

Wind River Linux Secure

Increasingly, companies charged with creating complex, connected, secure, and mission-critical solutions are looking to open source and open standards software to leverage rapid growth in the ecosystem and provide much needed functionality, flexibility, innovation, performance, and total-cost-of-ownership advantages. To address this need Wind River has created a commercial-off-the-shelf (COTS)

embedded Linux platform based on open standards.

Wind River Linux Secure is a Common Criteria security-certified, commercial-grade embedded Linux development and run-time platform for use where assured security is a project requirement. Wind River Linux Secure is built on Wind River's industry-leading embedded Linux platform and provides a flexible and

pervasive development environment that enables companies to develop, test, and support both COTS and highly customized devices quickly and cost effectively.

Wind River Linux Secure is certified on selected platforms of Intel Architecture, PowerPC, and ARM. It can be deployed securely on COTS and custom hardware from multiple vendors including Freescale, Intel, and Texas Instruments, reducing the overall cost of development and certification. Our flexible design makes it possible to enable new boards and facilitate incremental certifications to keep the total cost of ownership low.

Wind River Linux Secure Highlights

Wind River Linux Secure is based on the stable Linux 2.6.27 kernel and GCC 4.3.2 compiler and is certified to Common Criteria Evaluation Assurance Level 4+ (EAL 4+) and Federal Information Processing Standard (FIPS) 140-2. This certified platform also includes support for multiple hardware architectures, including ARM, Freescale, and Intel, enabling Linux system designers to deploy on the platform that best balances power and performance. It is the first and only commercial embedded Linux platform to achieve Common Criteria EAL4+ certification using the new and more rigorous General Purpose Operating System Protection Profile (GP-OSPP). The FIPS 140-2 certified cryptography module includes a comprehensive set of cryptographic algorithms to assure information security and integrity in connected settings.

Figure 1 is an overview of the components of Wind River Linux Secure. The product consists of source code and a build system that generates an optimized run-time image suitable for embedded devices. The components of the product are combined to create a defined run-time image.

Table of Contents

Wind River Linux Secure Highlights.....	1	Wind River Workbench.....	6
Security Features	2	Analysis Tools	6
Wind River Linux Distribution		Package Management.....	7
Assembly Tool.....	4	Run-Time Features.....	7
Project Creation.....	4	Kernel Features.....	7
Development	4	Networking Features.....	9
Package Building	5	Board Support Packages.....	10
File System Creation.....	5	Optional Add-on Products.....	11
Exporting a Layer	5	IPL Cantata++ for Wind River	
Pseudo	5	Workbench.....	11
QEMU.....	5	Wind River Workbench	
Layers.....	5	On-Chip Debugging.....	11
Server Install.....	5	Wind River Test Management	11
Toolchain.....	5	Wind River Network	
Toolchain Vendor	6	Management	11
C Library	6	Wind River SNMP	12
Validation	6	Testing and Validation	12
Custom Features	6	Partner Ecosystem.....	13
Multilib Support.....	6	Professional Services.....	13
Prebuilt Multilibs	6	Education Services	13
Toolchain Wrappers.....	6	Personalized Learning Program	13
Toolchain Building	6	Support Services	13
Toolchain Export	6	Technical Support.....	13
License Management	6	Appendix A: Package Summary	
Wind River Linux Secure		by Category	14
Development Tools	6	Appendix B: Supported	
		Target Boards	15
		Appendix C: Supported	
		Development Hosts.....	15

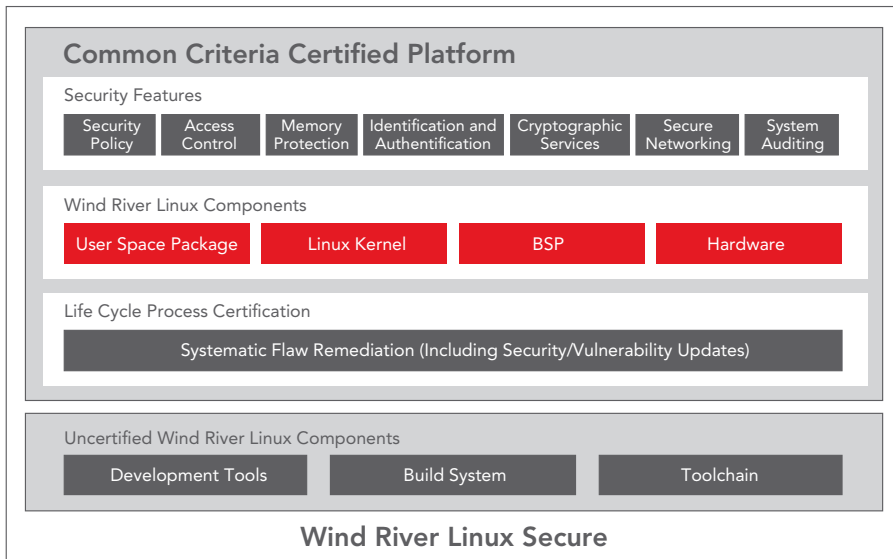


Figure 1: Major features of Wind River Linux Secure certified configuration

Wind River Linux Secure contains the following components:

- **Security features:** Common Criteria–evaluated security components, exceeding requirements for GP-OSPP
- **Systematic flaw remediation:** Certified security response methodology as additional requirement
- **User space packages:** Hundreds of software packages that operate in protected Linux user mode
- **Kernel source:** The 2.6.27 Linux kernel with many fixes and feature enhancements
- **Board support packages (BSPs):** Hardware enablement components
- **Tools:** Software development tools, including the award-winning Eclipse-based Wind River Workbench development suite
- **Build system:** The Wind River Linux Distribution Assembly Tool, which is used by the developer to compile and assemble these components
- **Toolchain:** The cross compiler based on the GNU Compiler Collection (GCC)

Wind River Linux Secure comes with a preconfigured system profile for Common Criteria certified configuration to fast track your development and security evaluation process. By identifying, assembling, and integrating commonly used packages, Wind River Linux Secure saves you weeks of specialized labor, allowing resources to be focused on creating highly optimized devices. It conforms to industry specifications on reliability, assurance, and high availability, with a base foundation of Carrier Grade Linux (CGL) and SELinux.

Because Wind River Linux Secure is based on Wind River's standard product line, it shares the following platform advantages:

- Commercial-product quality with extensive testing and quality assurance (QA), reliable service packs, and security patches with standard software product life cycle support
- Extensive hardware and software ecosystem support
- Lower total cost of ownership by eliminating the burden of building, supporting, and maintaining a Linux distribution, enabling research and development (R&D) teams to focus on differentiating applications
- Reduced complexity of present and future projects by leveraging the Wind River Linux cross-build system and layers development methodology
- Rich tools and development environment support based on the open Eclipse framework
- Predictable roadmap for long-term product development and support
- Detailed documentation of software package license information to enable compliance with legal and regulatory requirements

Security Features

Wind River Linux Secure contains the following security features:

- **Security policy:** The core of Wind River Linux Secure is an independently verified, multifaceted security policy:
 - **SELinux reference monitor:** Originally developed by the National Security Agency (NSA), based on the Flux Advanced Security Kernel

(FLASK) model, SELinux is a Linux Security Module (LSM) that provides a non-optional, verified, tamperproof security policy enforcement engine for all subjects and objects within a Linux system.

- **Strict reference policy:** Independently developed by the SELinux project, the reference policy is a basis for security policy definition and enforcement. With policy tailored specifically for the Wind River Linux Secure evaluated configuration and ready-to-use policy modules for a wide range of other open source packages, the strict reference policy enables rapid deployment of complete Linux-based platforms. The Wind River Linux Secure reference policy includes the following, all of which apply to both local and network-based objects and subjects, thus providing full multilevel secure (MLS) and multi-category secure features for networking:
 - **Domain and type enforcement:** Central to any role-based access control system, domain and type enforcement describe all actions and activities that should be allowed by the policy engine and provide fine-grained control beyond that provided by discretionary access control (DAC).
 - **Multilevel security:** This common, hierarchical security policy is implemented via extended file system attributes and security labels.
 - **Multi-category security:** This extends the security labels to include distinct, incomparable security information for a complete security solution.
- **Access control:** Both DAC and mandatory access control (MAC) security mechanisms are supported, which enables designers to optimize how the user and application access system objects (files, directories, and sockets). MAC enforces multilevel security based on the Bell-LaPadula model.
- **System hardening:** Expanding on the system security provided by SELinux and the associated policy, Wind River Linux Secure includes a proscriptive security model where common system stability issues are identified and protected against in a generalized way.
- **PaX:** PaX is technology implementing least-privilege access for memory pages that ensures data and executable code are stored in completely separate

pieces of memory using address space layout randomization (ALSR), effectively eliminating arbitrary code execution attacks. It ensures that freed and uninitialized memory is scrubbed. Program load addresses cannot be predicted and intercepted. Address space permission enforcement is also provided.

