Intel® QuickAssist for Windows*

Release Notes

Package Version: QAT1.5.0-0007

Revision 005US

June 2021

Document Number: 337758-005US

Contents

Figures

Tables

Revision History

Document Number	Revision Number	Description	Revision Date
337758	005	Intel® QuickAssist Software release 1.5.0-0007 • Added Windows* Host Virtualization support via SR-IOV. Currently, only Linux* QAT VF's are supported. • Added Software Fallback for the Windows* PF driver to support Linux* Guests with the QAT Linux* VF using QAT Engine Applications • Added QATzip support for the systems without QAT hardware and services. All deflate compression operations will take the software path using ISA-L if software fallback is specified	June 2021
337758	004	Intel® QuickAssist Software release 1.4.0-0007 • Updated Supported OSs • Add support for Intel® Atom® C3000 with Intel® QAT • Updated Windows* QATzip to use standard QATzip header file • Added support for the gzip and gzipext data format • Added support of ISA-L SW Fallback for gzip and gzipext data formats	March 2021
337758	003	Intel® QuickAssist Software release 1.3.0-0009 • Updated Windows* Software Release version v1.3.0-0009 • Added new sections 1.3.1 What's New and 1.3.2 Software Release History • Updated Features Paragraph 13, compression/decompression features • Updated Section 3.0 removing Neoncity security accelerators • Added Known Issues: QATE 38968, 40170 • Added Resolved Issue: QATE-37219	April 2020
337758	002	Intel® QuickAssist Software release v1.1.0-29 • Removed Support for Windows* Server 2012 • Added known issues QATE-37219 and QATE-36847 • Resolved QATE-15336, Parcomp/FVL25 Driver Compatibility Issue Server 2012 R2 Update 1 • Section 1.1 Supported Platforms updated	March 2019
337758	001	• Initial release	June 2018

§

1 Description of Release

This document contains information on the accompanying Intel® QuickAssist Technology (Intel® QAT) Windows* Software release v1.5.0-0007. This document also describes extensions and deviations from the release functionality described in Table 3, Intel® QuickAssist Technology Software for Linux* Software Programmer's Guide for the various platforms that support Intel® QAT.

Note: These release notes may include known issues with third-party or reference platform components that affect the operation of the software.

1.1 Supported Hardware Platforms

The software in this release has been validated against the following devices:

- Intel® QuickAssist Adapter 8960 and 8970
- Intel® Xeon® Scalable Platform with Intel® C62x Chipset (with Intel® QAT)
- Intel® Xeon® D Platform with Intel® C62x Chipset (with Intel® QAT)
- Intel® Atom® C3000 with Intel® QAT

Note: Intel® QAT supports Intel® Xeon® Scalable first and second generations.

1.2 Supported Operating Systems

The software in this release has been validated against the following Operating Systems (OS):

- Windows* Server 2019
- Windows* Server 2016
- Windows* Server 20H1
- Windows* Server 20H2
- Windows* 10 Enterprise 2019 LTSC (Intel® Atom® SKU only)

1.3 Package Version

The following table shows the OS-specific package versions for each platform supported in this release.

Package Version

Chipset or SoC	Package Version	SHA256 Checksum
Top-Level Package	W.1.5.0-0007.zip	4D86F7798C208C3929226234EA32AA1F17 6A6032B4033139DB650F16E22415B5

Note: This software release has passed the Windows* Hardware Lab Kit (HLK*) Certification and contains certified device drivers.

1.4 What's New

- Added Windows* Host Virtualization support via SR-IOV. Currently, only Linux* QAT VF's are supported
- Added installation modes. Standalone is the same as previous drivers; it will install the QAT base driver, compression, and crypto service. Hyper-V mode will only install the QAT base driver in SR-IOV mode. This requires a system reboot. Do not use Hyper-V mode if compression or crypto services are needed on the Host partition
- Added Software Fallback for the Windows* PF driver to support Linux* Guests with the QAT Linux* VF using QAT Engine Applications
- Added QATzip support for the systems without QAT hardware and services. All deflate compression operations will take the software path using ISA-L if software fallback is specified

