

Oracle® Java ME Embedded

Reference Platform Release Notes (Raspberry Pi)

Release 8

E48518-03

April 2014

Table of Contents

- [Introduction](#)
- [What's Supported in This Release](#)
- [Usage Notes](#)
- [Installation and Runtime Security Guidelines](#)
- [Known Bugs](#)
- [Product Documentation](#)
- [Documentation Accessibility](#)

Introduction

The Oracle Java ME Embedded release 8 software for the Raspberry Pi platform is a ready-to-run binary for use with an off-the-shelf Raspberry Pi board. The Oracle Java ME Embedded software uses an optimized platform stack for small embedded devices, which includes the Connected Limited Device Configuration (CLDC) HotSpot Implementation (Java Virtual Machine) version 8, the Micro Edition Embedded Profile (MEEP) application environment, the Generic Connection Framework (GCF) API, and enhanced support for various Java Specification Requests (JSRs).

What's Supported in This Release

The following features are included in the Oracle Java ME Embedded software:

- CLDC 8
 - Full API set
- General Connection Framework (GCF) 8
 - File protocol (file: scheme)
 - Datagram (datagram: scheme)
 - TCP/IP client socket (socket: scheme)
 - TCP/IP server socket (socket: scheme)
 - Secure client socket (ssl: scheme)

- HTTP (http: scheme)
- HTTPS (https: scheme)
- Secure datagram (dtls: scheme)
- Secure server socket (ssl: scheme)
- Access points
- The NetworkUtilities class
- javax.microedition.pki package and other security-related enhancements
- Java ME Embedded Profile (MEEP) 8:
 - javax.microedition.event
 - javax.microedition.power
 - javax.microedition.io (IMC, PushRegistry)
 - javax.microedition.midlet
 - javax.microedition.rms
 - javax.microedition.swm
- Device I/O APIs, which provide enhanced device controls and improved input/output (I/O) for small embedded devices:
 - GPIO
 - I2C
 - Serial Peripheral Interface (SPI)
 - Universal Asynchronous Receiver/Transmitter (UART)
 - Watchdog Timer
 - Modem Control (AT) command set
 - ATDevice, which is a simple AT commands-based device on the EmbeddedDevice1 emulator that responds with OK to any command. It has the following configuration:
 - * Device Name: EMUL, ID: 13, Device Number: 1, Hardware Channel's Number: 1
- Ongoing support for the following optional packages:
 - JSR 75 - (FileConnection API only)
 - JSR 172 - Web Services
 - JSR 177 - Security and Trust Services API (SATSA-CRYPTO package only)
 - JSR 179 - Location
 - JSR 280 - XML API for Java ME
- Tooling including:
 - Command-line interface (CLI)
 - Logging
 - File system commands
 - Debugging

- Memory monitoring is fully supported in this release and provides:
 - Contents of the Java heap
 - A call context for each object at its creation
- CPU profiling is fully enabled in this release and provides:
 - A way to identify bottlenecks in applications
 - The following data for each method: an execution duration, exact number of calls, and a method context
- Network monitoring support

Usage Notes

The Oracle Java ME Embedded software for the reference board platform includes an CLDC implementation with a high-performance Java Virtual Machine that can run IMlets and access input/output ports. This runtime is optimized for the reference board platform.

Getting Started Guide for the Reference Platform (Raspberry Pi) describes how to install the Raspbian distribution on the SD card, how to connect to the board from the development host computer, and how to install, run, and debug IMlets on the board.

Note the following important information before running the Oracle Java ME Embedded software on the board:

- This release does not support running multiple instances of its executable; avoid simultaneously starting several instances of any of scripts or executable files, regardless of whether these are from the same installation of the software or from different installations. The software can run multiple IMlets in the same instance of a virtual machine; you do not need to start multiple VMs to run several applications at a time. Not following this precaution can result in malfunctions with uninformative error messages and might cause corruption of the installation files.
- All devices, except MMIO, support only the exclusive mode. MMIO also supports the shared access mode.
- No generic device is implemented, as would be accessed using the package `jdk.dio.generic`.
- The `jdk.dio.power` package does not support the actual hardware power state switch.
- Access points are one-to-one mapped to network interfaces. However, the access point management functionality is implemented with a limitation which does not allow to change the state of the Linux network interface from Java. The state can only be read. The proxy setting is also available through an environment variable.
- System events and power management are not implemented due to lack of support on the Raspbian side.

Installation and Runtime Security Guidelines

The Oracle Java ME Embedded release 8 software installation requires an execution model that ensures certain networked resources available. These required resources

might include, but are not limited to, a variety of communication capabilities between the product's installed components.

It is important to note that the product's installation and runtime system is fundamentally a developer system that is not specifically designed to guard against malicious attacks from outside intruders. Given this, the product's architecture can present an insecure operating environment to the installation file system and its runtime environment, during execution. For this reason, it is critically important to observe the precautions outlined in the following security guidelines when installing and running the software.

Note: The security-related functionality of a final developed application for release into the field is supported by the available components of the Oracle Java ME Embedded software stack incorporated by the developer into the application. The security precautions required by applications in the field are beyond the scope of these recommendations, but must be observed by the application developer.

To maintain optimum network security, the software package can be installed and run in a *closed* network operating environment; the software system is not connected directly to the Internet or to a company intranet environment that could introduce unwanted exposure to malicious intrusion. This is the ideal secure operating environment whenever the application under development does not require an Internet connection.

