Intel

Optimize Intel Xeon Processors Performance Using Intel QuickAssist Technology

CORPORATE PARTICIPANTS

Gloria Nogales
Director, Ecosystem Strategy

Karen Shemer
Silicon Architecture Engineers, Network and Edge Group

Abhishek Khade
Silicon Architecture Engineers, Network and Edge Group

Georgii Tkachuk
Silicon Architecture Engineers, Network and Edge Group

PRESENTATION

Gloria Nogales

Welcome, everyone, to the Intel Network Builders Insights Series. I'm your host today. My name is Gloria Nogales, and I am Director of Ecosystem Strategy in the Network Platforms Group at Intel. Thank you so much for taking the time today to join us. Our webinar is very exciting today. It's called "Optimize Intel Xeon Processors Performance Using Intel QuickAssist Technology". Before we get started, I want to point out some of the features of the BrightTALK tool that may improve your experience.

There's a Questions tab below your viewer. I encourage our live audience, if you're seeing as live today, to please ask questions at any time. Our presenters will hold answering them until the end of the presentation. Below your viewing screen, you will also find an Attachments tab. This contains additional documentation and reference materials that pertain to this presentation today. Finally, at the end of the presentation, please take the time to provide feedback using the Rating tab. We truly value your thoughts and we'll use this information, your information, to improve our future webinars.

Our Intel Network Builders Insights Series takes place live every month, so please check the channel to see what's upcoming, and access our growing library of recorded content. In addition to the resources you see here, we also offer a comprehensive 5G and virtualization training program through the Intel Network Builders University. You can find a link to this program in the Attachments tab, as well as the link to the Intel Network Builders Newsletter. Everything is in there, so please check it out.

And today, I am very pleased to welcome Karen Shemer, Abhishek Khade, and Georgii Tkachuk. All of them, the three of them, are part of our Network Platforms Group and they're silicon architecture engineers. Specifically, Georgii focuses on performance analysis of networking workloads, including VPN stacks. Abhishek focuses on performance analysis of QAT at API level. And Karen focuses on performance analysis of networking TLS workloads, including NGINX.

Welcome, the three of you, thank you so much for taking the time today to join us. We're pleased to have you and now I'll hand it over to Abhishek to start off.

Abhishek Khade

Thank you, Gloria. So, as the world becomes more connected, we are seeing an exponential growth of data. It is projected to grow to 180 zettabytes by 2025, from 60 zettabytes in 2020. In order to process the data, new computing opportunities such as cloud, edge, analytics are emerging, transforming business operations. Those transformations can drive complexities from IT that could increase the risks to the business if security is not addressed. At Intel, we are focused on protection, protecting the data throughout all phases, starting with the lowest layer possible, the silicon. Encrypt everything, throughout the lifecycle, from data generation through retirement, keep workloads and data isolated, build a chain of trust that's rooted in silicon, and we are accelerating crypto functions to help ensure you don't have to sacrifice performance to implement security.

As the complexity of applications continues to grow, systems need more and more computational resources for workloads, including cryptography and data compression. The drivers and patches offered here assist application developers to take advantage of Intel QuickAssist Technology Integrated Acceleration offered with the Intel processor-based platforms. Intel QAT has been in the market for greater than a decade, and provides the foundation for many high-performance security and storage customer solutions.

Intel QAT offers three main services. Intel accelerates the cryptographic ciphers and hashes, or authenticated encryption and decryption. Second, public-key cryptography including RSA, Diffie-Hellman, and elliptic-curve crypto for public key exchanges, or digital signatures and authentication. Third, compression and decompression, high performance deflate compression and decompression for the applications from high-performance storage to database, and big data to content delivery networks.

