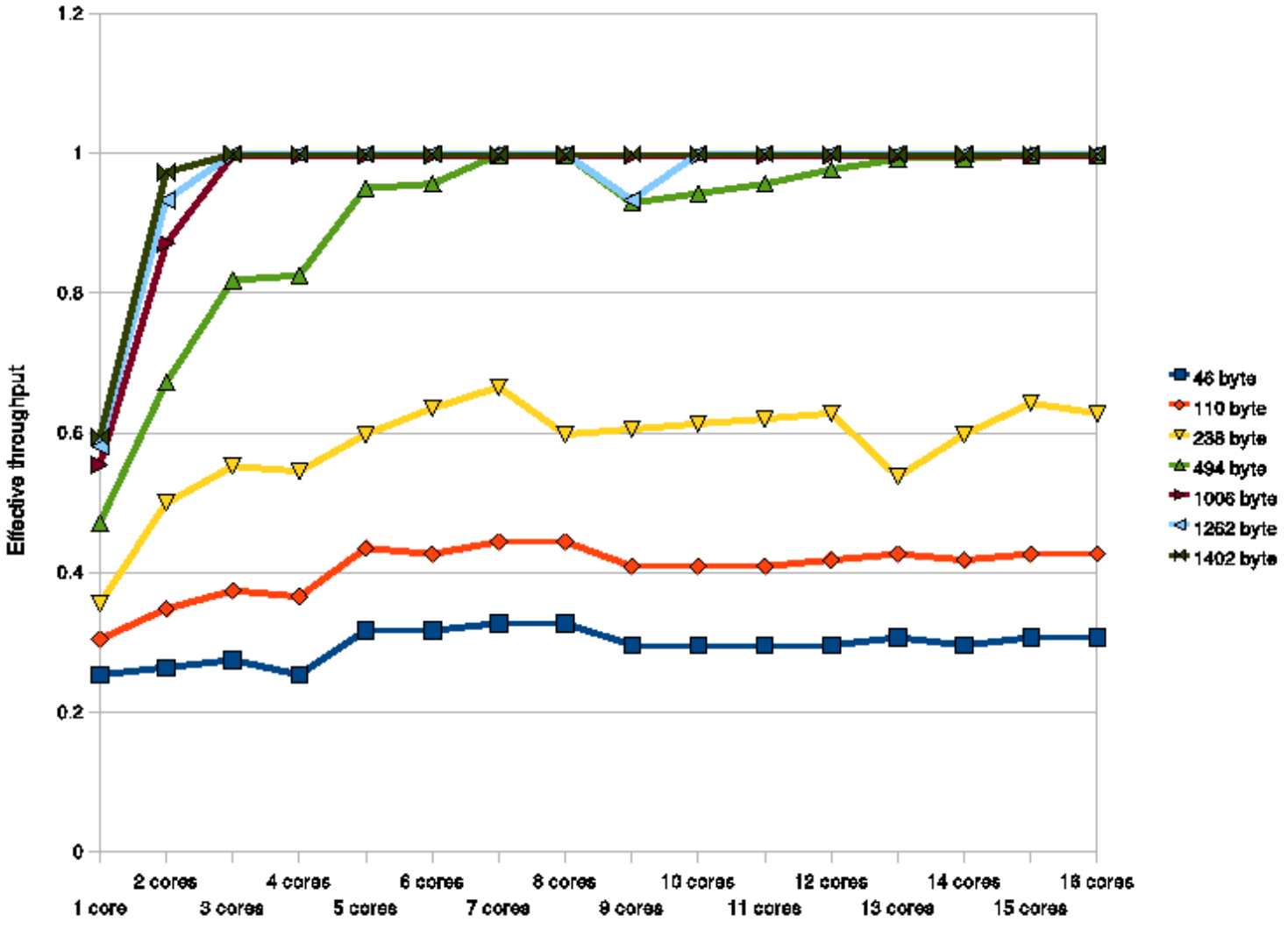
Effective throughput on Layer 3

$$ET(S) = \frac{\text{Measured throughput for packetsize } S}{MTT(S)}$$

$$0 \leq ET(S) \leq 1$$

