

INTEGRITY Operating System for Desktop PCs, Servers, Thin-Client Workstations, and PDAs

An operating system that can simultaneously support ubiquitous legacy applications (such as those that run on Windows and Linux) along with mission-critical applications that have a high security assurance and/or real-time requirement is a holy grail of computing (Figure 1). Without such a capability, system designers need to use multiple hardware devices to meet requirements: one device for each security domain running legacy applications and another for critical applications. This kind of hardware separation poses a tremendous burden in terms of cost and flexibility. An operating system that supports secure partitioning, legacy applications (including a rich human user interface), multi-level communication, secure user authentication and trusted path, and secure cross-domain information transfer is needed for the next generation of secure computing devices. Green Hills Software's INTEGRITY Operating System provides a highly-assured environment capable of meeting these requirements.

Application of the INTEGRITY Operating System

Potential applications of the INTEGRITY Operating System are widespread:

Defense IT Infrastructure

Within intelligence agencies and the DoD, IT users and administrators wrestle with multiple computers handling information at varying security levels. For example, personnel may require simultaneous access to the Joint Intelligence

Communications System (JWICS), the Secret Internet Protocol Router Network (SIPRNET), the Unclassified but Sensitive Internet Protocol Router Network (NIPRNET), and the Internet. Multiple computers are needed because no computing systems on the market today provide a sufficient level of assurance that information is properly secured within its appropriate security domain. An unclassified computer connected to the Internet may be infected with viruses, spyware, and Trojan horses that can compromise the integrity, availability, or confidentiality of top secret or secret data that is read or written by authorized IT personnel. So the current solution to this risk is to separate different security levels using multiple sets of hardware, resulting in a tremendous space, cost, and maintenance burden. Instead, the INTEGRITY Operating System can be used to run the desktop PCs, thin-client workstations, and servers used in these environments.

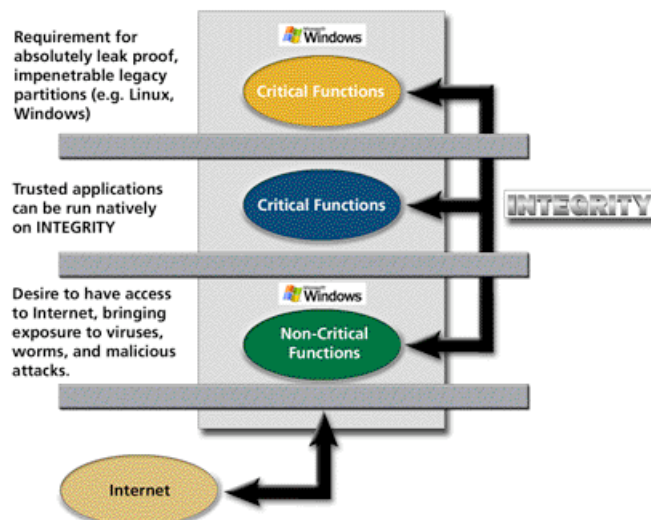


Figure 1—critical applications and non-critical, insecure applications can coexist on the same PC.

Command and Control Systems

Within deployed defense infrastructure, data at multiple security levels must be manipulated by military personnel on the battlefield. Again, the lack of high assurance multi-level computing systems requires these command and control systems to support multiple computers for processing commands and data transfers at different security levels. With the high assurance INTEGRITY Operating System, admirals and generals can use a

single handheld PDA or workstation to simultaneously communicate with high-level policy makers, coalition forces, and the Internet. They can even use Excel spreadsheets and PowerPoint presentations.

Industrial Control Systems

The human operator at the nearest nuclear power plant sits at a computer screen, reading and tweaking the virtual dials, gauges, and controls that monitor the plant. These dials, gauges, and controls may be Visual Basic GUIs running on Windows! A single bug or virus could compromise the control system. With the INTEGRITY Operating System, the high criticality interface programs could be ported to run natively on the secure microkernel while leaving less critical Windows applications (including the Internet connection) in an insulated partition.

Automotive Infotainment Systems

In the back seat of a high-end luxury car, you may find yourself watching a video screen with that familiar Microsoft Windows environment from which you can browse the Internet, play music and movies, and read Office documents. However, when the car was started, you had to stare at a blank screen for a minute waiting for Windows to boot—not something to which passengers are accustomed. In addition, some multimedia applications do not perform optimally due to the lack of real-time processing support in Windows. With the INTEGRITY Operating System, real-time processing can be distributed to the real-time microkernel. Furthermore, music and video are available in milliseconds (because INTEGRITY boots in milliseconds). On the same in-car PC, the INTEGRITY Operating System provides a secure screen running Windows, and now you can browse your Office files on the same system. The combination of secure, real-time performance coupled with the ability to run legacy software applications provides consumers with the responsiveness and reliability they expect without sacrificing their familiar environment.

Home Office, Set Top Boxes, and other Consumer Computing Systems

Let's face it – we have a love-hate relationship with our PCs. We love our email, internet brows-

ing, watching DVDs, and running loads of useful applications ranging from games to word processors. We hate the viruses, spam, spyware, and blue screens of death. We use our PCs with the express knowledge that new security vulnerabilities will continue to be discovered in the operating systems and other critical software components upon which we depend. We know full well that hackers and cyberterrorists will exploit some of these flaws and perpetrate attacks on our PCs via the Internet. All over the world, consumers ask themselves the same question: can we have the benefits of modern computing without the security risks? Using the INTEGRITY Operating System, we can.

The security and functionality benefits of the INTEGRITY Operating System extend to many other industries and applications—banking, finance, amusement park control systems—essentially any computing platform that requires a high assurance for correct, safe, reliable, and secure operation along with a robust user environment.

INTEGRITY Operating System Architecture

To build a secure server, desktop PC, thin-client workstation, or PDA, you need an operating system that can provide critical security guarantees. Key components must be carefully designed to ensure that those security guarantees are met. But to run a broad range of applications, you need to provide all of the capabilities of a mainstream, general purpose operating system. The INTEGRITY Operating System Architecture meets both these needs. The INTEGRITY Operating System consists of:

- ▲ The INTEGRITY Separation Kernel, which provides the secure foundation for the rest of the system;
- ▲ Secure virtualization technology that enables multiple legacy operating systems (e.g. Windows or Linux) to run on top of the separation kernel where each copy of the operating system runs in a separate security domain, completely partitioned from each other;
- ▲ High assurance network client applications that enable the INTEGRITY Operating System to control a thin client workstation, providing

secure access to remote servers running at varying criticality levels;

- ▲ High assurance device drivers and other systems services that enable standard, off-the-shelf devices (keyboard, monitor, mouse, Ethernet, USB) to be used securely;
- ▲ A high assurance window manager that provides secure login/authentication and manages multiple virtual windowing environments for each partition and enables users to switch between partitions securely;
- ▲ A secure “information transfer agent” infrastructure that enables data to be moved between partitions if consistent with system-specific security policies.

Let’s discuss each of these components in turn.

INTEGRITY Separation Kernel

Unlike desktop operating systems such as Windows and Linux, the INTEGRITY separation kernel is extremely limited in the services it attempts to provide: hardware initialization, device control, application scheduling, and application partitioning. This last feature is arguably the most important. By enforcing a separation policy (hence the appellation “separation kernel”), INTEGRITY guarantees that multiple independent environments cannot affect each other. A malicious application running in a Windows partition cannot steal resources, corrupt or read data, or otherwise harm the other partitioned environments.

The INTEGRITY separation kernel, used for many years in embedded systems, has been certified as part of FAA DO-178B Level A flight critical systems and is currently undergoing a Common Criteria EAL6+ high assurance evaluation (no other operating system has ever been evaluated at higher than EAL5). In addition, INTEGRITY has been used in a variety of other mission and safety-critical applications, including Class 3 safety-critical medical devices, safety-critical industrial control systems, U.S. Government cryptographic devices, and flight control and avionics systems in platforms such as the F-35 Joint Strike Fighter and the Sikorsky S-92 helicopter. The same, proven high assurance development process—including (but not limited to) con-

figuration management, requirements specification and design, structured testing, covert channel analysis, and secure maintenance and delivery—that has been used to develop the INTEGRITY separation kernel has been used to develop the other high assurance components of the INTEGRITY Operating System.

The separation kernel bears a tremendous burden in achieving overall system safety and security. Because the kernel controls the fundamental resources (e.g. memory, execution time) of the computer’s central processing unit, it has the power to prevent unauthorized use of these resources. Conversely, if the kernel fails to prevent or limit the damage resulting from unauthorized access, disaster can result.

Operating system security is not a new field of research. Yet today there are no operating systems that have been successfully evaluated at the highest levels of assurance—EAL 6 to 7—the highest assurance levels of the Common Criteria for Information Technology Security Evaluation (ISO/IEC 15408), an internationally conceived and accepted security evaluation standard. One of the reasons for the lack of secure operating systems is the approach taken in the past to achieve security. Legacy security kernels attempted to provide a kitchen sink of services—protection and partitioning, mandatory access controls, secure file systems, and secure network services. As a result, these systems were simply too large and complicated to evaluate at high assurance levels.

Green Hills Software, with its INTEGRITY separation kernel, has taken a new approach that attempts to divide and conquer the problem of operating system security. INTEGRITY adopts the MILS (Multiple Independent Levels of Security) architecture which stipulates a layered approach to security. At the foundation is the MILS separation kernel, a small, real-time microkernel that implements the following functional security policies:

- ▲ **Information flow**—Information can not flow between partitioned applications unless permitted by the mandatory system security policy.
- ▲ **Data Isolation**—The data within partitioned applications can not be read or modified by other applications.

- ▲ **Damage limitation**—If a partitioned application is damaged by a bug or virus, this damage can not spread to other applications.
- ▲ **Periods processing**—When switching from execution of one partitioned application to another, no latent information (such as data on the stack or in registers) from the old partition can be read by the new partition; in other words, the kernel must purge/scrub any resources of information before they can be reused.

The separation kernel realizes these policies by using the microprocessor's memory protection hardware to prevent unauthorized access between partitions and by implementing resource allocation mechanisms that prevent one partition's operation from affecting another (e.g. by exhausting a resource such as memory or CPU time).

The MILS architecture also specifies enforcement of these policies such that they are:

- ▲ Non-bypassable
- ▲ Always invoked
- ▲ Tamper proof
- ▲ Evaluatable

The requirement that the policy enforcement be evaluatable is absolutely critical and is the reason why the separation kernel enforces this limited set of policies. Since a high assurance Common Criteria evaluation requires a formal model and mathematical proof, a component consisting of too much code becomes too difficult and expensive to evaluate. The MILS security policies can be implemented with a microkernel that is small enough to be evaluated at the highest assurance level.

Under the MILS concept, higher level secure software, such as a secure communications mechanism, web server, or file system, can be layered on top of the separation microkernel. The MILS security policies are recursive: a MILS file system, using the fact that underlying separation kernel enforces its partitioning security policies, is used to ensure file system data isolation, information flow, and damage limitation properties. In addition, multi-level security (MLS), as is realized in

the INTEGRITY Operating System's window manager, can be built on top of the MILS components.

The result of the separation kernel paradigm is that it allows software at varying levels of criticality to run on a single microprocessor. An application containing classified data and algorithms can occupy one partition while another partition is connected to the unclassified Internet. The MILS security policies, if assured at the highest level, make this possible. This can lead to enormous cost savings in product development because complicated multi-function applications can run on a single powerful microprocessor without requiring all of these applications to be evaluated at the highest assurance level.

Virtualization Technology

A virtual machine is a software application that mimics a hardware platform so that the software (e.g. operating system and applications) that normally runs directly on the hardware platform can instead run under control of software. Since a virtual machine is itself just software, multiple copies of the virtual machine can run on a single computer. Consequently, multiple PC environments can run on the same computer. