



Release Notes

Barcode Scanner SDK for Linux v4.4
January 2024

Contents

Contents.....	1
Overview	1
Device Compatibility.....	2
Supported COM Protocols.....	3
Version History	3
Components.....	9
Installation Options (Supported Distributions)	13

Overview

These release notes describe the Software Development Kit (SDK) for Linux, including its functionality, supported scanners and supported distributions.

Functionality,

1. CoreScanner core components and drivers (USB, Imaging).
2. JPOS driver.
3. Development libraries and header files for application developers.
4. Test and Sample utilities
 - a. Standard I/O Console based sample application for CoreScanner (C++).
 - b. GUI sample application for CoreScanner (C++).
 - c. IBM JPOS sample application (Java).
 - d. GetAndSet Utility.



CoreScanner Driver

1. Supported Com Interfaces - See “Supported Com Protocols” section of document
2. Programmatic API Capabilities
 - a. Com protocol switching
 - b. Get device topology
 - c. Plug and Play (PnP) detection
3. Data Capabilities
 - a. Barcode data support
 - b. Scale data for MP products
 - c. Image capture support including view finder
 - d. Video capture support
 - e. Intelligent Document Capture (IDC)
4. Sample Apps with Source Code
 - a. C++ sample app – Graphical User Interface (GUI) based
 - b. C++ console app – Command line control based
5. Firmware update
6. Debugging Tools
 - a. Logging / monitoring support

JPOS Driver

1. XML configuration of scanner attributes.
2. Access to Zebra specific functionality through Direct I/O calls
3. Sample applications
 - a. JPOS sample app
 - b. Direct I/O sample app
4. Logging support
5. JPOS exceptions for failures (Troubleshooting)

JPOS Remote Management Solution

1. New tool to perform remote management activities via JPOS driver.
2. Command line tool and executed by a shell script
3. Logging support
4. Exceptions or error message will be printed to the console during the tool execution.

Host Device Details

See the “Installation Options (Supported Distributions)” table – shown below.

For the latest SDK updates, please visit [Zebra Scanner SDK](#)

For support, please visit <http://www.zebra.com/support>.

Device Compatibility

For the compatible devices list, please visit following page.



<https://www.zebra.com/us/en/support-downloads/software/developer-tools/scanner-sdk-for-linux.html>

Supported COM Protocols

SDK supported communication protocols include:

	Query Assets Information	Host Switching	Imaging and Video	Faster Firmware Update	Management and Firmware Update	Barcode	JPOS Driver
IBM Table-Top USB	X	X			X	X	X
IBM Hand-Held USB	X	X			X	X	X
IBM OPOS - IBM Hand-held USB with Full Scan Disable							
HID Keyboard Emulation		X					
USB CDC Host							
Symbol Native API (SNAPI) with Imaging Interface	X	X	X	X	X	X	X
Symbol Native API (SNAPI) without Imaging Interface	X	X			X	X	X
Wincor-Nixdorf RS-232 Mode B							
Simple Serial Interface (SSI) over RS232							
Simple Serial Interface (SSI) over Bluetooth Classic							
Simple Serial Interface (SSI) over Bluetooth Low-Energy (BLE)							
Simple Serial Interface (SSI) over MFI							

Version History

Version 4.4.1-38 – 01/2024

1. Command Line Utility (**CLU**) – Added new command to **retrieve asset information** based on scanner ID. It returns host mode, serial number, firmware version and date of manufacture of the scanner.



2. Bug Fix – Rarely used scenario addressed – **Full read/write permission to Corescanner log folder** now available. This enables a user to read/write files to the log folder. Access permission now exists for the general user of the zebra log folder regardless of its form of creation (both created through service/created during installation).

Version 4.4.1-36 – 10/2023

3. Bug Fix – **Record count** sent with firmware update progress events functions as expected; Intermittent issue addressed.
4. Bug Fix – Changed **default value** of “**bulktransfer_time_out**” in configuration file to resolve scanner reboot issue seen during firmware update.
5. Bug Fix – Command Line Utility (CLU) - Resolved a **firmware update failure** in IBMHID and IBMTT modes, seen with particular versions of scanner firmware.

Version 4.4.1-34 – 07/2023

1. Enhanced JPOS driver
 - a. Bug fix – Fixed issue that incorrectly appended a Label ID for NCR label with barcode type ISSN.
 - b. Bug fix – Fixed issue relevant to error arguments (locus and response) in JPOS read weight error events.
 - c. Sample App Security fix – Updated library “xercesImpl.jar” used in JPOS Sample application from v2.11.0 to v2.12.2 to resolve security vulnerabilities exposed from v2.11.0.
 - d. Sample App Bug fix – Device enable button now gets updated upon enabling auto device enable (button) in JPOS scale.
 - e. Sample App Bug fix – Barcode name now correctly displayed for Han Xin Code.
2. Command Line Utility (CLU) – Bug Fix – Addressed memory allocation issue with “reboot” and “getattribute” commands.

Version 4.4.1-32 – 04/2023

1. Active Focus Manager (AFM) Module - Alpha release for customer evaluation. Available for download separately, not bundled with the SDK.
2. Enhanced Driver Data Formatting (DDF) module – Alpha release for customer evaluation - Added support to incorporate ATL key combinations using the Driver Data Formatting (DDF) module. This functionality enables an ALT key combination to be added to barcode data when using Simulated HID Keyboard.
 - a. Configuring this capability is located in the CoreScanner configuration xml file.
 - b. An example of this capability is sending "ALT [+ Data + Enter"; also "ALT [+ Data + TAB".
 - c. This initial implementation can only support sending ALT + one key sequence like “ALT [“.
 - d. This new ALT key capability supports modifying data using a Prefix only (Suffix with ALT key is not supported).



3. Command Line Utility (CLU) – Initial release. This utility provides scripting control of Zebra scanner. Supported functionality includes:
 - a. Reboot of a scanner
 - b. Parameter Control
 - i. Param Get & Set
 - ii. 123Scan config file consumption
 - c. F/W update
4. Enhanced JPOS driver
 - a. Bug fix in Sample App – Fixed error message incorrectly displayed in JPOS Sample application when Zero Scale command is called and weighing item below 30 grams.
 - b. Update JPOS driver to fire Scale status update events whenever a status updates and a weight change is detected.
 - c. Bug fix in Sample App – Made the display format for scale weight consistent in the sample application for Read Weight, Live Weight and Direct IO NCR Live Weight calls.
 - d. Bug fix in JPOS Sample App – Fixed application lockup if enabling both live weight and auto disable simultaneously.

Version 4.4.1-31 – 01/2023

1. Enhanced IoT Connector - Added support for C98. Note C98 is a C++ compilation standard.
2. Added IoT connector support for following distributions.
 - a. RedHat 7.4 64bit
 - b. CentOS 7.9 64bit
 - c. Ubuntu 14.04 64bit
3. Added “Simulated HID Keyboard” support to CoreScanner.
4. Added “DDF” (Driver Data Formatting) support to CoreScanner.
 - a. DDF enables driver-based data formatting like adding an Enter or Tab key to be appended to scanned data before its transmission to a host.
 - b. DDF (Driver Data Formatting) is different than ADF (Advanced Data Formatting). DDF is performed by Zebra’s CoreScanner Driver versus ADF is performed within the scanner’s firmware.
 - c. DDF is supported over “Simulated HID Keyboard” using the following comm interfaces: SNAPI, IBM HandHeld, IBM TableTop.

Version 4.4.1-30 – 10/2022

1. Enhanced SDK Sample Application
 - a. Added Page motor support to the SDK Sample Application.
 - b. Now the Scale tab is disabled when the connected device is not a Zebra scale.
2. Bug fix – Resolved inconsistency in “get all RSM values” responses.
3. Enhanced JPOS driver
 - a. Enhanced driver to support NCR requested “HealthCheck” label IDs.
 - b. Bug fix – “Get Error Response” API now returns the correct error on Read Weight in Scale.
 - c. Minor UI optimizations in the JPOS Sample Application for Windows.



4. Enhanced firmware update in non-bulk mode to generate continuous firmware update events in SDK. Note bulk mode already works this way.

Version 4.4.1-29 – 05/2022

1. Enhanced JPOS driver
 - a. Bug fix, ReadWeight events now reported correctly when DataEventEnabled is false and LiveWeight is true.
 - b. Enhanced driver to support all NCR requested “ScanData” label IDs.