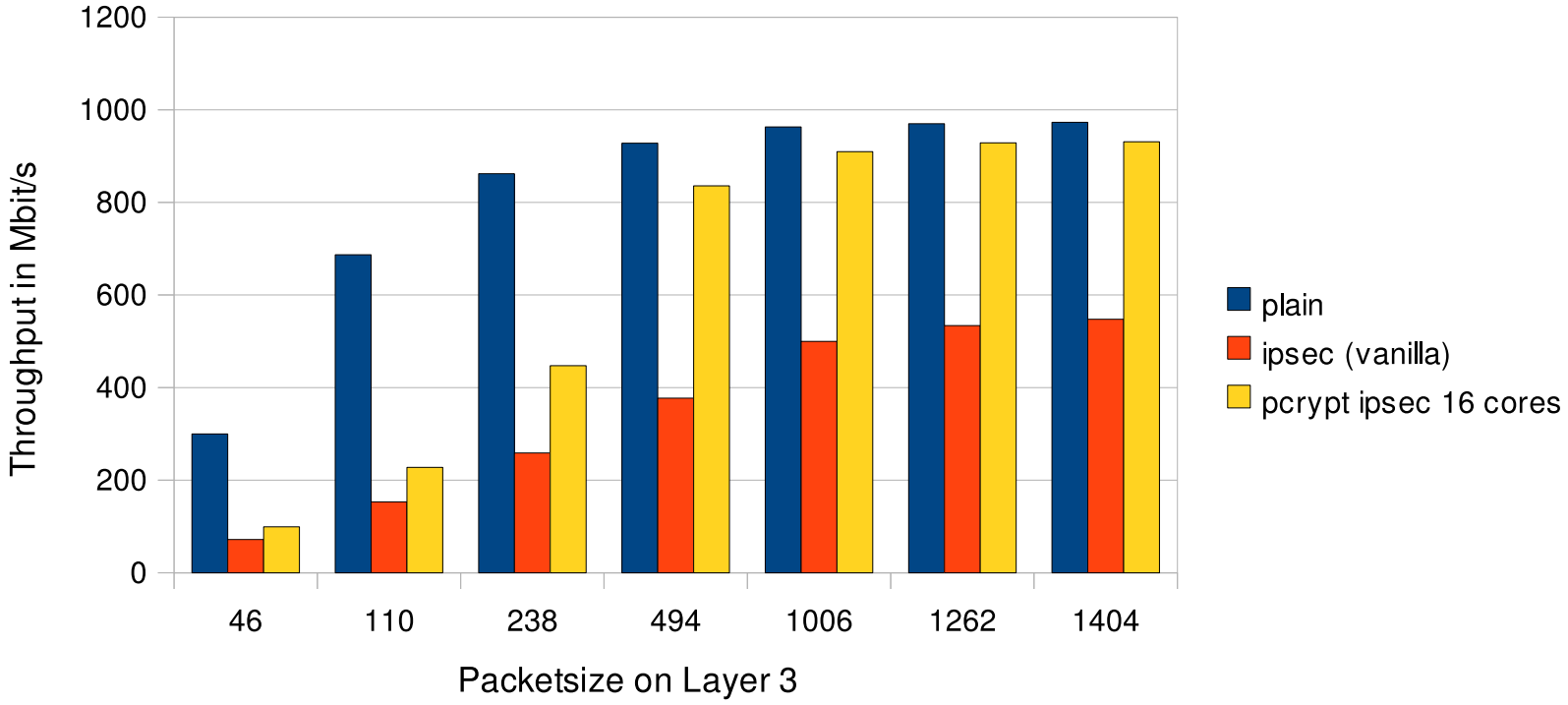
secunet

Unidirectional effective throughput benchmarks



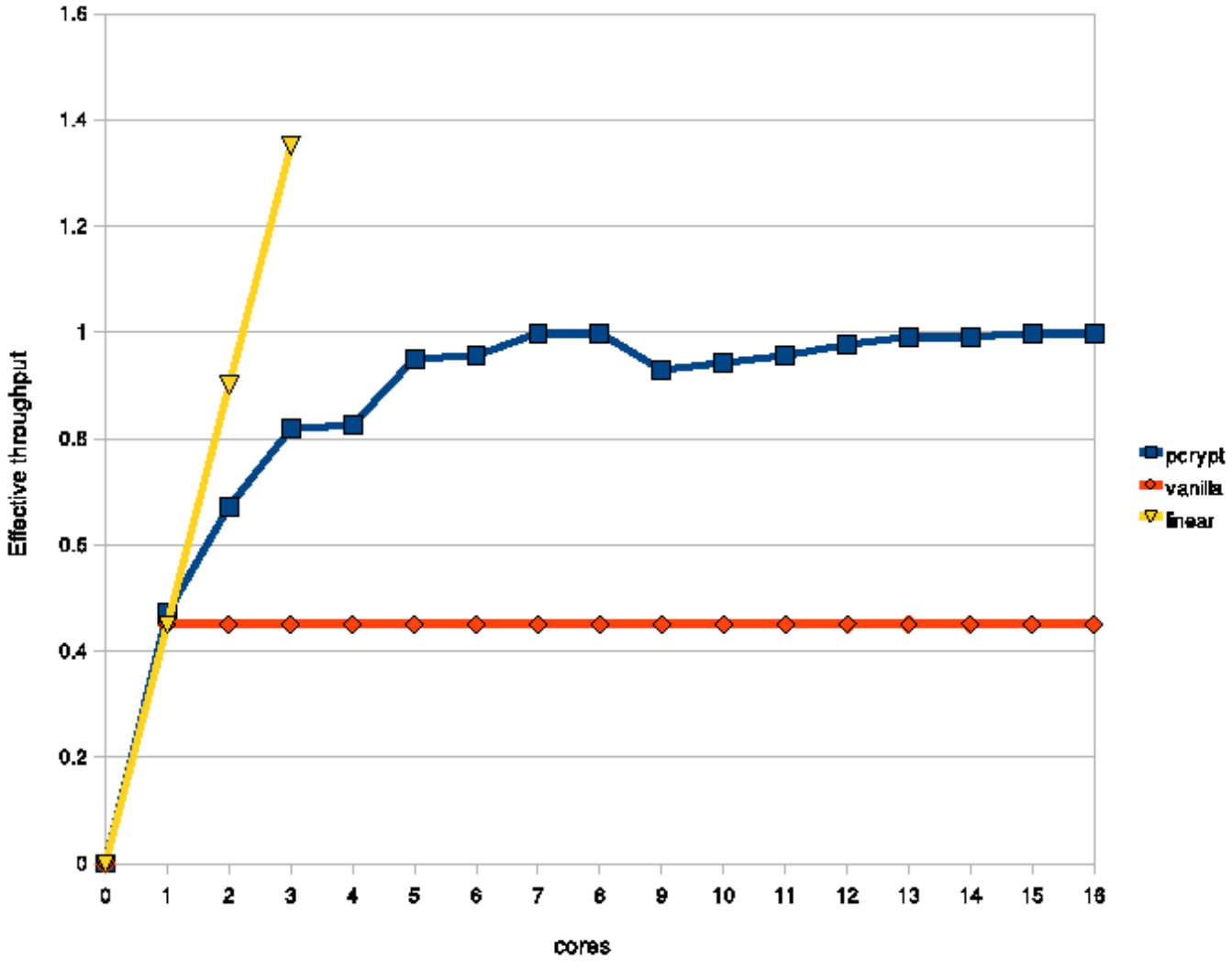
secunet