An example of a requirement for an Internet connection is when the system must communicate with a wireless network over the Internet to fully execute the application under development. Whether or not an Internet connection is required depends on the particular Java ME application running in the development environment. For example, some Java ME applications can use an HTTP connection. If the environment is open to network access, you must observe the following precautions to protect valuable resources from malicious intrusion:

- Locate the development environment behind a secure firewall that strictly limits unauthorized network access to its file system and services. Limit access privileges to those that are required for development while allowing all the bidirectional local network communications that are necessary for the application's functionality. The firewall configuration must support these requirements to run the software while also addressing them from a security standpoint.
- Follow the principle of least privilege by assigning the minimum set of system access permissions required for installation and execution of the software.
- Do not store any sensitive information on the same file system that hosts the installation.
- To maintain the maximum level of security, ensure that the operating system patches are up-to-date on host machines in the development environment.

Security Certificate Precautions

The Oracle Java ME Embedded software distribution bundle contains security certificates that are needed for testing during development of products for final release to customers. Some of these certificates are self-signed security certificates generated by Oracle that are mapped to privileged security domains. IMlets or MIDlets signed

by these certificates get high privileges to access restricted APIs; these certificates present a security vulnerability if they are released to end users on a customer's device. You must remove these certificates after final testing of the product is completed, when the product is being prepared for release to end users. You do not need to remove certificates issued by universally recognized certificate authorities (CAs), because these certificates are used only for signature verification, and they do not present a vulnerability.

Developer Agent Precautions

The CLI is incorporated in the Developer Agent, which communicates with a device through an unsecured protocol. The Developer Agent is a Java SE application that can be reverse engineered to tamper with or to get information about the communication protocol, which might be used by an untrusted entity to manipulate the device. If you decide to implement the Developer Agent in a product deployment, it is your responsibility to incorporate adequate security measures around the Developer Agent communication channel. This channel uses TCP port 2201 on the Raspberry Pi device for the communication.

Known Bugs

For generic bugs in this release of the Oracle Java ME SDK that might affect the Raspberry Pi platform, see *Oracle Java ME Software Development Kit Release Notes*.

The following are known bugs in this release of the Oracle Java ME Embedded software:

- The Raspberry Pi board cannot be started when its SD card is full
It is recommended that you clean the SD card manually and restart the board.
- The Raspberry Pi board is not recognized by the Device Connection Manager
On Windows 7, the Raspberry Pi IP address might be missing in the IP Address drop-down list after performing the following actions:
 - Close the Device Connections Manager
 - Delete the %HOME%\javame-sdk directory.
 - Run the emulator by using the following command:

```
bin>emulator
```
 - Run Java on the Raspberry Pi by using the following command:

```
$sudo ./usertest.sh
```
 - Open the Device Connection Manager window and click the **Add** button.
The IP Address drop-down list does not contain the Raspberry Pi IP address.
- Installation of a MIDlet whose dependency chain contains different versions of the same library (LIBlet) results in an installation error.
To enable the correct installation of the MIDlet, ensure that only one version of a LIBlet is specified in the MIDlet's dependency chain provided by the MIDlet-Dependency-JAD-URL- (n) and LIBlet-Dependency-JAD-URL- (n) attributes.
- The Probe test results in I2C, SPI, UART unavailability

If you start the Probe test and then try to open and use any of I2C, SPI, and UART peripherals, `UnavailableDeviceException` is thrown. Restart the Raspberry Pi board to make the listed peripherals available again.

- The `UART.read(ByteBuffer)` method does not read the requested number of bytes. The `ReceiveTimeout` and `TriggerLevel` properties cannot be set.
- The `SPIDevice.begin()` method does not throw an `IllegalStateException` exception in the case when the previous transaction has already started. Instead, an `IOException` exception is thrown.
- The `SPIDevice.end()` method does not throw an `IllegalStateException` exception in the case when the previous transaction has already ended. Instead, an `IOException` exception is thrown.
- An attempt to open a UART device by using its configuration when the device is opened in the exclusive mode, does not throw an `UnavailableDeviceException` exception.
- The `get` command does not provide a detailed description of the default `<hostFileName>` location.
- The `pwd` command when issued after the `cd` command, accumulates the previous `cd` calls and writes incorrect path.
- The `pwd` command writes an extra empty line after the `<<pwd, OK` line.
- The `pwd` command writes the usage information, which should not be in the response.
- The CLI does not provide a command to delete a directory.
- When connecting to the CLI, the device is not shown as connected even if the connection is available.
- Comm ports on the Raspberry Pi board have invalid names.
The current implementation uses the name of the device in the Linux operating system `/dev/ttyAMA0`. This name does not comply with the specification and cannot be used.
- The Java runtime throws a `java.lang.ClassCastException` exception when dealing with a certificate with the SAN field.
- The `UART.read()` and `UART.read(ByteBuffer)` methods might be blocked before closing of the corresponding device, terminating the corresponding thread, or restarting an application when amount of the expected data exceeds the size of the buffer.

As a workaround, specify a buffer of a bigger size than the amount of the expected data.

Product Documentation

The following documentation is included with this release of the Oracle Java ME Embedded software. See <http://docs.oracle.com/javame/8.0/index.html>.

Application	Title	Format
All (this document)	<i>Release Notes</i>	HTML PDF

Application	Title	Format
Demonstrates how to install, run, and troubleshoot the Oracle Java ME Embedded software on the Raspberry Pi platform.	<i>Getting Started Guide for the Reference Platform (Raspberry Pi)</i>	PDF HTML

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Oracle Java ME Embedded, Reference Platform Release Notes (Raspberry Pi), Release 8
E48518-03

Copyright © 2012, 2014, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