Intel has announced our 3rd Generation of Intel QAT in the recent Intel Xeon D-2700 product and has publicly announced future generation high-performance IP. Each generation of IP reflects the market climate and performance requirement at the time of its introduction, such as 10 and 20-gigabit per second performance in Gen 1 to 50 Gigabits per Second in Gen 2, and 100 Gigabits per Second and inline capability for the Gen 3. And integrated into the Intel Xeon D processor family. And finally, to the upcoming high-end performance of up to 400 Gigabits per Second in Gen 4. In each generation, it builds on the software drivers and the application and deployment of the previous generation, providing the high-performance applications in the market, ready to take advantage of the highest performance acceleration capabilities.

The recently announced Intel Xeon D-2700 processor, which is codenamed as Ice Lake-D, supports up to 20 Intel Xeon cores, and it has two by 100 gigabit ethernet connections with inherently flexible ethernet configurations, as well as the SKUs that support multi-port ethernet switching. This processor’s developed for high-performance 5G wireless, networking gateways, router, and security solutions that support integrated Intel QAT with performance of up to 100 Gigabits per Second. This crypto support can take advantage of the Intel Gen 3 technology where the packets can be encrypted and decrypted inline to the data path. The traditional lookaside and public-key crypto are also supported for the TLS load balancers or the service age applications, which are taking advantage of the Intel QAT, and it can provide a significant boost in the high performance at great power and cost efficiencies.

Intel's Network Platforms Group develops products to target Intel's communications, storage, and cloud market. NPG builds products to scale from high-performance 400 Gigabits per Second to mid-range 100 Gigabits per Second to entry level up to tens of Gigabits per Second. Intel develops its solutions to enable a scale from high-end gateways to base station and vRAN solutions. This allows customers the ability to design the software solutions that work on products throughout their stacks. The compression/decompression spans from greater than 100 Gigabits per Second to multi-gigabit network attached storage solutions. For the TLS and IPSec, SDWAN solutions customers can design a solution that takes advantage of the Intel QAT high-performance ciphers and hash accelerations, as well as the public-key cryptography support for verifying certificates and securing the client tunnels for the use in the content delivery networks. Webservers such as the NGINX and the TLS load balancer security solutions as well.

Now let's deep dive into the Intel QAT Gen 3 API level performance. So, the Gen 3 Intel QAT supports cryptographic ciphers and hashes, public-key cryptography engine, compression/decompression. It has advanced power management and virtualization capability. The scalability, the crypto PK, and the compression engines are independently scalable. For the connectivity, the CPU and the host configuration, and it has lookaside processing, and it supports simultaneous lookaside processing as well. The Intel QAT solution for the Xeon D-2700 adds new capabilities that provide some significant performance improvements. Like all Intel QAT accelerators, the Intel QAT Gen 3 can operate in standard lookaside mode. lookaside implies that the data uses the DMA engine within the accelerator move data from the memory through the accelerator pinpointing the attack service requested, such as encryption or the compression. In lookaside mode, the accelerator is seen as a PCIe endpoint, with the host requesting one of the three services, cryptographic ciphers, hashes, public-key crypto, or the compression or the decompression. The Gen 3 Intel QAT adds some new algorithms, as well as the ChaChaPoly authenticated ciphers and SM2/SM3 Chinese national standards.

Taking a quick look at the modes of operation of Intel QAT in the Intel Xeon D-2700 family. The new Intel QAT Gen 3 architecture supports what has been the well-known model of lookaside accelerators, as can be seen on the right. In this model, from Data I/O, all data is DMAed into main memory where the Intel Architecture CPU determines that it needs a Intel QAT operation decryption for example. The lookaside QAT Accelerator DMAs the data down through the accelerator and executes the function specified in the request, and then after the operation, the data is returned to main memory at the location or locations specified. For example, as shown in the diagram, on the Egress data may be required to be encrypted before it is sent. Again, the data goes from memory, through the accelerator, implementing the necessary encryption algorithm, and then places the data back in main memory to later be DMAed to the I/O. As can be seen in the lookaside model, the data is moved from memory through the accelerator and back to memory.

