IntervalZero

Product Release Notice

RTX64 4.4

General Availability Release Date

May 16, 2023

Product Overview

RTX64 4.4 is the latest 64-bit version of IntervalZero's market-leading hard real-time software products. This release provides several new features, usability improvements, and resolved issues.

To see a full list of all new features and a detailed list of new APIs added and issues resolved, please refer to the product release notes, found in the online RTX64 4.x Help.

Key Features

Runtime

General

- RTX64 now supports using .NET Standard 2.0, which allows for Windows application development with .NET/.NET Core/.NET Framework. (11101)
- 13th generation Intel® Processors have been added to our QA lab and are confirmed to work with RTX64. (11463)

Installation

- RTX64 Subsystem now checks if hypervisor features are enabled and if found will not start. The Runtime install allows users to override this check to enable installation and running of the Subsystem on virtual machines for development purposes. (11079)
- The RTX64 Runtime installer will now output a settings file on upgrade in case you need to revert/restore subsystem setting at some point in the future. (7413)

Network and Drivers

- Incorporates support for Realtek adapters, through the provided NAL driver RtNalRtl.rtdll. This driver provides support for the Realtek PCIe GBE Family Controller.
- Advanced functionality has been added to the RtNalIGC driver, which supports the Intel 2.5 Gigabit Ethernet Controllers I225/I226. These features include:
    - Interrupt Moderation (10777)
    - Multiple queues (10775)
    - Hardware time stamping (10776)
    - Forcing speed and duplex (10867)
    - MSI-X Single Vector and MSI-X Multi Vector support (11356)
- The RTX64 Virtual Network Interface is now an optional feature during installation. It is selected to be installed by default. (10359)

Tools and Utilities

- Adds functionality to the RTX64 Control Panel:
    - On the Configure and control the network page there is a new field to **Set the Stack timer execute priority** this sets the priority of the RT-TCP/IP Stack's second real-time timer thread, which executes functions for expired timers. (9915)
    - Adds a **Speed & Duplex** setting to the Manage Interfaces page, which specifies the method used to establish the link. (10189)
    - Adds support for additional Interrupt types of MSI-X Single Vector and MSI-X Multi Vector on the Manage Interfaces page. (11356)
- Adds functionality to the RtssPing utility that allows you to specify the source IP address for the ping request. (9365)
- Expanded RTX64 Analyzer output to include a Power Configuration section. (6678)

Software Development Kit

Application Development

- Adds support for Windows SDK for Windows 11 (10.0.22621). This Windows SDK can now be selected in Visual Studio when building an application. (11045)

Network and Drivers

- Includes updates to source files for the RtNalIGC driver.
- Adds source files for the RtNalRtl driver.

Real-Time APIs

- Adds a new function RtStopComponent which stops a specified Subsystem component. (10689)
- Adds support for RtNalTransmitEx and RtndTransmitEx to the RtNalE1000 driver. (10874)

Native Framework APIs

- The Network Interface related APIs now detect duplicate IPv6 addresses. (9887)
- Adds a new function RtfwGetRtssProcessorsActual, which returns the actual number of processors being used by the RTX64 Subsystem. (11030)
- Adds new function RtfwGetEnabledProcessors, which returns the total number of processors that are enabled on the system. (11031)
- Adds new member **SpeedDuplex** to enumeration RTFW_NETWORK_INTERFACE, which specifies the method used to establish the Ethernet link. (10191)
- Adds support for MSI-X Single Vector and MSI-X Multi Vector to the RTFW_INTERRUPT_TYPE enumeration. (11354)

Managed Framework APIs

- Adds new method GetNetworkInterfacesStatus to Managed class IntervalZero.RTX64.Control.Subsystem, which queries the status of each enabled network interface. (9089)
- The Network Interface related APIs now detect duplicate IPv6 addresses. (9887)
- Adds a new parameter **speedDuplex** to the CreateNetworkInterface method in Managed class IntervalZero.RTX64.Config.NetworkInterface, which specifies the method used to establish the Ethernet link. (11122)
- Adds new sub-class RTX64Win32Exception to the Managed RTX64Exception class. (7122)
- Adds MSI-X Single Vector and MSI-X Multi Vector to the IntervalZero.RTX64.Config.NetworkInterface.InterruptType enumeration.(11355)

Issues Resolved

Runtime

Subsystem

- Resolves an issue where under stress conditions, RtCloseHandle of an IPC object (Mutex, Event, or Semaphore) from a Proxy Thread could cause a Windows hang. (11227)

- Resolves an issue where RtTerminateProcess was not properly cleaning up process handles under stress conditions. (11416)
- Resolves an issue where under stress conditions, some Rtprintf data was not displayed in the console window. (10935)

Network and Drivers

- Resolves an issue where the Device ID and Vendor ID listed in the DriverList.xml file for the **Intel(R) Ethernet Controller I225 (blankNVM) (RTX64)** device was incorrect. (11063)

Tools and Utilities

- Resolves an issue where the RTX64 Console could not be cleared when a large amount of data was printed. (10829)
- Resolves an issue where links in the RTX64 Activation and Configuration Utility redirected to the wrong page in the RTX64 Control Panel. (10459)
- Resolves an issue where the RtssPing utility displayed output in multiple console windows. (9888)
- Resolves an issue where the RT-TCP/IP Stack did not work correctly with a specified server link-local IPv6 address. (9366)
- Resolves an issue where attempting to stop the Subsystem while KSRTM is running resulted in a system crash. (6922)
- Resolves an issue where RtssKill output truncated PIDs that were more than 5 digits. (10916)
- Resolves an issue where the RTX64 Task Manager had elevated privileges when launched from the RTX64 Control Panel. (10090)
- Resolves an issue where the Manage interfaces and filters page in the RTX64 Control Panel was inaccessible on Turkish-language versions of Windows. (9261)
- Resolves an issue where the status of the NAL and TCP/IP Stack did not update properly in the RTX64 Control Panel when auto-start was disabled. (10462)
- Resolved an issue where the RTX64 Activation and Configuration Utility succeeded unexpectedly in situations where it shouldn’t have. (9566)

Software Development Kit

Application Development

- Resolves an issue where attempting to compile a C program that included header file RtfwAPI.h in more than one source file resulted in a duplicate definition error. (11169)
- Tools and Utilities
- Resolved an issue where the RTX64 Activation and Configuration Utility succeeded unexpectedly in situations where it shouldn’t have. (9566)

Real-Time APIs

- Resolves an issue where RtGetModuleFileNameEx failed unexpectedly in Ansi/Windows configurations with a smaller buffer than what the module path returned. (10458)
- Improves error messaging to specify invalid parameters for RtnInstallStaticRoute. (10236)
- Resolves an issue where RtIsTcpStackLicensed returned an incorrect error code in a Windows build configuration when the TCP/IP Stack was not licensed. (10349)
- Resolves an issue where the NtNalIGB and RtNalE1000 drivers returned an incorrect value for MaxPacketSize when using RtndInitializeInterface. (10866)
- Resolves an issue where RtNalTransmit and RtNalReceive incorrectly returned error code ERROR_IO_DEVICE in some scenarios. (10941)
- Resolves an issue where RtNalEnumInterfaceInfo incorrectly returned false with error code ERROR_NO_MORE_ITEMS in some scenarios. (11199)

Native Framework APIs

- Resolves an issue where the RTX64 Native Framework (RtfwAPI.dll) incorrectly reported that the NAL needed to be restarted when RtfwSetNetworkVerbosity was used to set the network verbosity to a value it was already assigned. (10436)
- Resolves an issue where RtfwCreateNetworkInterface and RtfwModifyNetworkInterface would accept invalid IPv4 addresses. (10892)

Managed Framework APIs

- Resolves an issue where the RTX64 Managed Code Framework (IntervalZero.RTX64.dll), and the Control Panel would incorrectly report that the NAL needed to be restarted when IntervalZero.RTX64.Config.Network.Verbose was used to set the network verbosity to a value it was already set to. (10436)

Samples

- Resolves an issue where APIs exported from the Rtdll sample failed when the sample was explicitly loaded due to C++ name mangling. (10957)

Availability

RTX64 4.4 is available beginning May 16, 2023, through Partners and by contacting Sales: sales@intervalzero.com.

We look forward to any comments and feedback. If you have any recommendations, or wish to suggest any product enhancements, please contact Product Management at: productmanagement@intervalzero.com.