- **Compile-time stack protection:** All applications built with the Wind River Linux Secure toolchain include the GCC stack protection compile flags that ensure buffer overflows do not occur.
- **GRSecurity TPE:** Among the many features of GRSecurity, Trusted Path Execution (TPE) is a powerful tool for ensuring the stability and integrity of the system by preventing any uploaded application from executing unless it is owned by a trusted system user and is executing from a trusted system path.
- **GRSecurity RBAC:** GRSecurity includes a fast, integrated role-based access control mechanism featuring zero memory allocation even for modifying and loading new roll-based access control policies.
- **chroot jail hardening:** Wind River Linux Secure includes especially hardened chroot jails that are invulnerable to typical attacks that break out of standard chroot jails.
- **Identification and authentication:** Wind River Linux Secure includes many features specifically intended to address common user-based security failures.
- **User authentication:** Based on the Linux Pluggable Authentication Modules, the system administrator can define strict but flexible password or passphrase quality control. The standard configuration requires a minimum 16-character password or a four-word passphrase and, unlike standard Linux configurations, it is not possible for a user to opt to use a lower-quality password.
- **vlock:** Users who leave a login session unattended or idle will have their sessions locked with the console locking program, vlock, rather than being automatically logged out. Failed authentication attempts, whether in attempting to start a new session or to unlock a currently locked session, are logged and if the system policy dictates can result in accounts being locked either for a period of time or permanently.
- **Account expiry:** User accounts can be configured by the system policy to expire after a fixed period or a period of inactivity.

- **User data scrubbing:** When a user account is removed from the system, all data associated with that user is completely scrubbed from the system, including any residual data stored on local disks.
- **User security services:** Wind River Linux Secure provides security to more than just the system. Users on a Wind River Linux Secure system have a range of technologies available to improve their security and privacy:
 - **Secure backup of user data:** Users can safely and reliably back up their own data through two different technologies depending on their needs:
 - **Star:** The standard tape archiver (star) creates local, standards-based TAR archives, preserving all security labels and extended file system attributes. This provides users the confidence that their data is backed up in a way that will ensure it still has the same security attributes even when loaded on another Wind River Linux Secure system.
 - **duplicity:** If it is necessary for users to back up their data across an untrusted network or to an untrusted site, duplicity can provide fast, incremental backups that are either signed or encrypted with GNU Privacy Guard (GnuPG) public key cryptography.
 - **Cryptographic services:** FIPS 140–certified Network Security Services (NSS) provides a set of libraries designed to support cross-platform development of security-enabled client and server applications, thus providing system integrity checking using the FIPS 140-2 validated NSS module. Validated algorithms include Triple DES, AES, DSA, ECDSA, SHS, RSA, DRBG, and HMAC. For added security, key wrapping with a hash or signature when exporting keys is also supported.
 - **Wind River Cryptographic Framework:** A certified API to NSS is offered to users via the Wind River Cryptographic Framework, which provides separation between user space and the crypto layer, protecting the NSS libraries while facilitating user access to initialized cipher mechanisms.
 - **NSS:** The NSS API is still available to applications not making use of the certified cryptographic operations provided by Wind River Cryptographic Framework.

- **Open Cryptographic Framework (OCF):** Asynchronous hardware and software cryptographic acceleration is available through the Open Cryptographic Framework, with applications such as OpenSSH and OpenSSL clients ready to make use of the bulk encryption/decryption functions out of the box.
- **Secure networking:** Wind River Linux Secure offers many different technologies to support secure networking requirements:
 - **OpenSSH:** Remote access via OpenSSH protects network communication against wiretapping or eavesdropping.
 - **MLS and multi-category security labelled networking and IPsec:** These components ensure that security attributes such as sensitivity levels are preserved through the entire network communications channel.
 - **Kerberos:** Remote identification and attestation can be achieved through Kerberos, which is particularly hardened against eavesdropping and replay attacks. Both symmetric and asymmetric authentication modes are supported.
 - **Racoon:** Part of the KAME project, focusing on providing IKE operations, Wind River Linux Secure includes several fixes to better support labelled networking.
 - **OpenSSL and GNU Transport Layer Security (TLS):** Secure communication over the network layer is provided to applications via the Secure Socket Layer APIs presented in both OpenSSL and the GnuTLS libraries as well as common command-line applications such as an X.509 certificate manager and a strong password generator.
 - **Stunnel:** For existing applications that require secure communications but are not already using either OpenSSL or GnuTLS, the stunnel tool is able to provide a trusted, reliable SSL/TLS tunnelling service.
 - **Virtual LANs:** A full implementation of the 802.1Q VLAN tagging specification, as well as MAC-based VLAN functionality, is available and validated in Wind River Linux Secure.
- **KeyNote trust management library:** Wind River Linux Secure includes an integrated, validated trust management system in the KeyNote library, allowing flexible, reliable credential sharing between networked Wind River Linux Secure systems or other systems with compatible trust management systems.

- **System auditing:** Wind River Linux Secure provides reliable and effective system-level auditing of all security-relevant events in a number of ways:
 - **Linux Audit Framework:** Wind River Linux Secure uses the Linux Audit Framework, part of the SELinux feature set, and the SELinux user space audit daemon to ensure all security-relevant events are logged in a reliable, tamperproof fashion including the ability to provide file system integrity checking.
 - **GRSecurity:** With the ability to log dozens of different system events including signals sent to processes, mount operations on file systems, file and device access, resource usage, and attempts (successful or otherwise) to exercise POSIX capabilities, GRSecurity complements the Linux Audit Framework to provide the system administrator with a complete view of system usage patterns.
 - **Logsentry:** The task of processing the standard system logs is vastly simplified by the logsentry tool, which can watch any growing or rotating log file for events or messages of particular interest to the individual system administrator.
- **File access and modification:** Virtually any file in the system can be monitored with the following technologies:
 - **RPM:** Using FIPS 140-2-validated, strong cryptographic hashing, all installed binaries and configuration files can be monitored for changes in size, content, ownership, permissions, or location.
 - **Samhain:** Similar to tools such as the Advanced Intrusion Detection Environment (AIDE) or Tripwire, Samhain can provide a more complete picture of system activities and changes from the last “known good” state as well as watch for user activities such as successful and unsuccessful authentication attempts.
 - **mtree:** Available to both system administrators and regular users for monitoring arbitrary directory hierarchies, mtree comes from the BSD world and provides a fast, reliable tool for watching for any deviations from expected checksums of files.
 - **inotify:** For real-time notification of file or directory access, inotify is an ideal tool for both system administrators and users to know who is attempting to access their data.

- **Systematic flaw remediation:** Wind River Linux Secure is certified for its security response methodology as an additional requirement. Systematic flaw remediation levels 1 through 3 establish increasingly rigorous processes for identifying, documenting, communicating, and correcting system security flaws in a target of evaluation (TOE), up to and including the assignment of dedicated resources to ensure timely responses. This gives users a unique confidence in Wind River’s process of remediating system security flaws and assurance that no flaws will be introduced into a certified system after evaluation is completed.

Wind River Linux Distribution Assembly Tool

Wind River Linux Distribution Assembly Tool (LDAT) is a build system to cross-compile and assemble components for run-time images. LDAT is licensed under the GNU General Public License, version 2.

LDAT commonly addresses the use cases shown in Figure 2.