My colleague, Georgii, will review in detail the Inline model for a protocol such as IPSec, but essentially it provides an impactful performance improvement because the encrypt/decryption functions are managed in the local memory within the packet processing I/O data path. Data can be decrypted directly on the ingress before it has any packet match action or it is sent to the memory. The reverse is true on egress the data path and that can be DMAed from the memory, and the encryption is managed with the necessary flow security association inline before the data is sent to the physical layer. The Inline model frees the CPU from its utilization in the encrypt/decrypt process, and providing high performance and even greater CPU efficiency.

As you can see from the support of all of the various algorithms and modes of operation, it is a real task to verify the performance that we specified in our design requirements. We take great pains to ensure that customers can get the performance that we advertise for our products. In order to deliver products that meet the needs of our customers, we characterize each algorithm mode of operation and packet or block sizes across a broad spectrum to allow customers to determine the expected performance in a specific network or the storage loads and environments. To get the most accuracy, we measure in multiple ways. First, we measure directly from our driver APIs, as seen here, and provide customers the detailed data, as well as distinct data points at IMIX or the 1KB. Next we take measurements from distinct libraries such as QATZip or OpenSSL, and lastly, we take real-world applications such as IPSec, NGINX, QUIC, or WireGuard and get the hardware performance in real-world environments where it is expected to operate.

As can be seen above, from looking at the API level performance, if we take a snapshot at 320 bytes, you can see the various cryptographic ciphers such as AES XTS, AES CBC, AES GCM, CCP, SM4 are greater than 100 Gbps, and have shown greater performance that our architecture was specified for. Other authenticated encryptions such as AES XTS for storage or AES GCM also exceeded our specified performance. In general, the hashes at 100 Gbps and the public-key crypto, the famous RSA2K, at 80K, or 80 Key Operations per second, we have met or exceeded the performance levels that our hardware was specified for. And don't forget about wireless ciphers. Wireless ciphers such as ZUC, SNOW3G, KASUMI F8, their performance is greater than 50 Gbps. This information is fed back to the device planners to confirm that the design has met the requirements. As you will see, this is just the beginning of the analysis.

Next, we will look at the compression and decompression performance. We've seen many applications and markets, from storage to content delivery, to database, to VM migration embrace Intel QAT high performance compression. Intel QAT Gen 3 supports the deflate standard, which consists of two parts, LZ77 and Huffman coding. Deflate is commonly used in gzip or zlib. Again, to characterize with a high degree of accuracy for customers, we test verified compression, decompression, and latencies at a large array of block sizes, generally using the same dataset such as Calgary Corpus. We'll then document the performance for customers, highlighting common block sizes at common levels of the compression. We like to show throughput performance and the ratios for a given corpus at a given block size. As you can see here in Gen 3 Intel QAT at 128 KB blocks, we can get about 70 Gbps. At this block size, we can get about 40% of the compression ratio at a much lower latency than software compression solutions. Decompression shows an even higher performance coming in at about 80 Gigabits per Second for a Level 1 with Calgary Corpus. This is managed at a much greater core utilization efficiency to boot. Add to that the ability to compress and tag blocks with cryptographic hashes in a single pass through the accelerator, and the ability to encrypt the compressed blocks, makes for a compelling storage solution.

Next, we'll take a look at some of the interesting application-level performance. My colleague, Georgii, will review with you the real-world applications that we have run with the new Intel Xeon D-2700 family, showing the inline and the lookaside crypto.

Over to you, Georgii.

Georgii Tkachuk

Thanks, Abhishek. Yes, as Abhishek had said earlier we characterize Intel QuickAssist Technology performance at various levels, from the hardware directly, through APIs and libraries, as well as at the application level. So, I'm going to look at some exciting packet processing applications employing Intel Xeon D-2700 series CPU in the data plane, and offloading cryptographic operations to Gen 3 Intel QuickAssist Technology. I will take you through several VPN application examples we ran in our lab and show you the architecture and configuration of our tests.