1.5 Intel® QAT Software Release Feature History

Table 1. Intel® QAT Software Release Feature History

Release History	New Features
Release 1.4.0-0007	• Added support for Intel® Atom® C3000 with Intel® QAT • Updated Windows* QATzip to use standard QATzip header file • Added support for the gzip and gzipext data format • Added support of ISA-L SW Fallback for gzip and gzipext data formats
Release 1.3.0-0009	• Software fallback in the event of hardware failure for cryptography and compression services • Improved error handling with the Intel® QuickAssist cryptography and compression services
Release 1.2.0-0018	• Intel® Intelligent Storage Acceleration Library (ISA-L) integration with Intel® QuickAssist compression and decompression services. • Compression fallback support with ISA-L • Improved error handling with the Intel® QuickAssist compression services
Release 1.1.0-0029	• Add support for PKE cryptography services
Release 1.0.0-0022	• Initial release that supports Intel® QAT compression and decompression

1.6 Data Compression Services

This software package provides the following Data Compression services:

- Static Deflate Stateless compression/decompression
- Dynamic Deflate Stateless compression/decompression
- Includes sample code application for compression services – parcomp

For ISA-L integration, the source code and information to build the DLL can be found in Table 3, Intel® Intelligent Storage Application Library GitHub. The minimum required version is 2.26.0. The DLL should be placed in the Windows* system32 directory.

The QATZip file includes the following compression/decompression functions:

```
qzInit
qzSetupSession
qzCompress
qzDecompress
qzTeardownSession
qzClose
qzMalloc
qzFree
qzGetStatus
qzGetDefaults
qzSetDefault
```

1.7 Cryptography Services

This software package also provides the following cryptography services.

Support for PKE cryptography services include:

- Cryptography API: Next-Generation (CNG) support, sometimes referred to as the "BCrypt API." Refer to Cryptography API: Next-Generation, Table 2.
- An Intel® QuickAssist CNG provider is registered to support the following PKE algorithms:
  - Rivest-Shamir-Adleman (RSA)
  - Digital Signature Algorithm (DSA)
  - Elliptic Curve Digital Signature Algorithm (ECDSA) (P256, P384, P521)
  - Diffie–Hellman (DH)
  - Elliptic-curve Diffie–Hellman ECDH (P256, P384, P521)
- CNG API support in both user mode and kernel mode

1.8 Customer Support

Intel® offers support for this software at the Application Program Interface (API) level, defined in Table 2 and Table 3 of the Programmer Guides and API reference manuals. If the field representative has created an account for you, submit support requests via the Online Service Center, https://supporttickets.intel.com/?lang=en-US.

1.9 List of Files in this Release

The Bill of Materials (BOM) is included as a text file in the released software package. This text file is labeled "`filelist`" and located at the top directory level for each release package.

1.10 Reference Documents

Table 2 lists Intel® QuickAssist Technology's generic documentation.

Table 3 lists Intel® QuickAssist Technology specific documentation.

Table 2. Intel® QuickAssist Technology's Generic Documentation

Document	Document Number/ Location
Intel® QuickAssist Technology API Programmer's Guide	330684
Intel® QuickAssist Technology Performance Optimization Guide	330687
Cryptography API: Next-Generation	https://docs.microsoft.com/enus/windows/desktop/SecCNG/cng-portal

Table 3. Intel® QuickAssist Technology Software Specific Documentation

Document	Document Number/ Location
Intel® QuickAssist Technology Software for Linux* Software Programmer's Guide	336210

1.11 Terminology

Table 4. Terminology

Term	Description
API	Application Program Interface
AES	Advanced Encryption Standard
BOM	Bill of Materials
CNG	Cryptography API: Next Generation
DH	Diffie–Hellman
DSA	Digital Signature Algorithm

Term	Description
ECDH	Elliptic-curve Diffie–Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
WHLK*	Windows* Hardware Lab Kit
Intel® QAT	Intel® QuickAssist Technology
OS	Operating System
PKE	Public Key Encryption
RSA	Rivest-Shamir-Adleman

§

2 Limitations, Known Issues and Resolved Issues

This section provides the all known limitations and known issues for Windows* software release v1.5.0-00007. For detailed information on features/limitations, please refer to the README.txt file inside the software package (./QuickAssist/README.txt).