Project Creation

Users start by creating a project using LDAT. They reference configuration information such as the hardware target, kernel type, profiles, and pointers to other custom software. This creates a project. LDAT uses autoconf to generate a configured build directory. There are a large number of options to select the board, kernel configuration, user space

configuration, and so on. The profile option may be used to automatically select the kernel and user space configuration based on the selected BSP and profile combination. The core layers are selected automatically based on the LDAT configuration directory (toolchain version, kernel version, etc.) unless overridden by user arguments.

Additional layers may be included either at the user’s request or automatically by other layers. The configure script searches for layers specified without an absolute or relative path.

The configured project directory is itself a layer that can provide modified versions of packages or tools, change configuration information, and include other layers. The project directory can resynchronize these external layers. This allows for the user to include software being actively developed externally.

Development

Users can use tools such as Wind River Workbench to add packages, make changes, debug, and compile software. LDAT creates file systems in a multitude of configurable ways. Nevertheless, it may be the case that in the field, the original equipment manufacturer (OEM), one of the subsequent integrators or developers, or even the final users will want to modify the file system composition. To this end, all the user space components, regardless of their origin, open source trees, source

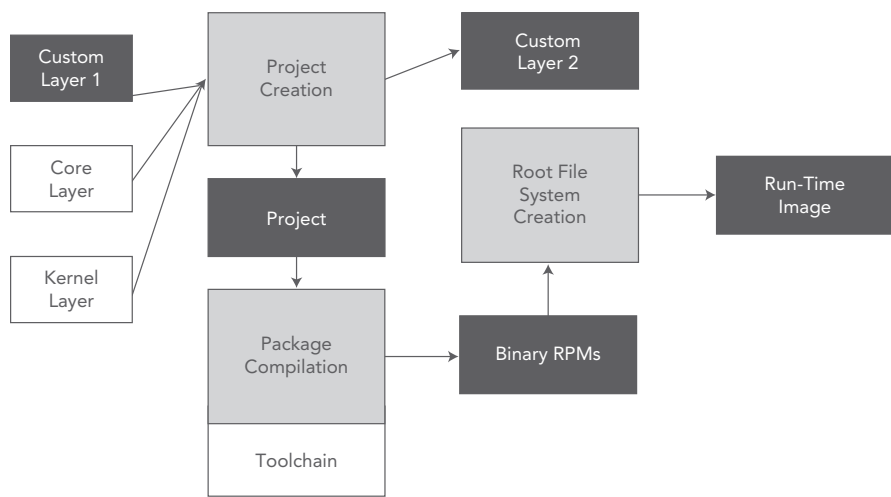


Figure 2: Typical use cases enabled with Wind River Linux Secure

RPMs, or customer trees, are packaged as binary RPMs, and they can be added or deleted or updated at will using the target bound RPM binaries, which also can be included to the file system.

Certification of customer-developed products will be required to maintain the certification status, but this is made easier by starting development from a known certified platform such as Wind River Linux Secure.

Package Building

Users compile packages from source packages to binary RPMs. This uses the Wind River toolchain. The LDAT cross-build environment uses a simple front-end Makefile fragment and, for SRPM packages, a modified spec file to provide the information required to make the package conform to LDAT and make it cross-buildable. If source code changes to the package are also required, patches are either applied once the package is extracted or during the normal prep stage of the RPM build. Patches (and the spec file) can be provided with the package in the layer and via templates included in the configuration if configuration-specific modifications are required. Most open source packages can be imported with minimal changes and are immediately buildable with LDAT.

Both package formats support a pristine source model. This means the original upstream open source code is provided, plus incremental patches that each implement one major feature. This allows for maximum transparency of source code.

LDAT understands dependencies so that packages are rebuilt when their dependents are changed. To determine whether a package is valid for the current configuration, LDAT uses a checksum value calculated for the current configuration and compares it with the version stored in the RPM. The configuration checksum includes the sources and patches for the package or tool, the LDAT Makefile fragment for the package, any additional configuration files, toolchain flags, and so on, along with the configuration checksums of any other package this package depends on. When the checksum doesn't match, the prebuilt version is ignored and the package will be rebuilt (as will any packages that depend on it).

To build packages, the build system calls the cross-toolchains with minimal performance overhead. The ability to use prebuilt versions of most or all host tools and target packages with checksums to ensure they are valid, along with the ability to update the build configuration with layer changes and to update the target file system without reassembling it from scratch, provides quick turnaround for development builds. Prebuilt versions of the GP-OSPP certified packages are all included in Wind River Linux Secure for ease of use and to ensure the customer has the exact software that was evaluated for the certification.

Package building also generates a sysroot to ensure that the right development libraries are being used. This can be exported to other users.

File System Creation

Users create a tuned root file system and run-time image. Once all of the packages are built, the target file system is assembled from the RPMs. If the file system has previously been assembled in this build directory, it can either be rebuilt from scratch or updated in place by uninstalling and reinstalling RPMs that have been changed, added, or removed.

Footprint optimization tools are included to reduce resources.

Exporting a Layer

Users can create a custom layer for reuse for other development groups or other projects. After the system is built, the *make export-layer* command can be used to generate a layer that incorporates any local changes to the configuration, packages, or tools. The *make install-prebuilt* target can be used to place prebuilt copies of the host tools, target packages, and kernel into either a new or existing layer for use by subsequent builds.

Pseudo

To allow package builds and file system creation without root privileges, Wind River Linux Secure supplies a tool that provides limited emulation of root access, preserving user ownership, modes, and device nodes correctly for the eventual target file system.

QEMU

Wind River Linux Secure ships with a hardware simulator built in. It simulates ARM, PowerPC, x86, and MIPS hardware. This is integrated tightly with LDAT so that launching a test version in simulated hardware is trivial.

Layers

LDAT uses a hierarchical "layer" structure where each layer may provide anything from a simple configuration setting or a single package to something more complicated such as completely replacing the kernel or augmenting toolchains, host tools, and so on.

The layers provided with Wind River Linux Secure include the "core" layer (called *wrll-wrlinux*), which supplies the user space packages and core configuration information, the kernel layer for the kernel sources, a host-tools layer for tools that run on the build host, and a toolchain layer that contains the toolchains for cross-development. Add-ons for ecosystem products are typically provided as layers that may augment or modify the default configuration. User-defined layers may provide local changes and prebuilt binaries to speed development.

Contact Wind River for more information.

Server Install

A possible output of LDAT is a bootable DVD image to boot an x86 machine that will then launch an interactive installer. The user can then use this tool to install the generated run-time image on that machine's hard drive.

Toolchain

The Wind River Linux toolchain is a core component of the Wind River Linux build system, providing a number of features specifically suited to embedded development. This section outlines some of the key toolchain features that make embedded development easier and more productive. The toolchain is based on open source components, such as the GNU Compiler Collection (GCC version 4.3.2), GNU binutils version 2.18.50, and the GNU debugger (gdb version 6.6). Note that toolchain version numbers can be misleading because Wind River makes so many changes to the upstream releases. All these changes are transparent to the developer.

The Wind River Linux toolchain supports ARM, x86, MIPS, PowerPC, and SPARC architectures.

Toolchain Vendor

The Wind River Linux toolchain is a customized distribution of Code Saurcery's Saurcery G++ distribution. Code Saurcery maintains the toolchain, providing support from some of the major contributors to the core toolchain components. Wind River's toolchain has full support for a broad range of target architectures and several host systems. Code Saurcery's toolchain expertise provides confidence that bugs will be fixed quickly and correctly.

C Library

To enhance support for embedded systems in the C library, Wind River has collaborated with other key vendors to found eGlibc, an embedded-oriented distribution of the GNU C library, and continues to sponsor the eGlibc project. The eGlibc project provides superior support for non-x86 architectures such as ARM, PowerPC, and MIPS and has a number of features to make it easier to build with a smaller footprint for target systems. Wind River also has uClibc support, suitable for small footprint targets.

Validation

Code Saurcery runs a large set of toolchain tests, including publicly available tests and proprietary tests. All Wind River Linux Secure toolchain changes are validated, no matter how minor the changes appear to be. Wind River performs additional testing and validation upon receiving a new toolchain. Validation is performed for all supported multilibs for each architecture. This testing procedure ensures that regressions are caught before the compiler ships to a customer.

