Solution Brief

Ubiquitous Availability of Crypto Technologies

Intel® is collaborating with network security vendors to deliver performant software solutions across private and public clouds.

Executive Summary

The rise in remote access and increasingly complex multi-cloud architectures is escalating organizations' reliance on network security solutions. Network security vendors are tasked with delivering solutions that are both performant and deployable across the multitude of compute environments.

Intel is accelerating this transition by offering a library adhering to the industry-adopted OpenSSL framework to hide the details of the underlying crypto technologies including Intel® QuickAssist Technology (Intel® QAT), Intel® AES New Instructions (Intel® AES-NI), AVX-512 Vector Advanced Encryption Standard (VAES) instructions, and other crypto technologies found on the Intel® architecture platform.

Network security software solutions now can transparently take advantage of the capability and associated performance of the underlying deployed platform in our multicloud world by simply linking with Intel's latest library release.

This document is part of the Network Transformation Experience Kit, which is available at https://networkbuilders.intel.com/network-technologies/network-transformation-exp-kits.

The software library (Intel® QAT Engine for OpenSSL*) is available at https://github.com/intel/QAT_Engine.git.

Introduction

Traditional security vendors offering solutions as physical appliances (for example, firewall, intrusion prevention systems, application security, access control) are being called on by IT departments to also provide cloud-delivered security solutions to deliver sufficient user security. Whereas a security application can count on specific crypto technologies designed into a hardware appliance, that is not the case when the application must be designed to be targetable to the variety of cloud compute environments.

Most of the network security vendors rely on the OpenSSL project, and specifically the libcrypto general purpose cryptographic library, when executing in the various cloud compute environments. To address the need to have both portability and performance, Intel offers the Intel® QAT Engine for OpenSSL* as an additional option to the default library.

NOTE: The general information contained within this document applies to both the 3rd Generation Intel® Xeon® Scalable processor and the Intel® Xeon® D processor. However, please note that the performance, configurations, and feature set in this document apply specifically to the 3rd Gen Intel® Xeon® Scalable processor and may vary for the Intel® Xeon® D processor.

Intel shields the details of the underlying crypto technologies that the application has landed on while assuring optimal performance. Our libcrypto library looks for specific CPU generation and instruction capability, as well as any hardware acceleration that may be present, in order to create a mapping of the best performing algorithm. Algorithms are offloaded to hardware when the performance gain is greater than the cost of going to the hardware accelerator. The automatic selection is established during the initialization of the library engine.

Solution Description

The Intel® QAT Engine for OpenSSL*, as seen in Figure 1 for OpenSSL 1.1.x, improves the performance of secure applications by directing the requested computation of cryptographic operations to the available hardware acceleration or instruction acceleration present on the platform. The engine supports both the traditional synchronous mode for compatibility with existing applications and the new asynchronous mode introduced in OpenSSL 1.1.0 to achieve maximum performance.

After the engine is loaded and initialized, all crypto operations that have been registered and executed via the EVP API are offloaded transparently to either Intel® QAT or executed in software by using available crypto software instructions. This provides access to all available performant crypto technologies while significantly reducing the time and the cost required to integrate the technologies into an application and keeping the application portable across the full suite of Intel roadmap offerings.