Version 4.4.1-28 – 04/2022

1. Enhanced IoT Connector capabilities - Added support for logging scanner details where Global Unique Identifier (GUID) is invalid.
2. Enhanced Toshiba Global Commerce Solutions (TGCS) POS system support
 - a. JPOS driver enhanced to support System Management Information calls from TGCS POS systems
 - i. Corescanner enhanced to support TGCS’ CIM Service Provider = “UPOS_BarcodeScanner” queries
3. Bug fix – Resolved issue in firmware update failure with DS9908 and DS8178 scanner models.

Version 4.4.1-27 – 01/2022

1. Logging Agent renamed “IoT Connector”.
2. Enhanced JPOS driver - Fixed rarely seen JPOS statistics retrieval issue on scanners.

Version 4.4.1-24 – 10/2021

1. Added support for SLES 15 SP3 (64bit), SLES 12 SP5 (64bit), Ubuntu 21.04 (64bit), Debian 10.10 (32bit and 64bit).
2. Enhanced JPOS driver
 - a. Added support to execute DirectIO commands without claiming the device.
 - b. JPOS sample application enhancement to display the “Live Weight” and the logs on live weight status update events.
 - c. Enhanced logging in JPOS driver including access to barcode data, power state, scale weight, and what API calls have been made.
3. Enhanced Logging Agent capabilities
 - a. Added support for logging of operating system environment variables like “host PC name”. The environmental variable check is performed real time on each logging occurrence
 - b. Added support for real time logging via a JSON call to cloud-based consoles like Splunk.

Version 4.4.1-19 – 04/2021

1. Added support for CentOS 7.9 (64bit).
2. Enhanced JPOS driver v1.14 to include support for PNP events for both Scanner and Scale.
3. Enhanced the SDK’s UI sample application to include Scale functionalities (Read Weight, Zero Scale, Reset Scale).



4. Bug fix – The JPOS driver v1.14 for code 49 now returns the correct Symbology types when the Zebra scanner is in IBM Hand-Held or IBM Table-Top host modes.

Version 4.4.1-16 – 01/2021

1. Enhanced log file - Resident scanner firmware update status (STATUS_FW_SWCOMP_RESIDENT) added to the log files.
2. Enhanced console-based sample application for CoreScanner - The console app now shows call success and failure (Logging API return values) and output XML.
3. Bug fix – The MS4717 now properly switches from HIDKB host mode to another com protocol and back.
4. Bug fix – The CoreScanner now properly returns the valid status codes for “invalid firmware file” and “failed to read DAT file”.
5. Bug fix – The SDK sample application now properly returns an error when launched without running the CoreScanner.

Version 4.4.1-15 – 10/2020

1. Added support for Open SUSE 15.2 (64bit).
2. Common JPOS driver bundled in both Linux and Windows SDKs. With this common code driver, the same features are now supported across both Linux and Windows.
3. Removed Apache Xerces XML parser dependency from the Zebra JPOS Service Object (SO).

Version 4.4.1-11 – 07/2020

1. Added supports for Ubuntu 20.04 (64bit) and CentOS 8.1 (64bit).
2. Logging Agent bundled with Linux SDK.
 - a. The Logging Agent allows a 3rd party management console to track scanner information including the scanner’s health by parsing a Logging Agent generated log file.
 - b. The Logging Agent is configurable for example it can be output for all devices (1,000s of log files) or just devices that meet a specific condition like failed a firmware update (a few log files).
 - c. The Logging Agent can document the following information:
 - i. Asset information
 - ii. Statistics for example battery charge level or UPCs scanned
 - iii. Firmware failures and or firmware success
 - iv. Parameter value(s) changed. Achieved by tracking parameter 616 (config file name changed to “Modified”)
3. Added support for composite barcode types.
4. Bug fix – Sample application freeze resolved on scanner PNP.
5. Bug fix – Sample application RSM attributes list shows the correct permissions for each attribute.

Version 4.4.1-8 – 04/2020

1. Added option for enabling new Zebra HID API instead of using native OS provided HID API. Zebra HID API enabled through a configuration setting, switching from libusb to hidraw.



2. Add support for additional distributions: Raspbian 8 and ARM7L (32-bit ARM processor). Note this Linux OS runs on Raspberry PI 3.0 hardware.
3. Bug fix – Resolved app responding delay by sending the Direct I/O commands using the HID API directly from the comlib to the scanner.
4. Bug fix - Synchronization block introduce when consuming the events from CoreScanner to JPOS driver. Otherwise, the single core POS system is missing different events from different logical devices (JPOS)

Version 4.4.1-4 – 01/2020

1. Added ability to discover, communicate and manage a third level cascaded device. The SDK now supports connecting cordless scanners like a DS2278 and cradle to a MP7000. The MP7000 is connected to the host.