Custom Features

The Wind River Linux toolchain has a number of features not found in corresponding versions of the upstream compilers and sometimes not available in upstream at all. Vendor-specific updates are included in the toolchain as well as a few features that were added to address specific embedded development concerns. Some such features include architecture- and chip-specific optimizations, support for new opcodes,

and improved kernel debugging support in gdb. Finally, all of these features are tested and validated, ensuring reliable behavior even when using custom features.

Multilib Support

The Wind River Linux build system supports simultaneous use of two different CPU types (within the same general family) on a single target; for instance, a target could be configured with both 32-bit and 64-bit binaries, allowing per-package choices of space/speed trade-offs.

Prebuilt Multilibs

The Wind River Linux toolchain provides prebuilt library components, including the C library, for a large variety of multilibs. Each prebuilt library is fully validated as part of the validation process. Prebuilt libraries reduce compilation time while providing extra security and certainty. For each multilib, the toolchain contains configuration files to set up the compiler and other tools to produce compatible code for that multilib. Additional CPU-specific optimizations are available for a broad range of CPUs and configurations.

Toolchain Wrappers

Wind River Linux Secure uses toolchain wrapper scripts that simplify cross development. The project configuration process sets up toolchain wrappers for each CPU type used in the project. These wrappers combine target sysroot configuration and toolchain compilation options to provide seamless building for a target CPU. Users can then use the wrapper program as a substitute for GCC rather than trying to embed needed compiler options in package build processes. The preconfigured CPU templates provide the right combination of options to get good results out of each CPU.

Toolchain Building

The Wind River Linux toolchain is distributed and fully supported as a collection of prebuilt binaries. However, source for every toolchain component is provided, along with configuration data and build scripts to rebuild the toolchain completely from source. Each toolchain component is distributed as a combination

of a specific upstream source release or snapshot plus any additional patches provided by Code Saurcery or Wind River. The toolchain build process automates the task of bootstrapping a new cross-compile toolchain, reducing hundreds of commands and configuration steps to a simple "make toolchain."

Toolchain Export

When a toolchain has been configured for a project on a Linux host, it is possible to export that toolchain as an archive that can be used as a toolchain on another host. This can help with distributed development environments.

License Management

The Wind River Linux Secure toolchain provides a license management feature. Customers who want to keep an eye on their compiler usage or who have site license agreements can use the provided license management software, which is fully integrated into the toolchain. If the license management software is not found, the toolchain produces diagnostic messages but runs with full functionality and full performance.

Wind River Linux Secure Development Tools

Wind River Workbench

The Eclipse-based Wind River Workbench development suite offers deep capability throughout the development process in a single integrated environment, with complete platform integration and tools for debugging, code analysis, advanced visualization, root-cause analysis, and test. These tools are valuable for development but should not be included in the target-certified configurations because they are not secure.

Analysis Tools

Workbench and Wind River Linux Secure offer a number of analysis tools available to the developer. Some are enhanced versions of open source tools related to profiling and memory usage, and some are specifically developed by Wind River:

- **Performance analysis:** The Wind River Workbench Performance Profiler analyzes how a CPU is spending its cycles by providing a detailed function-by-function analysis that shows how

individual routines within the processes consume those cycles. This feature is based on the open source tool `oprofile`, with additional visualization and integration in Workbench.

- **Memory analysis:** Wind River Workbench Memory Analyzer is a dynamic memory analysis tool that helps prevent and fix such problems as memory leaks, excessive number of memory allocations, and excessive memory allocation sizes. The memory analyzer uses the open source tool `mpatrol`, with additional visualization in Workbench.
- **Boot-time analysis:** This uses the `ftrace` tool to provide lightweight function tracing and includes dynamic `ftrace` and early-`ftrace` for boot-time analysis.
- **Code execution coverage:** The code coverage analyzer feature of Wind River Workbench determines the percentage of source code executed by your software test case and points to the sections of code that have not been fully tested.
- **System viewer:** Wind River System Viewer supports visualization of multi-core systems; per-core filter and search facilities; the recording of a number of custom events, which use a printf-like format string; graphical and tabular representations of various types of log file analyses, such as CPU usage (aggregate and per core), system load, and per-core ready and running states. System Viewer also supports a host-driven upload method for log files, resulting in log transfer without interference from task CPU use. It also allows for transfer of multiple logs, plus transfer without requiring you to call target functions.

Package Management

Wind River provides several tools to examine the file system's package list, examine package-level dependencies, perform safe package addition or removal based on those dependencies, and perform file-level examination and control of the file system contents:

- **Package lists and snapshots:** The user space file system is built up from discrete packages, from open source, user source, and virtual packages from custom content. Workbench provides a way to control that package list, to explore different package combinations, and to preserve safe combinations as the user explores them.

- **Dependency tracking:** Workbench allows the user to visualize the (deep) forward and reverse dependencies and to add or remove packages, knowing that the dependencies are reported and managed.
- **Direct package updates to target:** Workbench facilitates RPM management on the target as on a regular Linux host. Packages can be developed and compiled and then pushed to the running target for fast turnaround debugging using incremental updates.

Wind River provides several tools to examine and directly control the file system content below the package level. There are also tools to import new open source packages, import new patches, and directly examine a package's patch tree:

- **File system layout:** The user can see the final file system content and directly remove or add files at a fine-grain level below the coarser package-level dependencies, allowing direct control of the file system footprint. Wind River also provides tools to discover and visualize which files are touched during a target run.
- **Package import tools:** Workbench has a feature to handle most of the initial package importation and cross-compilation setup, to help speed up the adoption of new open source packages into a user's project.
- **Patch import and export tools:** Workbench enables the user to view the patch tree directly and patch files for both source RPM and regular packages. It also provides tools to help import and resolve new patches and to export user changes as new portable patches into a layer directory.
- **Export layer:** This tool can automatically export many changes made in a project into a new portable Wind River Linux Secure layer. This includes package list changes, file system trimming, kernel configure changes, and new local packages.

Run-Time Features

The following are features of the run-time image that runs on the target.

Kernel Features

The 2.6.27 Linux kernel forms the basis of the Wind River Linux Secure kernel. Wind River adds many features and bug fixes to this kernel, and this specific kernel source configuration is tested and supported.

Kernel Changes

The Wind River kernel adds to the kernel.org 2.6.27 base by importing and validating changes from the following categories:

- **mainline:** The feature set of the Linux kernel from kernel.org; extended or validated features in particular configurations and applications, by Wind River
- **external:** Features imported from another external source and merged into the Wind River kernel
- **internal:** Features that are in layers or merged into the kernel, developed by Wind River
- **fixes:** Bug fixes for drivers; features in mainline or external projects

These features are tested individually, merged, and then tested as a complete system. This includes stress and use case testing and ensures that the features are stable individually, integrate with Wind River tools, and form a solid base for deployment tuning.

Kernel Presentation

The Wind River kernel is presented to the developer via a fully patched, history-clean Git repository (see <http://git-scm.com/> for more information on Git). This stores the selected features, board support, and configurations extensively tested by Wind River. Presenting the Wind River kernel in this manner allows the end user to leverage community best practices to seamlessly manage the development, build, and debug cycles.

The build system generates a flat tree from this Git tree that contains the specific features required for the target kernel's use. Storing the source code in Git enables users to more easily understand what changes have been made to the kernel and why. Wind River uses a combination of tags and branches to assist in delineating between the various added features.

The workflow of the Wind River kernel follows the recognized community best practices. In particular, the kernel as shipped with the product should be considered an "upstream source" and viewed as a series of historical/ documented modifications (commits) to the kernel. These modifications to the kernel represent the development and

stabilization done by the Wind River kernel development teams.

Contact Wind River for more information about kernel development workflow with Wind River Linux Secure.

Browsing Changes

The Wind River Linux Secure kernel development methodology simplifies the following use cases for browsing and understanding kernel code:

- Showing changes, e.g., “What changes were made to foo.c?”
- Showing foo.diff
- Showing groups of changes, e.g., “Show me only the LTTng patches.”
- Comparing branches, e.g., “What’s different between the ixm27 and imx31 BSPs?”
- Completing annotation for all changes, e.g., “Where did feature X come from and why is it there?”
- Showing standard commit IDs, e.g., “I see a kernel change on another tree; is this included in my tree?”

