

Product Release Notice

RTX64 4.2

General Availability Release Date

January 14, 2022

Product Overview

RTX64 4.2 is the latest 64-bit version of IntervalZero's market-leading hard real-time software products. This release provides a number of 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

RTX64 Runtime

General

- Adds support for Windows 10 Update Version 2105 (21H1).

Subsystem

- Adds support for collecting pre-trigger events during a monitor session. (10023)

Network and Drivers

- Adds support for configuring Enable Receive Polling in the Windows Virtual RTX64 Ethernet Adapter properties. (9550)
- Upgrades the Treck Stack to version 6.0.1.69 to resolve security vulnerabilities. (9374)

- Adds support for multiple devices to the RtNallPCH driver. See the *RTX64 Supported NICs* document for details. (9617)

Tools and Utilities

- Adds settings to the Change monitoring settings page in the RTX64 Control Panel that allows for configuration of pre-trigger events, maximum disk space to use, and default stop behavior. (9822, 10016)
- Adds settings to the Set session triggers page of the Monitor Utility that allow you to change the maximum number of events to collect and optionally collect pre-trigger events. These settings override their equivalent default settings in the RTX64 Control Panel. (9822, 10017)
- Adds new local memory MSpace fields to the text output for the Process Create monitoring event. (9156)
- The RTX64 Control Panel now displays whether the Monitoring feature is Disabled or Enabled. When monitoring is enabled, the Control Panel also displays its current state: Stopped, Started, or Paused. In addition, it displays whether monitoring is configured to be enabled and started with the Subsystem. (7703)
- Adds functionality to the RTX64 Console window that allows you to save the console output to a file. (10108)

RTX64 SDK

General

- RTX64 SDK no longer requires a dongle for activation on a Virtual Machine. (9816)
- Updates the version of Tracealyzer by Percepio installed with RTX64. (10056)

Application Debugging

- Adds support for hard-coded (embedded) breakpoints. (3751)
- Adds functionality that causes the Visual Studio debugger to break automatically when a process being debugged is about to starve. This allows you to further inspect the process memory using either the Visual Studio IDE or WinDbg extensions. (9943)

Real-Time APIs

- Adds new Real-time APIs that set or retrieve the number of pre-trigger events and the maximum number of events to save, respectively, per monitoring session.
- Adds new constants to Real-time enumeration RTX64_MONITOR_CONTROL_OP to set and retrieve the number of pre-trigger events and the maximum number of events to save, respectively, per monitoring session.

Native Framework APIs

- Adds function RtfwGetRTX64DevicesEx and structure RTFW_DEVICEEX which retrieves the properties of all devices owned by RTX64.
- Adds function RtfwFreeDeviceMemory, which frees the memory associated with the array of RTFW_DEVICEEX structures pointed to by parameter *pRTX64Devices*.
- Adds new members to Native structure RTFW_MONITOR_CONFIGURATION to set up monitoring collection boundaries and pre-triggers. (9822, 10051)

Managed Framework APIs

- Adds new properties to Managed class MonitorEventProcessCreate which provides information on the process's internal and external MSpace configurations (9156).
- Adds a new property and several new methods to Managed class IntervalZero.RTX64.Monitor.Subsystem to configure monitoring data collection and pre-triggers. (9822)

Samples

- Added Visual Studio 2017 project files for the RtkIPC sample. (10031)

Issues Resolved

RTX64 Runtime

General

- Resolves an issue where uninstalling RTX64 Runtime from a system on which RTX64 Vision Runtime was installed resulted in removal of RTX64 Vision Runtime registry settings. (9993)

Activating and Configuring

- Resolves an issue where the RTX64 Activation and Configuration Utility sometimes crashed when the user attempted to set the RTSS boot configuration after disabling Hyper-Threading. (10066)

Subsystem

- Resolves an issue where debugging an RTSS process sometimes caused the Subsystem to become unresponsive when the RTX64 Watchdog timer was configured to monitor for runaway threads. (9930)
- Resolves a group permissions issue that allowed members of the RTX64Users group to access RTX64Administrators files. (9952)