I'll start with a chart that demonstrates the performance advantage of the new 3rd Generation QuickAssist Technology over the previous generation. As Abhishek mentioned earlier, Intel Xeon D-2700 series can operate QuickAssist in lookaside mode. So, here we take a look at the difference in performance of 2nd Generation QuickAssist Technology integrated into our previous generation Skylake-D SOC, and the updated 3rd Generation QuickAssist integrated into our current generation Ice Lake-D SOC. One of the applications we use to measure performance of QuickAssist in our lab is a vector packet processing platform. It's an open source…excuse me. It's an open source software under FD.io Linux Foundation Project umbrella. In our lab we use VPP for many packet processing scenarios, including switching, routing, ACL, and IPSec. VPP software uses DPDK libraries under the hood, allowing it to use Intel Multi-Buffer Crypto Library with vectorized AS Intrinsics, or offload cryptographic operations to QuickAssist.

In this case, we configured the VPP platform to serve as a realistic IPSec gateway with QuickAssist hardware lookaside offload, and measure its throughput by sending network traffic to it from a hardware traffic generator. For this test, we allocated three CPU cores to the application. This ensures that QAT is not starved for compute and is fully utilized. This graph demonstrates the upper bound hardware performance of QuickAssist with a AES128-CBC-HMAC-SHA1 algorithm at a mid-range packet size of 512 bytes and large packet size of 1420 bytes. As shown in the graph, the latest generation of QuickAssist Technology performs 24% better with 512 byte packet sizes, and up to 48% better with large packets.

Next, let's talk about one of the really exciting new features of the Intel Xeon D-2700 series CPU. It’s the ability to offload ingress and egress IPSec protocol operations in line in the NIC data path. Inline IPSec capability is supported in Xeon 2700 series NX models, where integrated ethernet and QuickAssist are enhanced with a flexible packet processor and switch component. Together, they're assembled into a NAC, which is a network acceleration complex. With the addition of the switching hardware, NAC can look up IPSec security associations and schedule crypto operations directly to QAT, directly to QuickAssist hardware, offloading compute-intensive operations and saving CPU cycles. In lookaside mode conversely, the operation of receiving and offloading the packets to QuickAssist costs extra CPU cycles on compute, and requires the data is first placed into the host’s main memory. This is not the case with inline IPSec, making the inline operation more efficient.

Now that the traditional lookaside mode is still available in the CPU model, even with NAC in switch mode. In NAC, standard Intel ethernet controller features like Flow Director and ACLs are enhanced with switching flexible packet parsing, fair queuing, traffic management, flow shaping, and other features. Adding Intel QuickAssist to this mix to offload IPSec inline makes for high performance and a very efficient product.

Inline IPSec supports up to 16,000 offloaded IPSec security associations, nine kilobyte jumbo frames, tunnel and transport mode, and concurrent NIC offloads. IPSec with new authenticated encryption and associated data, or AEAD standards, calls for support of AES GCM, AES CCM, with various key sizes and modes, 128, 192, 256. It also includes the latest, newly supported by QuickAssist, crypto algorithm ChaChaPoly. Intel QuickAssist also supports many other algorithms and modes, such as AES 256 cipher block chaining, for example, and other modes that Abhishek mentioned earlier.

The diagram on the right shows how we set up these tests in the lab. We use DPDK IPSec security gateway example application. In setting up a real-world application for benchmarking inline IPSec, we look to create a model where we can test the performance of a completed bidirectional IPSec data path. We configure two identical systems as IPSec security gateways with egress data going through an encrypt function, and the ingress of the second unit taking the encrypted data and decrypting it inline, before making any forwarding decisions on the header.

The performance chart shows a comparison of three IPSec modes, all run on Intel Xeon D-2700 NX series CPU. The three modes are IPSec with no hardware offloads at all, IPSec with lookaside offload of crypto operations, and inline IPSec. We established 5,000 IPSec tunnels between the two security gateways and we used AES 256 GCM algorithms for all incoming packets. Note that because we used only a single CPU core for this test, our throughput in all cases is limited by the compute. We decided to test this way to demonstrate the CPU savings that can be achieved by offloading expensive operations to NAC and Intel QuickAssist Technology. So, here we can see a few obvious items leap out at us. Even at the 512-byte packets, with a single CPU core, we can get up to 27 Gigabits per Second, and up to 73 Gigabits per Second with large packets when offloading IPSec inline. As you can see, offloading cryptographic operations to Intel QuickAssist Technology can substantially improve performance per CPU core. At 1420 byte packet size, we get 220% improvement of performance over software crypto libraries with lookaside mode, and 270% of software libraries performance when using inline IPSec.