2.1 Limitations

This release does not support the following:

- Static Deflate Stateful compression/decompression
- Dynamic Deflate Stateful compression/decompression
- Symmetric (bulk) cryptography algorithms like Advanced Encryption Standard (AES)
- Fallback for Cryptography services
- Virtualization using SR-IOV with Microsoft® Hyper-V and Windows* QAT VF devices

2.2 Known Issues

The known issues and resolved issues with this software release are listed below:

2.2.1 WCAT workload has ECDHE curve25519 failure

Title	WCAT workload has ECDHE curve25519 failure
Reference #	QATE-38965
Description	The `ECDHE` curve "`curve25519`" is the default curve for `ECDHE` in Windows*. The WCAT workload on IIS fails to authenticate when using Intel® QAT to run `ECDHE` and RSA, using the default curve preference order.
Resolution	Two possible resolutions: Change the default `ECDH` curve in Windows* to be a curve that is supported by Intel® QAT. The result is that ECDH is executed on Intel® QAT (but not using `curve25519`). Use the `CPMCNGInstaller` tool to unregister ECDH provider for QAT. The result is that `ECDH` is executed on the CPU using the default `curve25519`.
Affected OS	Windows* Server 2019/2016
Driver/Module	QAT IA – Crypto

2.2.2 Parcomp unable to read > 1GB file for compression

Title	Parcomp unable to read > 1GB file for compression
Reference #	QATE-40170
Description	Parcomp is unable to read large files (test file was 2.2 GB) for compression. Thus, compression would fail.
Resolution	When writing an application with QATZIP, chunk the file into at most 1GB increments.
Affected OS	Windows* Server 2019/2016
Driver/Module	QAT IA - Compression

2.2.3 Cngtest does not validate fallback operations are working correctly

Title	`Cngtest` does not validate fallback operations are working correctly
Reference #	QATE-38968
Description	Currently, the `Cngtest` does not include tests to validate the fallback to the Microsoft* provider works for unsupported algorithms and curves. Environment: Supermicro* X11 Intel® QAT Microserver with 2x 37C8 devices Windows* Server 2016 The `Cngtest` cannot validate fallback operations. If encryption is performed by SW, it needs to ensure that decryption can be performed by the Intel® QAT HW or vice-versa.
Resolution	There is currently no workaround for this, and it may be added in a future release.
Affected OS	Windows* Server 2019/2016
Driver/Module	QAT IA – Crypto

2.2.4 Sending malicious data to the VF may result in PCIe Push/Pull Parity Error or NMI

Title	Sending malicious data to the VF may result in PCIe Push/Pull Parity Error or NMI
Reference #	QATE-41844
Description	When sending malicious or malformed data to the QAT VF driver, especially in kernel mode operations, you may see a PCIe Push/Pull Parity error or in the worst and rare case, a NMI error.
Resolution	There is currently no workaround for this, it is a hardware limitation.

Title	Sending malicious data to the VF may result in PCIe Push/Pull Parity Error or NMI
Affected OS	Windows* Server 2019 Hyper-V and newer
Driver/Module	QAT IA

2.2.5 Compression may randomly fail (qzSetupSession error) after driver installation.

Title	Compression may randomly fail (qzSetupSession error) after driver installation.
Reference #	QATE- 71057
Description	After a driver installation, there may be a rare occurrence where compression operations will fail qzSetupSession continuously.
Resolution	Restart the system.
Affected OS	Windows* Server 2016 and newer
Driver/Module	QAT IA - Compression

2.2.6 Multiple concurrent PF/VF comms operations will put VF device in bad state.

Title	Multiple concurrent PF/VF comms operations will put VF device in bad state.
Reference #	QATE- 67282
Description	Operations that require multiple concurrent PF/VF communications may result in putting the VF in a bad state. Such operations generally include bring the VF up or down simultaneously (such as during a precisely timed driver installation across multiple VM's and VF's).
Resolution	Restart the affected VM.
Affected OS	Windows* Server 2019 Hyper-V and newer Linux* using KVM on kernel 4.04 and newer
Driver/Module	QAT IA