Version 4.4.1-2 – 10/2019

1. Added “DotCode” and “GridMatrix” symbology support to JPOS and SDK Sample Applications. Note the Linux SDK already supported these symbologies.
2. Added Java 12 UI enhancements to JPOS Sample Application. Note no change required in the JPOS driver.

Version 4.4.0-18 – 08/2019

1. Added Scale Live Weight event support for Multi-Plane scanners.
2. Vendor specific capability, ‘CapStatusUpdate’ enabled in JPOS Scale driver.
3. Implemented ‘StatusNotify’ property in JPOS driver.

Version 4.4.0-15 – 06/2019

1. Added get next Attribute RSM command support in CoreScanner.

Version 4.4.0-14 – 06/2019

1. “EAN13 Supplementary of 5” symbology support added into JPOS Sample Application.

Version 4.4.0 – 03/2019

1. Added JPOS 1.14 standard compatibility for the JPOS driver.
2. Introduced a JPOS based asset query capability (model #, serial #, date of manufacture, firmware version).

Version 4.3.3 – 01/2019

1. Enhanced Linux log file format to mirror Zebra’s Windows SDK logging format
2. Added ability to customize logging capabilities via the CoreScanner configuration file. Settings include
 - a. Log file storage path
 - b. Number of stored log file. For example, save the last 50 files.



Version 4.3.1 – 08/2018

1. Added support for JPOS Scale Asynchronized Read Weight -1 (forever) timeout support.
2. Added support for Intelligent Document Capture (IDC).
3. Added ability to specify an on board udev library using the CoreScanner's configuration file instead of the system default.

Version 4.2.4 – 04/2018

1. JPOS Scale Read Weight Asynchronized mode support.
2. JPOS Statistics reporting support for Scale.

Note C++ applications cannot be compiled with C++ 11 based compilers without setting `_GLIBCXX_USE_CXX11_ABI` macro to 0. Below definition will be used to set ABI to the old C++ version.

```
#define _GLIBCXX_USE_CXX11_ABI 0
```

Version 4.1.11 – 01/2018

1. Host variant switching from CDC to management modes.

Version 4.1.0 – 10/2017

1. Support for custom PIDs.

Version 4.0.0 – 05/2017

1. Source tarball.
2. Install Verifier application added.

Version 3.0.0 – 08/2016

1. IBM TT, IBM HID, SNAPI firmware download support.
2. SNAPI fast firmware downloads support.
3. SNAPI Imaging and Video capturing.
4. API events like PNP events.
5. Detecting HIDKB devices.

Version 2.0.0 – 02/2016

1. Existing drivers were rebranded from Motorola to Zebra.

Version 1.3.0 – 08/2015

1. Supports for Ubuntu 14.04, Ubuntu 12.04, SLES 11, CentOS 6.3 distributions.
2. Concurrent application support for Scanner and Scale of MP6200 in JPOS.
3. Cascaded device support for MP6200, Presentation Cradle and Standard Cradle.

Components



Component or Module	File	Installation Path
CoreScanner libraries and symbolic links	libcs-iudev.so.4.4.0	/usr/lib/zebra-scanner/corescanner
	libcs-comm.so.4.4.0	
	libcs-common.so.4.4.0	
	libcscl-snapi.so.4.4.0	
	libcs-client.so.4.4.0	
	libcs-clientscanner.so.4.4.0	
	libcs-clientscale.so.4.4.0	
	libcscl-ibmtt.so.4.4.0	
	libcscl-ibmhh.so.4.4.0	
	libcscl-hidkb.so.4.4.0	
	libcs-client.so -> libcs-client.so.4.4.0	
	libcs-clientscale.so -> libcs-clientscale.so.4.4.0	
	libcs-clientscanner.so -> libcs-clientscanner.so.4.4.0	
	libcs-comm.so -> libcs-comm.so.4.4.0	
	libcs-common.so -> libcs-common.so.4.4.0	
	libcs-iudev.so -> libcs-iudev.so.4.4.0	
	libcscl-hidkb.so -> libcscl-hidkb.so.4.4.0	
	libcscl-snapi.so -> libcscl-snapi.so.4.4.0	
	libcscl-ibmtt.so -> libcscl-ibmtt.so.4.4.0	
libcscl-ibmhh.so -> libcscl-ibmhh.so.4.4.0		
CoreScanner Daemon	Cscore	/usr/bin/
	cscored-ubuntu	