Another interesting VPN use case for QAT is WireGuard. The WireGuard protocol is becoming much more popular, with many customer requests coming in for us to characterize the performance in our labs with Intel silicon. WireGuard exclusively uses the new ChaChaPoly AEAD algorithm, so it is a perfect test for the newly-added support for this algorithm in Gen 3 Intel QuickAssist Technology. We tested WireGuard performance in several ways as well. First, using the default Linux kernel WireGuard package from Ubuntu 2004 and Linux kernel networking stack. Then using the recently-added VPP WireGuard plugin that can take advantage of fast VPP packet processing, as well as the Intel vectorized AS&I instructions. And finally with the same VPP stack, but offloading the ChaChaPoly encryption and decryption to Intel QuickAssist hardware. As you can see, VPP plugin improves performance of WireGuard by moving it to userspace, but the QAT…but the QuickAssist implementation bests the performance of Multi-Buffer Software implementation by up to 5.4 times with a single CPU core.

To round up the VPN section, I would like to summarize that real-world VPN stacks can achieve between 50 and 100 Gigabits per Second throughput with just a few cores. This allows us to reserve the remaining cores for the application or other services.

With this, I'm going to turn it over to the next topic, TLS, that will be covered by our performance colleague, Karen Shemer, who has mastered the public-key and cipher performance of TLS.

Karen Shemer

Thank you, Georgii. TLS is a common part of our existence, that we don't even realize when we encounter it. Nearly every app or website or content we consume is managed via TLS and uses public-key cryptography to manage secure tunnels. I wanted to take a few moments to break down the handshake that sets up given tunnels and establishes the secure connections.

I've highlighted the areas where new optimized Intel instructions or Intel QAT supports the crypto, hash, public key, or digital signature acceleration. OK, so let's get into it.

For every HTTPS communication, there is first a TLS handshake that needs to be established. This handshake serves two primary purposes, one being authentication of the server, and the second being establishing symmetric keys for the encryption of all future data. This process happens over many messages between the client and server, each progressing the handshake another step further. The diagram on this slide is simplified to some extent, but highlights the main operations and complexities of an ECDHE-RSA TLS handshake. In the context of ECDHE-RSA, the key exchange, ECDHE shown with solid lines, and authentication, RSA shown with dashed lines, are two distinct steps, and since the key used in ECDH is ephemeral and not associated with the long-lived public/private key pair, it provides perfect forward secrecy, which was highlighted as a requirement during the Heartbleed vulnerability.

These cryptographic calculations can be expensive operations, in particular some of the signature algorithms. Starting with the latest Intel Xeon generation, there are new crypto instructions that can be used for improved software performance for ECDH, ECDSA, and RSA operations. Additionally, QAT offload can provide acceleration of the mathematical operations involved in ECDH, ECDSA, RSA, and pseudo random functions, or PRF for short, used in the TLS handshake.

We like to use common applications that require lots of handshakes, serving lots of clients, such as a webserver. In this use case I use the NGINX webserver application to test connections-per-second performance of TLS 1.3 handshakes. A measure of how many client tunnels can be established every second.

This metric is very important not just for performance reasons, but also for managing Distributed Denial of Service attacks. The objective of those attacks is to overwhelm the site or the security element with a barrage of bot requests. The webserver or load balancer must have the capacity to manage the requests, and determine which are real or which should be blocked. Having high performance public key operations enables fast establishment with performance in reserve for the oncoming bots.

So, we fashioned our setup to max out the QAT performance and beyond. For creating client requests, there were separate machines connected to the device under testing using network interface cards and a switch. For each client there were 2,000 continuous and simultaneous connection requests. When one handshake is complete, the connection is closed and a new one is opened in its place. For connection-per-second tests, the clients do not request any file, and multiple clients were used to ensure that there were no limitations from the client side.