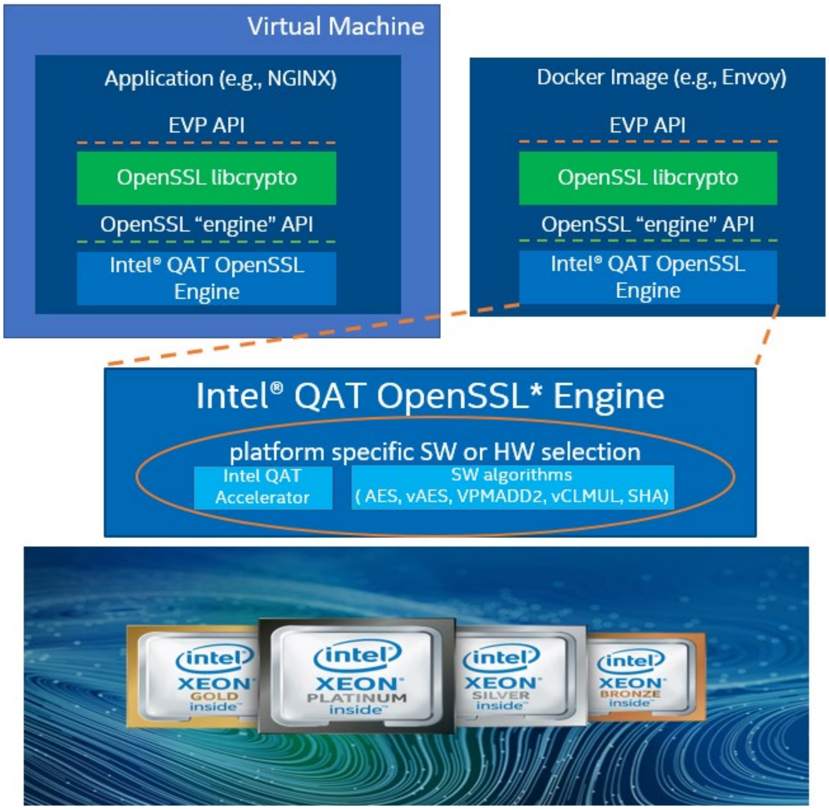
Virtual Machine
Application (e.g., NGINX)
EVP API
OpenSSL libcrypto
OpenSSL "engine" API
Intel® QAT OpenSSL Engine
Docker Image (e.g., Envoy)
EVP API
OpenSSL libcrypto
OpenSSL "engine" API
Intel® QAT OpenSSL Engine
Intel® QAT OpenSSL* Engine
platform specific SW or HW selection
Intel QAT Accelerator
SW algorithms (AES, vAES, VPMADD2, vCLMUL, SHA)
intel XEON GOLD inside
intel XEON PLATINUM inside
intel XEON SILVER inside
intel XEON BRONZE inside

Figure 1. Intel® QAT Engine for OpenSSL*

Our library engine is written from the point of view of performance. The library engine offloads to hardware when the performance gain is greater than the cost of going to the hardware accelerator. For each given crypto operation, the engine establishes the highest performant option by considering the CPU generation, the speed of the CPU, and the cost of available hardware offload. The automatic selection is established during the initialization of the library engine. Any optional application preference (for example, never use hardware offload, minimize latency over throughput) is also factored into the initialization logic.

Technologies Implemented

Encryption Focused Instructions

Intel® AES New Instructions (Intel® AES-NI) are a set of instructions available beginning with the 2010 Intel® Core™ processor family based on the 32 nm Intel® microarchitecture code named Westmere. These instructions enable fast and more secure data encryption and decryption by using the Advanced Encryption Standard (AES), which is defined by FIPS Publication number 197. Since AES is currently the dominant block cipher, and it is used in various protocols, the new instructions are valuable for a wide range of applications. Intel® AES-NI consists of six instructions that offer full hardware support for AES. Four instructions support the AES encryption and decryption, and the other two instructions support the AES key expansion. The AES instructions have the flexibility to support all usages of AES, including all standard key lengths, standard modes of operation, and also some nonstandard or future variants. They offer enhanced performance compared to the current pure software implementations.

Intel has also announced the introduction of additional encryption instructions with the 3rd Generation Intel® Xeon® Scalable processor. The additional instructions include VPMADD2 - vector instruction that performs integer multiply accumulate, vAES - vector version of the AES-NI instructions, vCLMUL- vector version of the CLMUL instruction, and SHA-NI - secure hash algorithm new instructions. The combination of vAES and vCLMUL on wide registers that are available on the AVX-512 architectures further speed up AES modes such as AES-CTR and AES-CBC. VPMADD2 is targeted at significantly reducing the instructions needed to generate public/private keys as part of an RSA-2K sign operation. We also take advantage of the VPMADD2 instruction to accelerate the Poly1305 message authentication algorithm favored with newer Wireguard and QUIC protocols. SHA-NI works to improve hashing functions utilized in cryptographic protocols such as SSL/TLS as well to help with data deduplication in storage workloads.

Intel® QuickAssist Technology

The following compute-intensive workloads are common to many multicloud infrastructure applications:

- Symmetric cryptography, also known as bulk cryptography, is used to help secure data in flight or data at rest, thereby helping to providing data confidentiality, integrity, and authentication. This workload is used in networking in applications such as IPSec gateways and SSL/TLS applications, including secure web servers, SSL proxies, load balancers, and application delivery controllers. It is also useful in storage applications.
- Asymmetric cryptography, also known as public key cryptography, is typically used to perform a key exchange between parties that then use the derived keys to perform the bulk cryptography later. For example, the SSL/TLS protocols perform one or more public key cryptographic operations as part of an SSL handshake at the beginning of each new connection.
- Compression is used to reduce the size of data in flight or data at rest, thereby saving on the cost and/or latency to transmit the data over the network, or to read the data from, or write the data to, a storage device.