Kernel Styles

The code base of the Wind River Linux Secure kernel supports many features that are available for specific applications but not necessarily suitable for all. Wind River provides predefined kernel styles that are specific to these applications. Wind River Linux Secure uses a modified CGL kernel style to meet the requirements of the EAL4+ GP-OSPP certification. This kernel includes upgrades such as shelf management, security, fault tolerance, threaded interrupt request lines (IRQs), and crash analysis that are not available in the other kernel types and functionality added specifically for the secure requirements such as enhanced password protection, screen locking, and network labeling. This kernel is a suitable jumping-off point for high availability solutions.

Networking Subsystem

The following are specific networking subsystem features:

- TCP/IP v4
- IPv6, MIPv6
- IPsec
- Stream Control Transmission Protocol (SCTP)

- VLAN tagging
- Transparent Inter-process Communication (TIPC)
- Network block device (NBD)
- cgroups and controllers (Control group support adds support for grouping sets of processes together, for use with process control subsystems such as Cpusets, CFS, memory controls, and device isolation. It includes net traffic controller, memlimit controller, dm-ioband bio_tracking, and group scheduling controllers.)
- Other RFCs (Contact Wind River for details.)

Security

The following are specific kernel security features that are included in Wind River Linux and available in Wind River Linux Secure:

- **BSD jail (bsdjail):** The Linux port of the FreeBSD “jail” facility provides the ability to partition the operating system environment while maintaining the simplicity of the UNIX chroot model.
- **Simplified Mandatory Access Control Kernel Support (SMACK):** This lightweight implementation of name-based security labels is useful for providing Mandatory Access Control (MAC) without a full SELinux policy.

Debugging and Profiling

The following are specific debugging features:

- **oprofile:** Kernel.org oprofile enhanced with tracing through the syscall boundary
- **ftrace:** Lightweight function tracing, includes dynamic ftrace (backport from 2.6.29) and early-ftrace for boot-time measurement enhancements
- **ptrace:** Process trace, single step, multi-threaded trace support
- **kprobes:** Kernel address trapping
- **KGDB:** Kernel debug support over serial, Ethernet, and console
- **lockdep:** Lock dependency checking and analysis
- **wrnote:** ELF image annotation for core dumb debug
- **On-chip debugging:** Bugs found and fixed before project milestones and deadlines
- **Linux Trace Toolkit (LTTng):** Extensible, lightweight kernel instrumentation for tracing program execution and debugging parallel and real-time behavior

- **Latency top:** Latency visualization support
- **Boot-time reduction:** Enhancements for measuring and streamlining boot time
- **Footprint reduction:** Kernel configuration and modifications to limit the runtime kernel footprint
- **kmemcheck:** Kernel memory checking and leak detection

File System

The following are kernel file system features:

- **Boot technologies:** ramdisk, execute in place (XIP), kernel libc support for boot environments (klibc), initial ramfs support (initramfs), fastboot (asynchronous boot/init)
- **Flash file systems:** yaffs, yaffs2, jffs, advanced XIP file system (axfs)
- **Logical volume manager:** LVM and LVM2
- **RAID:** Increased storage functions and reliability through redundancy
- **Network file systems:** NFS, smb
- **Disk file systems:** ext2, ext3, FAT, VFAT
- **Other file systems:** Stackable unification file system (unionfs), file system for large device scalability (logfs), compressed read-only file system (squashfs), compressed ROM file system (cramfs)
- **Revoke:** revokeat() system call for inode-based revocation

Input and Output

The following are input and output (I/O) features of the kernel:

- **I/O splice:** A system call that copies data between a file descriptor and a pipe, or between a pipe and user space, without actually copying the data
- **User space I/O:** Drivers that allow programs easy access to kernel interrupts and memory locations; used for user mode drivers
- **Asynchronous I/O:** Processing that permits other processing to continue before the transmission has finished

Real-Time and Deterministic Scheduling Behavior

The following are specific real-time and deterministic scheduling behavior features:

- **Voluntary kernel preemption (desk-top):** Kernel latency reduced by adding more explicit preemption points to kernel code

- **No forced preemption (none):** Traditional Linux preemption model
- **Robust priority inheritance mutex:** Robust and priority inheritance support for user space mutexes
- **High resolution timers (HRT):** Time measured at a microsecond-level resolution
- **Dynamic tick support (NOHZ):** Timer interrupts only triggering on an as-needed basis both when the system is busy and when the system is idle

Hardware Support

The following are highlights of hardware features:

- **Multi-architecture:** Three architectures (ARM, x86, PowerPC) and five subarchitectures (ARM, x86, x86-64, PowerPC, PPC64)
- **SMP/AMP/multi-core:** SMP safety of drivers and core kernel functionality
- **CPU isolation:** cpuisol
- **CPU hotplug:** Removal of CPU for provisioning to keep it off system execution path
- **IEEE float:** IEEE floating point conformance for PowerPC processors supporting Signal Processing Extensions (SPE)
- **SEC (Talitos Freescale Security Engine):** Hardware acceleration for PowerQuicc E processors
- **Peripherals:** Device drivers for peripherals such as audio, Ethernet, GPIO, SDIO, SCSI, MTD, serial, framebuffer, VGA (graphics), keyboard, USB (gadget, host, OTG), touch screen, PATA/SATA, sound, PMEM, wireless (Wi-Fi), Bluetooth, MTP

Other Features

- **kexec:** A system call that provides the ability to shut down the current kernel and start another without rebooting hardware
- **kdump:** Kernel crash dump
- **Kernel virtual machine (KVM):** OS virtualization for certain architectures
- **Clock API:** Wind River interface for manipulating clock sources and data sampling

Networking Features

These highlights describe features relevant to network equipment. They may overlap with the kernel features described previously.

System Black Box

Taking cues from the aviation industry, the persistent memory framework (PMEM) of Wind River Linux Secure provides a system black box acting much like the combined flight data recorder (FDR) and cockpit voice recorder (CVR). Scheduler decision history, logs of all exceptions, panic and console logs, kernel log messages, system reset and reboot logging, Linux Trace Toolkit (a set of kernel patches and supporting user space tools to control tracing) logs, even end-user defined events can all be logged to dedicated nonvolatile memory, external memory, peripherals, or even protected segments of normal system RAM. This enables faster recovery and better system uptimes by allowing all necessary debug information to be preserved by the PMEM driver in these protected regions of memory for later analysis while allowing the system to reboot and re-enter service immediately.

Transparent Inter-process Communication

As a major contributor and one of the maintainers of the Transparent Inter-process Communication (TIPC) project, Wind River actively develops this cross-platform, high-speed communications technology aimed specifically at clustered computing environments. TIPC is a communications protocol that provides developers with an extremely flexible means of creating distributed, cooperative applications that may migrate as required throughout the cluster seamlessly. Wind River continues to invest in TIPC, and Wind River Linux Secure remains up-to-date with developments in the TIPC project.

Security

Originally developed by the National Security Agency (NSA), SELinux remains the premier method of ensuring a flexible and trusted computing environment. SELinux is both a Linux Security Module (LSM)—a piece of the kernel that arbitrates access to all system resources based on the system policy—and a collection of supporting user space tools for developing, applying, enforcing, auditing, and debugging the security policy used by the LSM. Wind River Linux Secure includes

three levels of security out of the box for SELinux-enabled configurations and all the tools necessary to customize or develop new policies from the ground up.

Additionally, Wind River Linux Secure includes advanced, preemptive security technologies such as run-time stack and buffer overflow protection and a suite of tools that together provide a complete intrusion detection and prevention system. The Wind River Linux Secure kernel also includes the PaX patch set implementing least-privilege protections for memory pages or memory segments as well as GRSecurity, further kernel patches that build upon PaX and implement a trusted execution model, role-based access control, detailed system accounting logs, and fine-grained privilege separation.

Further user containment features that resist all known chroot-jail attacks protect your system even when deployed in a hostile environment or with an unknown user base.