Init scripts for CoreScanner daemon for many distributions	cscored-suse	/usr/share/zebra-scanner/sbin/
	cscored.service	
	cscored.lsb	
JPOS driver and SDK libraries	xml-apis.jar	/usr/lib/zebra-scanner/javapos/jpos/
	xercesImpl.jar	
	JposServiceScanner.jar	
	JposServiceScale.jar	
	JposServiceOnScanner.jar	
	JposServiceOnScale.jar	
	JposServiceJniScanner.jar	
	JposServiceJniScale.jar	
	JposLogger.jar	
	javaPOS114.jar	
	JLogger.jar	
	logback-core-1.2.3.jar	
	logback-classic-1.2.3.jar	
lsf4j-api-1.7.25.jar		
CoreScanner libraries for JPOS JNI	libcs-jniscanner.so.4.0.0	/usr/lib/zebra-scanner/javapos/jni/
	libcs-jniscale.so.4.0.0	
	libcs-jniscanner.so -> libcs-jniscanner.so.4.0.0	
	libcs-jniscale.so -> libcs-jniscale.so.4.0.0	
CoreScanner console sample application source	src/ConsoleSampleEventListener.cpp	/usr/share/zebra-scanner/samples/console-app/
	src/ConsoleMain.cpp	
	include/ConsoleSampleEventListener.h	



codes and executable files	include/ConsoleMain.h	
	bin/corescanner-console-app	
CoreScanner GUI sample application (Source code and executable files)	include/xml_formatter.h	/usr/share/zebra-scanner/samples/gui-app/
	include/util.h	
	include/scanner.h	
	include/debug.h	
	include/pugiconfig.hpp	
	include/pugixml.hpp	
	src/xml_formatter.cpp	
	src/util.cpp	
	src/pugixml.cpp	
	src/main.cpp	
	src/debug.cpp	
	gui.xml	
	Makefile	
	images/empty.png	
bin/corescanner-gui-app		
JPOS sample application	JposTest.jar	/usr/share/zebra-scanner/samples/jpos-sample-app/
	jpos-sample-app.sh	
	build.xml	
JPOS direct I/O sample application	JposTestDio.jar	/usr/share/zebra-scanner/samples/jpos-directio-app/
	jpos-directio-app.sh	
JPOS based Remote Management Solution	jposRemoteManagement.jar	/usr/lib/zebra-scanner/javapos/jpos
	jposRemoteManagementTool.jar	

	remote-management.sh	/usr/share/zebra-scanner/remote-management
	jpos-remote-management.xml	/usr/share/zebra-scanner/javapos/xml/
JPOS configuration and property files	xml/jpos.xml	/usr/share/zebra-scanner/javapos/
	config/rsm-id.xml	
	config/rsm-id.dtd	
	config/jpos_trace.properties	
Source code for JPOS sample application	All .java files located inside the directory	/usr/share/zebra-scanner/samples/jpos-sample-app/src/com/jpos/POStest
Source code and resources for JPOS direct I/O sample	All .java files located inside the directory	/usr/share/zebra-scanner/samples/jpos-directio-app/src/com/jpos

Installation Options (Supported Distributions)

Supported distributions and their installation options.