As we can see from the results, both Intel Optimized Software solutions and Intel QAT Accelerator do a very effective job at improving the connections per second. First, let's look at the architecture upgrade itself. Simply using the new generation Intel Xeon D-2798 provides a gen-to-gen improvement when running standard software libraries. If we assume that developers will want an efficient system that needs to run the TLS application of 4 Cores 8 Threads, we can see that just from gen-to-gen we get performance improvements of 22% simply from the architectural improvements. If we then seek to use the newest instructions on the most optimized libraries, we see we can gain another 255% performance improvement. Then taking that one step further, if we add the Intel QAT Accelerator, we can get another 61% improvement from that. If we have a high-end performance system requiring on the order of 50K operations per second, we can assume an allotment of a large webserver consisting of 12 hyper threaded cores. We can gain 408% performance from standard software to using Intel QAT, which tops out near 49K perfect forward secrecy handshakes, with elliptic curve X25519 and RSA2K key.

As the graph illustrates, a single SOC, like the Intel Xeon D-2700 family, can support a high-performance security solution running a TLS-based stack with 12 cores for optimal performance per watt and performance per cost efficiency.

And with that, I think we can turn it back to Gloria to answer some of your questions.

Gloria Nogales

OK, well, thank you, everyone. That was very insightful, and it looks like some questions are coming through. So, let's see. I can see one here, let's see. It says, was the best compression ratios were...that you were... No. What was the best compression ratios were you able to get in testing of deflate compression? Hmm, I hope I made sense. Let me read that again. What was the best compression ratios were you able to get in testing of deflate compression? Hopefully, that makes sense to you guys.

Abhishek Khade

Yes, yes, and I can answer to that question. So, yes, it's a really good question. So, the best compression ratio that we have seen, actually, it's with the dynamic compression, dynamic deflate compression at the 64k packet sites, with the Calgary Corpus as the input data file, and the compression ratio measured was 35%.

Gloria Nogales

35%, OK. Thank you. Let's see, we've got another one here. Will my existing QuickAssist Technology adapter boards work in this new platform alongside the Xeon D, the new chip?

Karen Shemer

Yes, I'll take this one. So, another good question with a pretty simple answer. Your existing QAT adapter will work with the new platforms along with Xeon D.

Gloria Nogales

Oh, OK. Very good. Let's see. I can see another one here. Will I get inline QuickAssist Technology with every Xeon D-2700 device?

Georgii Tkachuk

Yes, I can answer this. So, you will only get inline QuickAssist, inline IPSec with Xeon D-2700 series NX CPUs. These are the CPUs that have the switching component in the NAC. By the way, the support for inline IPSec is upstream in DPDK, it's an open source. And if you want to get started with inline IPSec, please contact your friendly Intel customer service representative.

Gloria Nogales

OK, sounds great. Oh actually, that's a great point. We're leaving our email addresses here, and if you guys have any questions, please reach out to us, and we'll sort out who can best answer your question.

So, thank you, looks like I'm not seeing any other additional questions here, but those were really good, So, thank you to those in the audience. And with that, I think that we'll thank you, Karen, and Abhishek, and Georgii, for your great work here. Lots of hours in the lab, I can tell. So, this was great information, great insights today.

Thanks, all of you out there for joining us today. We truly appreciate that you're here, and thanks for being live. Please do not forget to give us a rating. We truly appreciate it, and we continuously improve the quality of our webinars based on your feedback and your input. So, thanks a lot for that, and please make sure to join us next time. We'll be back on April 6th. That's another Wednesday. It'll be at eight o'clock in the morning, Pacific Time, and we'll go through the topic, the next-generation Intel Xeon D SOC system on a chip, and platform built for the edge. So, please join us then.

Until that moment, thank you again. Thanks for those of you watching live and thanks for those of you watching us later. So, thank you and this concludes our webcast. Bye-bye.