Carrier Grade and Network Equipment

Wind River Linux Secure offers expanded support of some CGL features including better support for clustered computing environments and better support for developing and deploying highly available systems. These include persistent shared memory with the system black box, coherent user and kernel tracing framework with LTTng, run-time analysis tools, and common command-line tools such as strace and ltrace for doing system call and library tracing. The following are additional functions included in Wind River Linux Secure:

- **Coarse resource enforcement:** Wind River Linux Secure group scheduling and resource controllers allow memory and scheduling limits to be enforced on a group basis rather than simple per-process or per-object.
- **Layer 2 Tunneling Protocol (L2TP) support:** This tunneling protocol supports virtual private networks (VPNs).
- **File access tracing:** Linux kernel features such as inotify as well as Wind River Linux Secure features such as GRSecurity provide extensive logging and notification options for monitoring file access and recording system events.

Advanced Network Equipment Features

With Wind River–developed proven technologies such as the virtual management controller, it becomes easy to develop complex applications that can run on a variety of hardware using common intelligent platform management interface (IPMI) commands for health monitoring even on systems without a baseboard management controller (BMC). The application monitoring and migration feature (memmon) allows complex applications to transparently migrate between systems to ensure zero service interruption even in the case of a scheduled outage. Additional system engineering tools such as flexible out of memory (OOM) killer behavior, Ethernet link bonding, and statistics gathering and reporting on a per-socket and per-interface basis allow designers to engineer their systems to the absolute maximum capabilities of the hardware. This is combined with Wind River’s support for error detection and correction (EDAC) on new chipsets.

Network-Based Storage Solutions

Integrating technologies such as the distributed replicated block device (DRBD), multiple redundant communication paths to external storage over fiber channel links, ATA over Ethernet, the Oracle Cluster File System version 2 (OCFS2), and Internet Small Computer System Interface (iSCSI), Wind River Linux Secure provides functionality for centralized logging servers, centralized billing and accounting servers, and share file system servers.

Board Support Packages

Wind River Linux Secure board support packages (BSPs) are hardware-enablement components that contain elements such as drivers and settings needed to make Wind River Linux Secure support specific hardware.

BSPs are separable configuration components that can be created and added to Wind River Linux Secure at any time. In addition to the BSPs Wind River Linux Secure ships with, Wind River continues to add boards according to customer demand and hardware availability.

These additional BSPs are available on Wind River’s Online Support website to customers under an active platform subscription. Also, Wind River Services can create customer-specific BSPs if your hardware is not covered by the existing ones.

A typical BSP includes board-specific configuration files that overwrite or add configuration options defined by the common platform templates. Additional kernel patches included in the BSP can add new device drivers or apply necessary changes to existing Linux code. BSPs can also contain additional user space components or other files.

Wind River has validated proper operation of the Linux run-time for each supported reference board. The supported features are board-specific and depend on availability and maturity of the code in the open source community.

The product ships with four BSPs that have been EAL 4+ certified covering the following target processors:

- ARM
- Intel x86 32
- Intel x86 64
- PowerPC

BSPs are also created and shipped asynchronously, after the product is released. For this reason, the list of BSPs is not static. Contact Wind River to get an up-to-date supported BSP list with detailed descriptions of supported peripherals.

Applications

Wind River Linux Secure provides more than 200 integrated user space application packages. They implement functionality typical of an embedded Linux run-time. The Wind River build system (LDAT) generates binary RPMs from these sources. LDAT can then use these to generate a root file system.

Origins and Porting

A variety of open source projects forms the origins of the user space code base. About 50 packages are based on traditionally prepackaged trees containing source code, configuration scripts, and Makefiles or Makefile precursors (i.e., a classic package format). The remaining 150 have source RPMs as their base.

Wind River patches these upstream sources for integration and bug fixing. These packages generally contain the following types of patches:

- **Cross compilation:** Many packages are expected to be compiled on x86 architectures for x86 architectures. This often means host libraries can be referenced or linked in.
- **Multilib:** This ensures that packages can be built for both 32- and 64-bit targets.
- **Other defects:** Wind River ensures that packages are properly integrated together.

The Wind River Linux Distribution Assembly Tool, LDAT, will access the ported components, pass the appropriate cross-compilation parameters, and create a file system matching the target’s architecture and the kernel’s configuration and including the features needed.

Customers can add their own user space packages using the same mechanism. Instructions for this are included in product documentation. Adding new packages to the certified configuration will invalidate the certification. If a certified configuration is required, starting with Wind River Linux Secure will facilitate the certification of the updated software product or full system.

List of Packages

It is easiest to consider the package list in terms of categories used. The following is a list:

- Administration
- Basic C support
- Booting and startup
- Daemons
- Databases
- Debugging
- Devices
- DirectFB
- File systems
- File transforms
- Graphics
- Hardware
- Kernel
- Languages
- Middleware
- Networking
- Shells and scripting
- Security
- SELinux
- Setup
- System
- Host tools

- Test
- Utilities
- Various
- Wind River instrumentation

A full list of the package names can be found at the end of this document in “Appendix A: Package Summary by Category.” For a full list of package details (source package names, binary package names, versions, licenses, etc.), contact Wind River.

Profiles

Wind River includes several profiles that define preassembled root file systems and kernels for specific functionality. The following are the profiles shipped:

- General Purpose Operating System Protection Profile
- Labeled Security Protection Profile
- Controlled Access Protection Profile
- Role-Based Access Control Protection Profile

Optional Add-on Products

Wind River provides other products to implement functionality not available in the base product.

IPL Cantata++ for Wind River Workbench

IPL Cantata++ for Wind River Workbench (formerly Unit Tester), now available for Wind River Linux Secure, is a set of tools that allows developers greater efficiency in completing unit testing, integration testing, and code coverage analysis on the tests. The integration of Cantata++ with the Wind River Workbench development suite places these capabilities within easy reach. Cantata++ increases software quality, decreases time-to-market, and reduces support costs through better, faster, more automated testing in the development life cycle.

Wind River Workbench On-Chip Debugging

In the early stages of hardware and software development, a robust connection to the microprocessor through its run-control port is essential. Wind River Workbench provides connectivity between the host development environment and the target device via the JTAG or on-chip debugging interface of the microprocessor that resides on the device.

The on-chip debugging interface of most microprocessors enables full control of the microprocessor itself, access to core and peripheral registers, and access to on-chip switch fabrics and memory controllers, along with access to external buses and many devices attached directly to the bus. In addition, some microprocessors support either internal or external trace buffers, allowing developers to capture information regarding the exact code that ran on the target and when.

On-chip debugging provides developers with complete system-level control of their environments at all times, enabling more efficient and effective hardware bring-up, firmware development, and device driver and BSP generation. Specifically for Linux development, Workbench On-Chip Debugging provides visibility into hardware and software interactions for kernel and kernel modules and enables development and debug of user space applications. The JTAG-based debug capability is a useful alternative to agent-based debugging in applications where serial, Ethernet, or USB interfaces are not available or in environments where agent instrumentation of the operating system is not desired.

The Wind River debugger provided with Wind River platforms can be enabled for on-chip debugging. This capability, along with Wind River ICE, Wind River Trace, and Wind River Probe hardware, provides access to significant additional capability within Workbench.

For more information, visit <http://www.windriver.com/products/workbench>.

Wind River Test Management

Wind River Test Management is a scalable system that links device development and test teams with a collaborative suite of applications for efficient system testing and defect resolution. The system leverages a unique, dynamic instrumentation technology to measure code coverage, profile performance, and diagnose and repair the system at run-time. The product is designed to manage multiple devices under test at multiple lab locations, maximizing resource utilization and accelerating the testing process.

Benefits of Wind River Test Management 3.1 include the following:

- Higher quality, faster time-to-market, lower cost
- More testing, more often
- Faster defect resolution
- Management of progress, quality, and resources
- Benefit to both development and QA
- Powerful sensorpoint technology
- Open, scalable architecture
- Broad platform support

For more information, visit http://www.windriver.com/products/test_management/.

Wind River Network Management

Wind River network management includes advanced SNMP, CLI, and web-based management interface development tools. Our network management products are designed and implemented as cross-platform and validated on both VxWorks and Wind River Linux Secure. This makes it easy for developers to implement management interfaces for both VxWorks and Wind River Linux Secure-based devices or to migrate from VxWorks to Wind River Linux Secure.