Unidirectional throughput: plain, IPsec vanilla, IPsec pcrypt



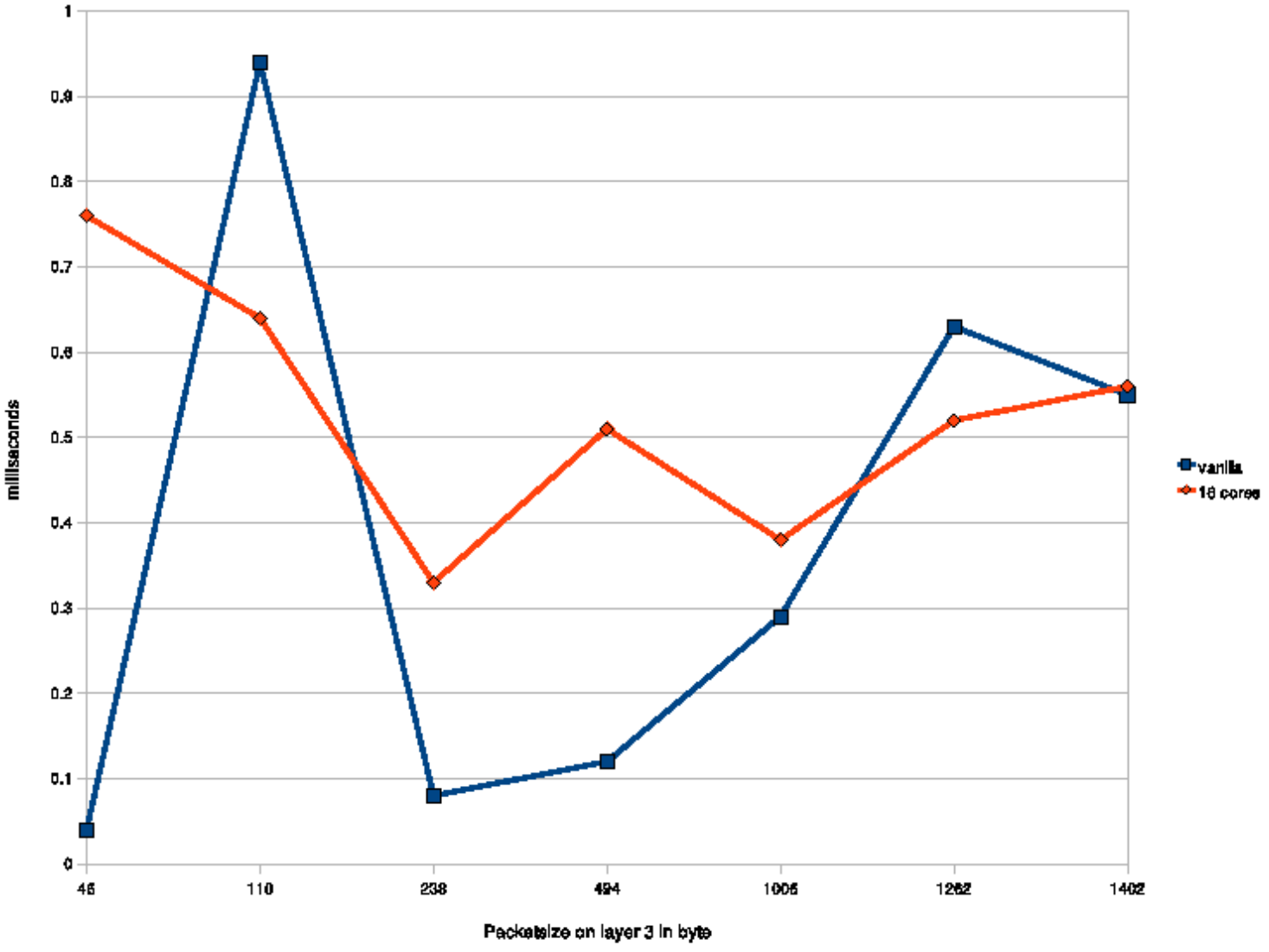
secunet

Unidirectional effective throughput 494 Byte on Layer 3



secunet

Latency with linux-2.6.33-rc7 vanilla and pcrypt 16 cores



secunet

Thanks to Apligo for providing me with test hardware!

secunet

Thanks for listening!