2.2.7 Length headers are not populated for gzipext in SW Fallback on Linux* QATzip.

Title	Length headers are not populated for gzipext in SW Fallback on Linux* QATzip.
Reference #	QATE-74339
Description	When using gzipext on Linux* QATzip version 1.0.4 in software fallback mode, the gzipext header will not populate the length field. This may result in Windows* QATzip not being able to decompress this using gzipext (qatgzipext in parcomp sample application).
Resolution	Workaround is to use gzip decompression (qatgzip for parcomp).
Affected OS	Windows* Server 2016 and newer
Driver/Module	QAT IA - QATzip

2.3 Resolved Issues

2.3.1 Default curve order for elliptic curves not supported by QAT

Title	Default curve order for elliptic curves not supported by QAT
Reference #	QATE-37219

Description	The default curve order on Windows* when using cipher suites with ECDHE is as follows:
	`curve25519 NistP256 NistP384`
	Since `curve25519` is not supported by Intel® QAT, cryptography operations will fail when using cipher suites with `ECDHE`.
	However, the `NistP256` and `Nist384` curves are supported by
	Intel® QAT, so if the curve priority order is changed as shown below, cryptography operations when using cipher suites with `ECDHE` will succeed:
	`NistP256`
	`NistP384 curve25519`

Resolution	Modify the default ECC Curve Order as below: Launch the Group Policy Editor: `gpedit.msc` Open Computer `Configuration/Administrative Template/Network/ SSL Configuration Settings` Double-click ECC Curve Order (in the right pane) Click **Enabled** Edit the ECC Curve Order in the priority order described above. Click **'Apply'** and exit the application

Title	Default curve order for elliptic curves not supported by QAT
Affected OS	Windows* Server 2019/2016
Driver/Module	QAT IA – Crypto

2.3.2 QAT driver and service are sometimes not removed after uninstallation

Title	QAT driver and service are sometimes not removed after uninstallation
Reference #	QATE-65388
Description	After uninstalling the QAT driver using control panel or the command line, the driver file icp_qat.sys and the associated Windows* service do not get removed properly. Upon reboot, devices in Device Manager will show error code 32.
Resolution	In device manager, right click and uninstall with remove files checked, manually remove icp_qat.sys file.
Affected OS	Windows* Server 2019
Driver/Module	QAT IA – General

2.3.3 Cannot disable driver while parcomp (compression) is running

Title	Cannot disable driver while parcomp (compression) is running
Reference #	QATE-36847
Description	When running `parcomp` stress tests, you cannot disable all `37c8` Intel® QAT devices. Doing so may cause the driver to disable to spin until the `parcomp` process is stopped. The issue has been observed mostly on Skylake-D systems. Environment: Supermicro* X11 Intel® QAT Microserver with 2x 37C8 devices Windows* Server 2016 W.1.1.0-0029 drivers Steps: Run a `parcomp` stress test. Automation runs with the following parameters: `.\parcomp.exe -i C:\\CompressionFiles\silesia -o C:\\CompressionFiles\compress -p qat -Q -t 6 -k 4096 j 60 -x 2 -n 200` Disable `37c8` devices, one at a time until no more left (sometimes may occur on the first `37c8` disable). Last, disable should keep spinning until `parcomp` thread is stopped.

Title	Cannot disable driver while parcomp (compression) is running
Resolution	This is resolved with the QAT1.5.0-0007 release.
Affected OS	Windows* Server 2019/2016
Driver/Module	QAT IA – Compression

2.3.4 Windows* Setup /passive install has crypto failures

Title	Windows* Setup /passive install has crypto failures
Reference #	QATE-38404
Description	When you use the '/passive' option for installation, it seems that Crypto will fail after a few iterations.
Resolution	This is resolved with the QAT1.5.0-0007 release.
Affected OS	Windows* Server 2019/2016
Driver/Module	QAT IA – Crypto