The Wind River network management SDK includes a standards-based implementation of SNMP, consisting of SNMP v1/v2c/v3 and AgentX support as well as a scalable, unified, small-footprint management framework to create web-based, CLI-based, or custom management interfaces to manage networked elements. The scalable framework consists of a management backplane that acts as a conduit for data handling between management interfaces (consumers) and manageable elements (producers); it can have any type of consumer and any type of producer.

Wind River network management SDK 3.2 includes the following:

- Wind River SNMP 10.3
- Wind River CLI 4.7.1, Wind River Web Server 4.7.1, and Wind River MIBway 4.7.1
- Web and CLI-based network management interfaces
- Standalone web server: HTTP and HTTPS
- Integration with SNMP via MIBway
- Wind River Management Integration Tool (Windows host support only)
- Management Configuration Editor

Wind River SNMP

The Simple Network Management Protocol (SNMP) is designed to facilitate management and configuration of networked devices. Wind River SNMP is a highly portable, memory-efficient, and standards-compliant implementation of SNMP specifically designed for original equipment manufacturers (OEMs) and system integrators who require full compliance with SNMP standards in a fast, small SNMP agent. This complete solution for integrated SNMP design and implementation includes a full MIB development platform. It is composed of SNMP v1/v2c/v3 and AgentX.

Features of Wind River SNMP 10.3 include the following:

- Bilingual SNMP agent supports SNMPv1/v2c protocols
- Asynchronous support
- SNMPv3 security features
- SNMP notifications
- "Target" and "notify" MIBs
- SNMP proxy
- SNMP v1/v2/v3 coexistence
- AgentX module
- MIB compiler
- Compact, interoperable, standards-based configuration
- Integration and validation with Wind River Advanced Networking Technologies (Interpeak-based)
- Portable design and implementation
- A new API to support SNMPv3 INFORM PDU operations

Testing and Validation

Wind River is committed to providing quality products for both proprietary and open-source-based technologies. Our quality policies include formal code inspections, peer reviews, project reviews, program audits, and traceable requirements change management. Wind River Linux Secure was created following a methodical process to thoroughly test key features on every supported reference configuration (defined by development host, kernel and package configurations, and supported board). In addition, Wind River Linux Secure completed an independent test validation by an external certification body to receive final EAL 4+ certification. This was completed on the GP-OSPP configuration.

Automated testing packages for Wind River Linux Secure include the following:

Test Suite	Description
Automated Boot Login Test	This tests the booting process of any target architecture for a given kernel and rootfs. The process is completely automated for a set of targets, which helps in determining the boot sanity of the target.
CD Sanity Test	This automation suite covers CD installation on a new release, followed by building the rootfs for various target combinations using prebuilt RPMs. It boots the target with the prebuilt kernel and rootfs and executes KGDB and user-mode tests on the target, then reports the results to the database.
Linux Test Project (LTP)	This test suite validates the reliability, robustness, and stability of the Linux kernel and its network components.
Open HPI	This is the Open Hardware Platform Interface (Open HPI) conformance test.
Open POSIX	This test suite is for POSIX 2001 APIs not tied to specific implementations. It provides conformance, functional, and stress testing, with an initial focus on threads, clocks and timers, signals, message queues, and semaphores.
Perl_test	This tests the Perl package.
RT Feature Testing: LMBENCH Realfeel	Real-Time Feature Testing tests performance. LMBENCH is used to measure I/O of the kernel. Realfeel tests scheduler behavior.
Safest	This tests the Open HPI package.

In addition to automated testing, significant manual testing—including feature testing, Workbench testing, and complete system testing—for Wind River Linux Secure has been completed.

Test Suite	Description
Bug Fix Testing	Wind River Linux 3.0 bug fixes have been tested and resolved in Wind River Linux Secure.
Wind River Linux Secure Certification Testing	Wind River Linux Secure has gone through extensive testing for secure functionality: <ul style="list-style-type: none"> • Labeled network testing • Audit testing • LTP testing • LTP add-on testing • OSPP testing • LSPP testing • CAPP testing • RBACPP testing
Board-Specific Testing	The specific boards that were certified have gone through extensive testing. These tests were executed as part of regression testing on Wind River Linux Secure.
Documentation Testing	Documentation for Wind River Linux Secure was tested to make sure all steps are properly recorded.
Host OS Testing	Installation testing was done on various host OSes supported for Wind River Linux Secure as well as sample application build and debugging from Workbench and platform build.
HRT Regression Testing	HRT features for previous releases were tested for regression on the supported platforms.
Pre-PRT Testing	Some user scenarios for Workbench, Wind River Run-Time Analysis Tools, and the build system were tested on supported hosts and platforms, as done by the PRT.
RT Regression Testing (based on Wind River Linux 3.0.3)	Regression testing features for previous releases were tested for regression on the supported platforms.
SNMP Testing	SNMP tests were performed.
Stress Testing	Stress tests were performed.
Usability Testing	Wind River Linux 3.0 and Workbench 3.1 usability testing is based on the usability testing document.
Use Case Testing	The use cases for Workbench, Run-Time Analysis Tools, and build system were tested on supported hosts and platforms.
Workbench Integration Testing	Wind River tests the feature integration of Workbench with Wind Manage, System Viewer, and Run-Time Analysis Tools.

Wind River has developed a robust, scalable, and automated build and test infrastructure with more than 4,000 test cases and 301,336 test runs. This infrastructure supports many processor architectures and uses a combination of commercial, open source, and proprietary tests, including LTP Core, LTP Network, LSB, TAHI, and Open POSIX. Wind River uses coverage tools, such as gcov and lcov, to optimize test development and close gaps in existing test suites.

Partner Ecosystem

Wind River's world-class partner ecosystem ensures tight integration between our core technologies and those of the premier hardware and software companies we've chosen to build out our solutions. Our partners help extend the capabilities of Wind River Linux Secure by offering out-of-the-box integration and support for key technologies in a number of fast-moving markets. Our team is trained to troubleshoot partner technologies in use with Wind River products, making ours the best-supported ecosystem in the embedded and mobile software industry.

The Wind River Partner Ecosystem is constantly expanding. Contact us for more details or visit <http://www.windriver.com/partners/>.

Professional Services

Wind River Professional Services, a CMMI Level 3–certified organization, enables you to reduce risk and focus on development activities that add value and differentiate design. As part of our comprehensive solutions, Wind River offers a Linux Services Practice, with focused offerings that help you meet strict market deadlines while keeping development costs down. Our experienced team delivers device software expertise that solves key development challenges and directly contributes to your company's success. Backed by our commercial-grade project methodology, Wind River Professional Services include device design, Linux BSP and driver optimization, software system and middleware integration, and legacy application and infrastructure migration.

Education Services

Education is fundamentally connected not only to individual performance but to the success of a project or an entire company. Lack of product knowledge can translate into longer development schedules, poor quality, and higher costs. The ability to learn—and to convert that learning into improved performance—creates extraordinary value for individuals, teams, and organizations. To help your team achieve that result, Wind River offers flexible approaches to delivering product education that best fit your time, budget, and skills development requirements.

Personalized Learning Program

Wind River offers a unique solution to minimize the short-term productivity drop associated with the process of adopting new device software technology and to optimize the long-term return on investment in a new device software platform. The Wind River Personalized Learning Program delivers the right education required by individual learners to accomplish their jobs. The program identifies work-related skill gaps, generates development plans, materials, and learning events to address these skill gaps, and quantifies the impact of the development activities for each individual user.

This programmatic, focused, and project-friendly approach to skills development results in a significant increase in the personal productivity of your teams, improved efficiency in the processes they employ, and faster adoption of the technology you have purchased. Personalized Learning Programs deliver improved business performance—customers have reported a return on investment ranging from 18 percent to 80 percent over a traditional training approach. Consult your local Wind River sales representative for more information on Personalized Learning Programs.

Support Services

Wind River Customer Support, a certified Service Capability and Performance (SCP) organization, provides support for Wind River Linux Secure. Your subscription to Wind River Linux Secure includes full

maintenance and support, delivered through Wind River's Online Support website and our worldwide support team. Wind River Support includes the development suite and cross-toolchain, Linux kernel, and the reference root file system, as validated on supported boards and development host operating systems. While under subscription, customers receive both maintenance updates and major upgrades.