Supported Architecture for CoreScanner and JPOS Drivers for Linux

Supported Distribution

	RPM_x86_64bit	RPM_x86_32bit	Debian_x86_64bit	Debian_x86_32bit	Debian_ARM_32bit	Debian_ARM_64bit
CentOS 8.1	X	NA	NA	NA	NA	NA
CentOS 7.9	X	NA	NA	NA	NA	NA
CentOS 7.3	X	NA	NA	NA	NA	NA
CentOS 7	X	X	NA	NA	NA	NA
CentOS 6.3	X	X	NA	NA	NA	NA
SLES 15 SP3	X	NA	NA	NA	NA	NA
SLES 12 SP5	X	NA	NA	NA	NA	NA
SLES 11 SP3	X	X	NA	NA	NA	NA
SLES 11 SP2	X	X	NA	NA	NA	NA
Open SUSE 15.2	X	NA	NA	NA	NA	NA
Open SUSE 13.1	X	NA	NA	NA	NA	NA
Red Hat 7.4	X	NA	NA	NA	NA	NA
Ubuntu 22.04	NA	NA	X	NA	NA	NA
Ubuntu 20.04	NA	NA	X	NA	NA	NA
Ubuntu 18.04	NA	NA	X	NA	NA	X
Ubuntu 16.04	NA	NA	X	X	NA	NA
Ubuntu 14.04	NA	NA	X	X	NA	NA
Ubuntu 12.04	NA	NA	X	X	NA	NA
Debian 10.10	NA	NA	X	X	NA	NA
Debian 9.5	NA	NA	X	X	NA	NA
Debian 8	NA	NA	X	X	NA	NA
Debian 7	NA	NA	X	X	NA	NA
Raspberry Pi OS	NA	NA	NA	NA	X	X
Other *	NA	NA	NA	NA	NA	NA
	x86_64	i386	x86_64	i386	armhf arm7l	arm64

* Customer Testing Required

Supported Architecture for IoT Connector for Linux

Supported Distribution	RPM_x86_64bit	RPM_x86_32bit	Debian_x86_64bit	Debian_x86_32bit	Debian_ARM_32bit	Debian_ARM_64bit	
	CentOS 8.1	X	NA	NA	NA	NA	NA
	CentOS 7.9	X	NA	NA	NA	NA	NA
	SLES 15	X	NA	NA	NA	NA	NA
	Ubuntu 20.04	NA	NA	X	NA	NA	NA
	Ubuntu 18.04	NA	NA	X	NA	NA	NA
	Ubuntu 16.04	NA	NA	X	X	NA	NA
	Ubuntu 14.04	NA	NA	X	NA	NA	NA
	Debian 9.5	NA	NA	X	X	NA	NA
	RedHat 7.4	X	NA	NA	NA	NA	NA
	x86_64	i386	x86_64	i386	armhf arm7l	arm64	

Supported Architecture for AFM for Linux

Supported Distribution	RPM_x86_64bit	RPM_x86_32bit	Debian_x86_64bit	Debian_x86_32bit	Debian_ARM_32bit	Debian_ARM_64bit	
	CentOS 8.1	X	NA	NA	NA	NA	NA
	CentOS 7.9	X	NA	NA	NA	NA	NA
	Ubuntu 22.04	NA	NA	X	NA	NA	NA
	Ubuntu 20.04	NA	NA	X	NA	NA	NA
	Ubuntu 18.04	NA	NA	X	NA	NA	NA
	Ubuntu 16.04	NA	NA	X	X	NA	NA
	Ubuntu 14.04	NA	NA	X	NA	NA	NA
	Debian 10.10	NA	NA	X	X	NA	NA
	Debian 9.5	NA	NA	X	X	NA	NA
	x86_64	i386	x86_64	i386	armhfarm	arm64	

Supported Architecture for CLU for Linux

		RPM_x86_64bit	RPM_x86_32bit	Debian_x86_64bit	Debian_x86_32bit	Debian_ARM_32bit	Debian_ARM_64bit
		x86_64	i386	x86_64	i386	armhfarm	arm64
Supported Distribution	CentOS 8.1	X	NA	NA	NA	NA	NA
	CentOS 7.9	X	NA	NA	NA	NA	NA
	Ubuntu 22.04	NA	NA	X	NA	NA	NA
	Ubuntu 20.04	NA	NA	X	NA	NA	NA
	Ubuntu 18.04	NA	NA	X	NA	NA	NA
	Ubuntu 16.04	NA	NA	X	NA	NA	NA
	Ubuntu 14.04	NA	NA	X	NA	NA	NA

Dependencies and Package Installation

Dependencies for CoreScanner Driver for Linux and Scanner SDK for Linux are as follows,

Library	Version
Libudev	147 or above
GTK (for GUI applications)	2.24.10
Java	6 (JDK 1.6)

NOTE: Linux C11 vs C98: GNU Compiler Collection (GCC) is an optimizing compiler produced by the GNU Project supporting various programming languages, hardware architectures and operating systems. GCC/g++ version 5.3.1 or later has C11 support. GCC prior to 5.3.1 use C98 versions. Command to show the GCC version are **gcc --version** or **g++ --version**.