All these workloads can be performed in software on a CPU. However, encrypting or compressing large quantities of data at line rate on a network function can consume significant CPU cycles. Public key cryptography is even more compute-intensive, for example, a single RSA private key operation, using 2048-bit keys can consume a million CPU cycles. A web server may have to perform thousands or even tens of thousands of such operations per second, consuming many CPU cores just to perform the cryptography, over and above the cores required to run the rest of the networking stack and application.

Benefits of Solution

To highlight the use and performance benefits of the Intel® QAT Engine for OpenSSL*, let us analyze a web server that is using NGINX (http://nginx.org/) as the management application focusing on SSL/TLS performance.

SSL/TLS is utilized in client-server based applications to provide security to the data being communicated. From a security protocol perspective, the ability of TLS to provide more secure communications between two TCP ports is one of its primary advantages. In such condition, the individual ports typically translate to an individual application on a client system and provide isolation between the multitude of applications that could be potential attack points for the client.

This client centered view is important, as our benchmark metrics will show, the client metrics are the primary performance driver. For a client, the ability to connect to many services and transfer data seamlessly is key. On the server side, this translates into the number of new connections per second a server can create. With this viewpoint, we have now identified the key metric to drive the analysis: number of SSL/TLS handshakes per second for an SSL/TLS server.

Benchmark Topology

For more information on benchmark topology, see details in the Intel® QuickAssist Technology & OpenSSL-1.1.0: Performance white paper.

Figure 2 shows the TLS Handshake performance on a single core of Intel's latest Intel® Xeon® Scalable processors (3rd Generation Intel® Xeon® Scalable processor HCC) offering that uses a default OpenSSL engine, Intel® QAT Engine for OpenSSL* with the latest crypto instructions, and Intel® QAT Engine for OpenSSL* with the Intel® QuickAssist Technology.

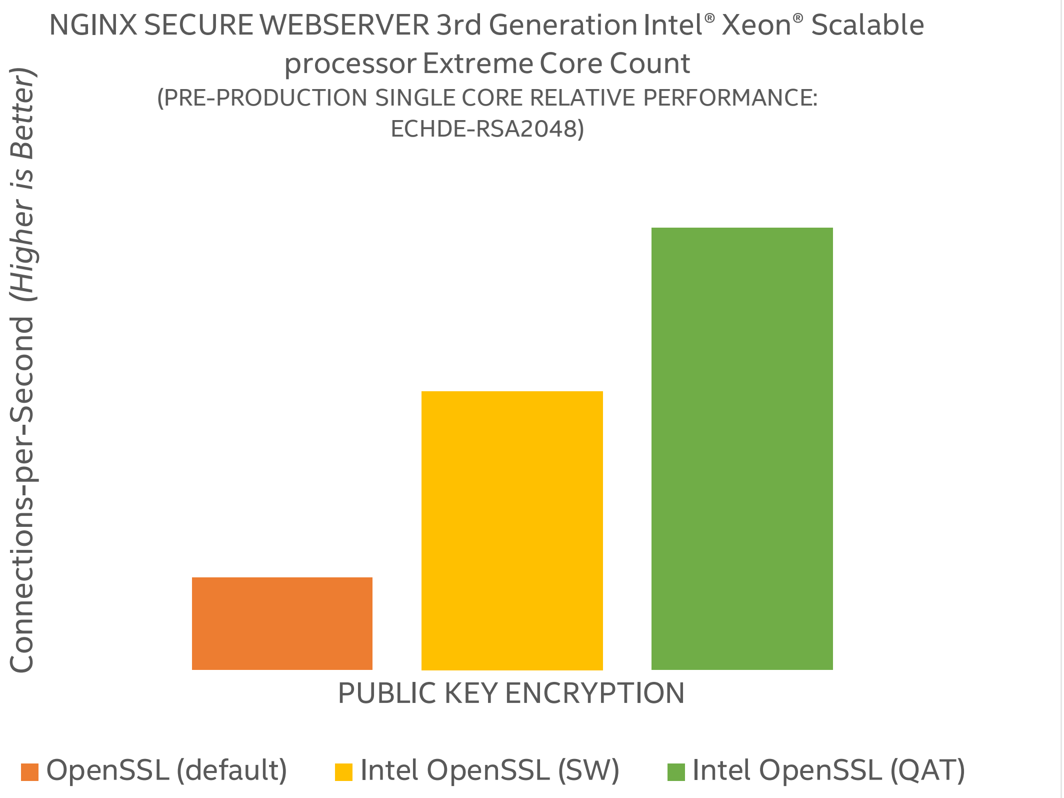

Figure 2. Example of TLS Handshake performance[1]