2.3.5 System crash when calling QATZip API function incorrectly

Title	System crash when calling QATZip API function incorrectly
Reference #	QATE-71816
Description	Calling QATZip API function QzCompress directly without properly setting up a session via qzSetupSession, and then calling either QzClose or QzTeardownSession can cause a system crash.
Resolution	This is resolved with the QAT1.5.0-0007 release.
Affected OS	All supported Windows* OS
Driver/Module	QAT IA – Compression

§

3 Software Installation

The release package includes the Setup.exe installation application. Use this application to install the package on the targeted OS. For more information on how to install the package, refer to the Readme file included in the package:

```
.\quickassist\README.txt
```

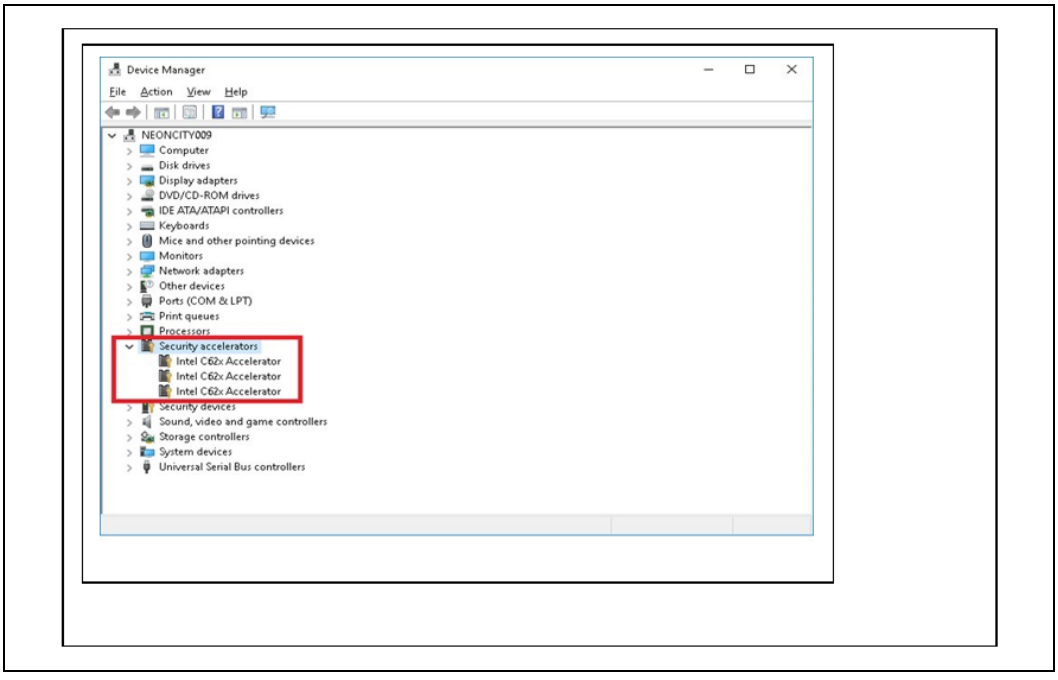
Upon completion of the installation, the `README` text file can also be found in the following folder:

```
<Program Files>\Intel\Intel(R) QuickAssist Technology
```

Note: For those customers that had already have installed the previous version of the Intel® QAT software package, uninstall it and reboot before installing this new production package.

To ensure software installation completed successfully and that Intel® QAT devices are functional, refer to Figure 1. The screenshot lists three "Intel® C62x Accelerator" devices under the "Security accelerators" PNP Classification.

Figure 1. Device Manager with Intel® QuickAssist driver installed in Microsoft® Windows*



§

4 Test Applications

4.1 Compression Test Application

A compression test application, parcomp, is included in this package. For more information on how to use the parcomp application, refer to the Readme file included in the package. You can find the README file in the following folder upon completion of the installation:

```
<Program Files>\Intel\Intel(R) QuickAssist Technology
```

4.2 Cryptography (PKE) Test Application

A cryptography test application for PKE operations, Cngtest, is included in this package. For more information on how to use the Cngtest application, please refer to the Readme file included in the package. You can find the README file in the following folder upon completion of the installation:

```
<Program Files>\Intel\Intel(R) QuickAssist Technology
```

§