Technical Support

Wind River works with every customer to help you solve technical support problems. We may not be able to support every configuration of hardware and software that a customer may have selected, but we will do everything we can to provide support. Linux Technical Support on modified or unsupported configurations is best-effort-based. Wind River Customer Support will try to reproduce the problem on a supported configuration. If the problem can be validated, we will provide a fix that will be tested on a supported configuration. Wind River Professional Services can provide support for boards or host operating system versions that are not supported by the standard product, as well as for customized versions of the source code or additional nonstandard packages.

Customer Support will provide bug fixes following the process outlined in Wind River's Customer Support User's Guide (CSUG), available at <http://www.windriver.com/support/resources/csug.pdf>.

If appropriate, Wind River will submit changes in open source code to the open source project maintainer for inclusion in a future release of the open source package. Wind River will maintain changes until a new version from the open source project is available and can be released for Wind River Linux Secure.

Customers with a valid support or subscription agreement are eligible for all respective updates free of charge. If customers cannot update to a new version but need critical parts of the update applied to an older version of the product, Wind River Professional Services can be engaged to backport the required functionality on a case-by-case basis.

Visit Wind River Online Support (OLS) for fast access to product manuals, downloadable software, and other problem-solving resources. OLS offers a comprehensive knowledge base with a robust search feature for locating product information and manuals by keyword, author, published date, document type, language, and solution

category. OLS also provides new BSPs, updates to existing packages, patches, manuals, the latest errata, and other announcements about Wind River Linux Secure. Wind River will also provide new contributed Linux packages through our support website. These packages have been contributed by the open source community and are prebuilt and tested with Wind River Linux Secure.

Additional support features, including proactive email alerts covering particular technologies, platforms, or product patches and technical tips for common problems, are available for all customers on subscription. OLS visitors can also access a community of developers to discuss their issues and experiences.

Appendix A: Package Summary by Category

If you are interested in more detailed package information, contact Wind River.

Administration	evlog, memstat, openais, quota
Basic C support	binutils, boost, glibc, libaio, libatomic_ops, libcap, libdrm, libgcc, libstdcxx, prelink, wrs_kernheaders
Booting and startup	grub
Daemons	acpid, audit, crontabs, daemontools, fam, izone, iscsi-initiator-utils, mcelog, pcsc-lite, pulseaudio, quagga, samba, vixie-cron, vsftpd, wcf, xinetd
Databases	hwdata, libtermcap, mysql, openldap, postgresql, python-ldap, sqlite, unixODBC
Debugging	eventlog, gdb, kexec-tools, libevent, logrotate, ltrace, oprofile, smartmontools, strace, sysklogd, syslog-ng, sysstat, watchdog
Devices	ccid, device-mapper, device-mapper-multipath, ethtool, ipmitool, ipmiutil, kbd, libfakekey, libusb, lm_sensors, madev, mingetty, minicom, nbd, openipmi, parted, pciutils, scsidev, setserial, udev, usbutils, vblade
DirectFB	dfbtutorials, directfb, directfb_headers
File systems	acl, attr, dmapi, drbd-tools, e2fsprogs, filesystem, fuse, installsw, lsof, lvm2, mdadm, mtd-utils, rdist, rsync, samhain, xfsdump, xfsprogs, yaffs2
File transforms	bzip2, cpio, gzip, libidn, lzo, shared-mime-info, tar, unzip, zip, zlib
Graphics	atk, cairo, fbset, fontconfig, freetype, glib2, gtk, libjpeg, libpng, libtiff, ncurses, pango, sdl, SDL_image, SDL_mixer, SDL_ttf, tslib
Hardware	hal, hal-info
Kernel	hotplug, kvm, pth
Languages	ruby
Middleware	ace, paste
Networking	gent-proxy, aoetools, apache-ssl, atftp, bind, boa, curl, dhcpheartbeat, ifenslave, inetutils, iproute, iptables, iputils, klibc, libnet, libnl, libpcap, lkscpt-tools, lrzsz, mailx, mipv6-daemon-umip, netcat, net-snmp, net-tools, nfs-utils, nfs-utils-lib, nspr, portmap, ppp, pyca, radvd, rdate, rsh, sendmail, socat, tcpdump, telnet, tipc_demo, tipc-utils, tnftp, traceroute, tunctl, usagi-tool, vlan, wget
Shells and scripting	bash, expect, gawk, grep, less, microp Perl, pcre, perl, perl_tests, perl-Convert-ASN1, perl-LDAP, perl-XML-Parser, python, python-imaging, sed, tcl, xerces
Security	adduser, beecrypt, cracklib, ecryptfs-utils, freeradius, gnupg2, gnutils, gradm, ipsec-tools, keynote, keyutils, krb5, libassuan, libgcrypt, libgpg-error, libmsec, libsepol, logcheck, lssp-eal4_config, nss, ospd-utils, opendiameter, openssh, openssh-sftp-only, openssl, pam, pam_passwdqc, passwd, shadow-utils, stunnel, sudo, tcp_wrappers, vlock
SELinux	libselinux, libsemanage, mcstrans, policycoreutils, pyetree, repolicy, repolicy-strict, sepolgen, setools
System	checkpolicy, crackerjack, hdparm, ipmi-test, ltp-full, ocfs2-tools, oncpu, openhpi, posixtestsuite, robust-tests, saftest, screen, unionfs
Host tools	chkconfig, db4, elfutils, expat, flex, libtool, libxml2, neon, paxctl, rpm
Test	application_args_proprietary, crypto_proprietary, cyclictst, hello_proprietary, lmbench, low_latency_mem_proprietary, m4, mailbox_proprietary, named_block_proprietary, perl-net-telnet, queue_proprietary, traffic_gen_proprietary, uart_proprietary, wifitest, xreg, xts, zebra
Utilities	at, bc, bootlogger, bridge-utils, coreutils, diffutils, duplicity, file, findutils, gettext, gmp, inotify-tools, make, mhash, mktemp, mpatrol,mtree, ncftp, newt, ntp, popt, readline, slang, star, sysfsutils, syslinux, time, timezone, ustr, util-linux, util-linux, which
Various	cyrus-sasl, ed, ElectricFence, freeglut, gdbm, gpm, mce-dev, vim
Wind River instrumentation	wbagent-pttrace, wr-coverageagent, wr-opagent, wrproxy, wrsv-ltt

Appendix B: Supported Target Boards

The following boards are supported and EAL 4+ GP-OSPP certified in the Wind River Linux Secure distribution:

- Intel 5500 server reference board (S5520HC, Hanlan Creek)
 - intel_5500_server
- Common Intel-based 64-bit PC platform (D630)
 - common_pc_64
- Texas Instruments OMAP3530 evaluation module
 - ti_omap3530_evm
- Freescale 8572ds reference board
 - fsl_8572ds

New boards can be requested and taken through certification.

Supported Host	Architecture	Platform Developer	Application Developer
Windows XP Professional	x86 32-bit		✓
Windows Vista Business	x86 32-bit		✓
Windows Vista Enterprise	x86 32-bit		✓
Fedora 9	x86 64-bit	✓	✓
Red Hat Enterprise Linux Workstation 5	x86 32-bit	✓	✓
Red Hat Enterprise Linux Desktop with Workstation 5	x86 32-bit, x86 64-bit	✓	✓
SUSE Linux/openSUSE 11	x86 32-bit	✓	✓
Novell SUSE Linux Enterprise Desktop 10	x86 32-bit, x86 64-bit	✓	✓
Ubuntu Desktop Edition 8.0.4	x86 64-bit	✓	✓
Sun Solaris 9 (Update 9/05, GTK only)	SPARC 32-bit		✓
Sun Solaris 10	SPARC 32-bit		✓

Appendix C: Supported Development Hosts

The following table contains a complete list of supported development hosts with the necessary updates. It lists which hosts support the Application Developer package only and which hosts also support the Platform Developer package.

Note that although development may be possible on other Linux distributions and versions, Wind River has not certified the product on them.

For more details on features of Wind River Linux Secure, contact Wind River or visit <http://www.windriver.com/linux/>.