Use Case Examples

As the Intel® QAT OpenSSL Engine adheres to the OpenSSL libcrypto APIs, all cloud-delivered security solutions relying on the OpenSSL libcrypto APIs for cryptographic algorithms can benefit from linking with our library.

Network security solutions, including firewall, intrusion prevention systems, application security, access control, secure web gateways, web application firewalls, and application delivery controllers, benefit from the performance offered by the OpenSSL engine while also maintaining the portability desired for the diversity of deployments found in a multicloud environment.

Summary

The Intel® QAT Engine for OpenSSL* Engine improves the performance of secure applications by directing the requested computation of cryptographic operations to the available hardware acceleration or instruction acceleration present on the platform. The engine supports both the traditional synchronous mode for compatibility with existing applications and the new asynchronous mode introduced in OpenSSL 1.1.0 to achieve maximum performance.

Network security software solutions can transparently take advantage of the capability and associated performance of the underlying deployed platform to deliver solutions that are both performant and deployable across the multitude of compute nodes found in a multicloud environment.

Our benchmarking of the NGINX web server application that focused on SSL/TLS performance as illustrated in Figure 2 delivers a transparent performance gain of 3x to 8x when linking with our engine.

Intel is committed to maintain the engine current with each new underlying crypto technology thereby shielding the application from the details while at the same time providing performance boost.

[1] See backup for workloads and configurations or visit www.Intel.com/PerformanceIndex. Results may vary.

REFERENCES

TITLE	LINK
Intel® QuickAssist Technology & OpenSSL-1.1.0: Performance whitepaper	https://01.org/sites/default/files/downloads/intelr-quickassist-technology/intelquickassisttechnologyopensslperformance.pdf
Telco/Cloud Enablement for 2nd Generation Intel® Xeon® Scalable platform - Intel® QuickAssist Technology Application Note	https://networkbuilders.intel.com/solutionslibrary/telco-cloud-enablement-for-2nd-generation-intel-xeon-scalable-processors-intel-quickassist-technology
QAT_engine [github]	https://github.com/01org/QAT_Engine
Intel® QuickAssist Technology engine build options	https://github.com/01org/QAT_Engine#intel-quickassist-technology-openssl-engine-build-options

Appendix 1: Configuration Supporting Performance

These test results[2] are as of November 6, 2020.

	ITEM	DESCRIPTION
	Product	3rd Generation Intel® Xeon® Scalable processor
	Speed (MHz)	2200
	No of Cores	32
	Stepping	6
	Technology	14
	Level 1 Data Cache	48 K
	Level 2 Data Cache	1280 K
	LLC Cache	49152 K
Memory	Vendor	Micron
	Type	DDR4
	Part Number	18ASF2G72PDZ_3G2E1
	Size	16384
	Channels	8
	Speed	3200
BIOS	Vendor	Intel Corporation
	Version	WLYDCRB1_86B_0017_D68_2006260042
	Date	06_26_2020
	Microcode	0x8d000050
Compiler Version	GCC Version	7_5_0
	Linker Version	2_30
	Assembly Version	2_30
Operating System	OS Version	Ubuntu_18_04_5
	Kernel Version	5_4_70
Benchmark Software	nic_firm	3.25
	nic_driver	5.6.0-k
	NGINX Version	1.16.1
	OpenSSL version	1.1.1f
	QAT Engine	v0.5.46
	QAT Driver	L49000008
	Type	webserver
	GBE	300
	File Name	10 mb
	File Size	10000000

[2] See backup for workloads and configurations or visit www.Intel.com/PerformanceIndex. Results may vary.

Document Revision History

REVISION	DATE	DESCRIPTION
001	January 2021	Initial release.
002	April 2021	Revised the document for public release to Intel® Network Builders.
003	February 2022	Added a note regarding Intel® Xeon® D processor in the Introduction section. Also updated the solution overview to include a more recent acceleration (VPMADD2 instruction).

Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

Your costs and results may vary.

Intel technologies may require enabled hardware, software or service activation.