IKEV2 SUPPORT FOR PER-QUEUE CHILD SA

IPsec, IETF 109
November 2020

Antony Antony, Steffen Klassert, Paul Wouters

Current IPsec SA limitation

- An IPsec SA implementation typically can only use 1 CPU
- An IPsec SA implementation typically can have only 1 QoS
- Launching multiple IPsec SAs is possible, but can lead to interoperability issues:
  - Duplicate IPsec SAs getting deleted as “old”
  - Disagreement about how many IPsec SAs to use leading to TS_UNACCEPTABLE errors until optimum found
- QoS requires both sides to signal QoS level per IPsec SA

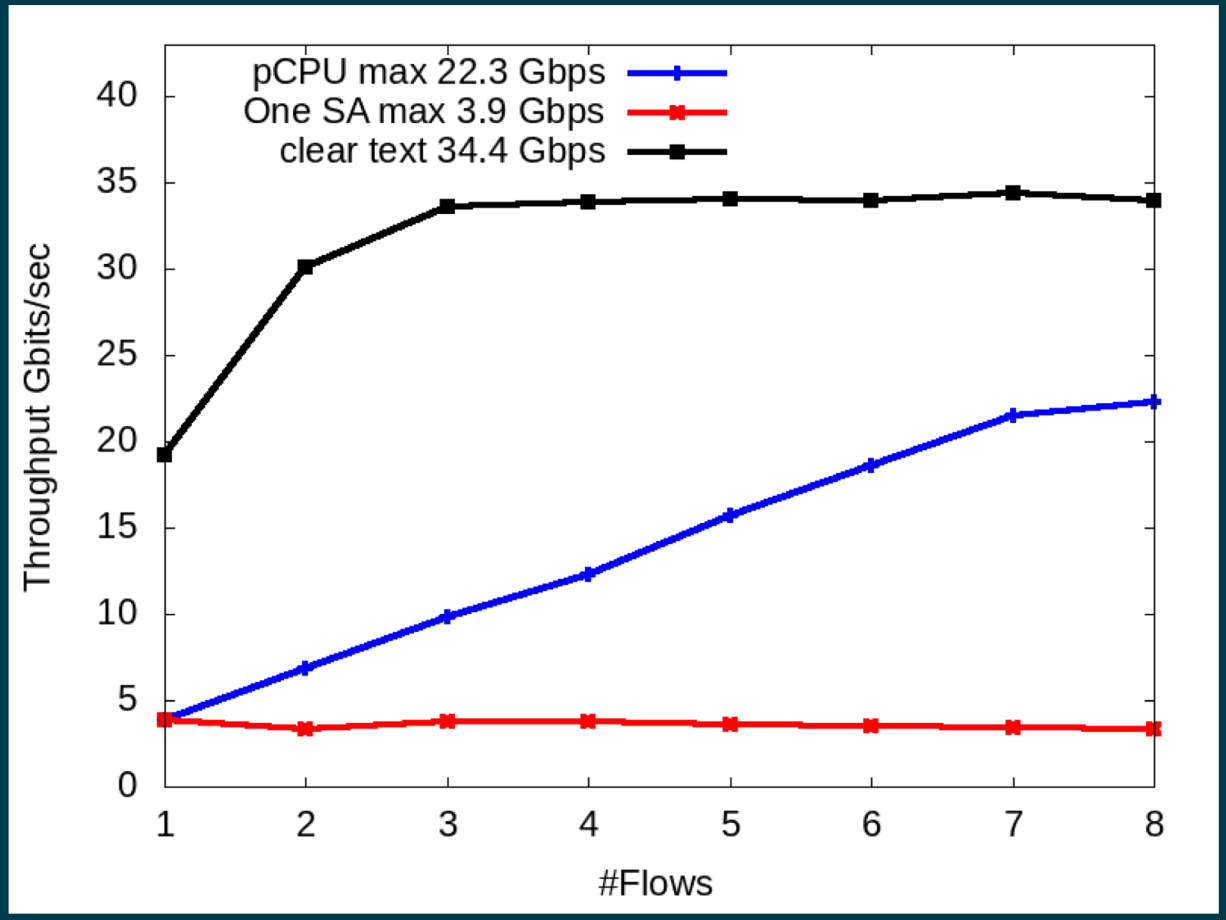
Resolve limitation by:

- Give implementation advise on how to handle multiple IPsec SA's with identical Traffic Selectors (please review document)
- Two new NOTIFY payloads for IPsec SA
  - NUM_QUEUES(pref, max)
  - QUEUE_INFO(opague)

Implementation Status:

- Linux kernel XFRM implementation (Steffen Klassert)
  - Including per-cpu (on-demand) ACQUIRE messages
- Libreswan implementation (Antony Antony)
  - Basic: implements preconfigured number of IPsec SAs
- Strongswan implementation (Antony Antony)
  - Basic: implements preconfigured number of IPsec SAs
- See draft Implementation Status for links to software

Benchmarks


pCPU max 22.3 Gbps
One SA max 3.9 Gbps
clear text 34.4 Gbps
Throughput Gbits/sec
#Flows
0
5
10
15
20
25
30
35
40
1
2
3
4
5
6
7
8

Open Issues for IKE

- Is NUM(preferred, max) the right negotiation ?
- Is there value (and/or danger) in signaling CPUID ?
- Would QUEUE_INFO need a sub registry ?
- Corner cases (eg both ends initiate for final slot) ?
- IPsec rekey changes SPI, might change CPU affinity
- NAT mapping updates causing RSS hashing changes

Hardware (issues)

- Sender assumed to use different CPUs (eg server with threads)
- Receiver hardware is where real support is needed
- Network card support for RSS
  - RSS usually only supports UDP/TCP port hashing selector
  - RSS support for ESP if there, often incomplete/lacking
  - n-tuple support – rarely available for SPI selector
  - n-tuple – if available, requires ‘manual’ configuration
  - Virtual NIC support ongoing (RSS, RFS/aRFS, “multinic”)
- Better and standardized hardware support would be good

Feedback

- Any questions?
- Is there interest in the WG ?
- Especially interested to hear from HW vendors

Bonus Slide(s)

- To use the references linux / libreswan / strongswan, you need to have support for one of these:
  - NIC with RSS for ESP support
  - NIC with RSS support with enabling UDP encap (usually done by lying in NAT_DETECTION_* payloads)
  - NIC with n-tuple support for ESP, eg:
    ethtool --config-ntuple eth0 flow-type esp4 src-ip \
    192.168.1.1 dst-ip 192.168.1.101 spi 0x12345678 \
    action 1 loc 2
    (ideally with n-tuple SPI selector for ESPinUDP)
  - NIC with n-tuple support for UDP, using UDP encap ESP
    ethtool --config-ntuple eth0 flow-type ip4 src-ip \
    192.168.1.1 dst-ip 192.168.1.101 action 1 loc 2