Network and Drivers

- Resolves an issue where the RtNal10GB driver did not fully support the Intel® X550 T 10GB Network Connection (0x1563) device. (9002)
- Resolves an issue where the RtNal10GB driver continued to transmit frames even after the Ethernet cable was disconnected from the NIC. (9360)
- Resolves these issues with the RtTcpipClient sample:
 - Resolves an issue where the sample sometimes erroneously matched a sent packet with a response to a previous packet. (9742)
 - Resolves an issue where the sample could not bind IPv6 addresses to the Client socket. (9577)

Tools and Utilities

- Resolves an issue where data-less custom events appeared as a blank line in Monitor Utility output. (10154)
- Resolves an issue where some non-applicable Control Panel settings, such as *Interrupt Type*, were accessible when the RTX64 Virtual Network Interface was selected. Such non-applicable settings are now grayed-out. (10039)
- Resolves an issue where RtssRun returned an error when passed an argument for a directory path ending in a backslash (\). (10015)
- Resolves an issue where Network Abstraction Layer (NAL) and/or TCP/IP Stack output appeared in separate consoles when the Subsystem was configured to *Display a separate*

console window for each real-time process. NAL and TCP/IP Stack output now appears in a single system console that closes when the Subsystem stops. (9885)

- Resolves an issue where the keyboard shortcuts used to scroll the RTX64 Console window did not function correctly. (9436)
- Improves error messaging for settings on the Configure real-time application output page in the Control Panel. (9940)
- Resolves an issue where some device-specific information could not be enumerated by the Real-time Framework (RtfwAPI.dll). As a result, an interface could not be added through the RTX64 control panel for some RTX64-converted network devices. (9912)
- Resolves an issue where RTX64 Analyzer output contained information for an obsolete tool. (9512)

RTX64 SDK

Installation

- Resolves an issue where Debugger Extension files remained on the system after the RTX64 SDK was uninstalled. (10172)
- Resolves an issue where uninstalling RTX64 SDK from a system with multiple RTX64 SDKs installed also removed StampTool from the system. (10053)

Tools and Utilities

- Resolved an issue where some memory-related monitoring events were not being generated, causing degraded memory graphs in Tracealyzer. (9899)
- Resolved an issue where Tracealyzer displayed a valid Subsystem SRI as *Unknown SRI*. (9593)

Supported Windows APIs

- Resolves an issue where parameter *dwInitialSize* in Windows function HeapCreate was not ignored as intended. (10073)

Real-Time APIs

- Resolves an issue where the description for error code RT_ERROR_MONITORING_NOT_ENABLED was incorrect. (10313)

- Improves error messaging in Real-time network device functions `RtndInitialize`, `RtndConfigure`, and `RtndUpDown`. (10282)
- Changes function `GetAdaptersAddresses` to return `ERROR_BUFFER_OVERFLOW` when parameter `AdapterAddresses` is `NULL`, matching the Windows version of the API. (10176)
- Resolves an issue where Real-time network device function `RtndRequest` reported incorrect values for disconnected devices. (9826)
- Fixes routing table issues in Real-time network function `RtnDisplayRoutingTable`. (9011)
- Resolves an issue where TCP/IP Stack interfaces in `RtStartComponent` could not be started if the Network Abstraction Layer (NAL) was stopped. (9327)

Native Framework APIs

- Improves error reporting for invalid TCP/IP Stack heap allocation by adding a new error code, `RT_ERROR_TCPIP_HEAP_MSPACE_INVALID`, which appears when the TCP/IP MSpace size is not at least 2176 KB larger than the MSpace memory allocated to the TCP/IP Stack heap when *Auto expand MSpaces* is disabled. (9537)

Samples

- Resolves build errors with the `WindowsRTX64UsingSTL` sample on Visual Studio 2019 v16.10.2. (10118)
- Resolves an issue where compiled files for the `RtNalPCH` driver ended up in the `RTSSDebug` folder when the driver source was built with the `RTSSRelease` configuration. (9956)
- Adds information on supported drivers and functionality to the `Readme.txt` files packaged with the Network Abstraction Layer (NAL) samples. (9741)
- Resolves an issue where latencies and system hangs sometimes occurred when running the following samples on certain systems: `SRTM`, `mSRTM`, `kSRTM`, `RtPerfMonitor`, `Shutdown`. (9702)
- Resolves an issue where the `NalMultiplePacketTx` sample worked when passed incorrect flags. (8737)

Availability

RTX64 4.2 is available beginning January 14, 2022 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.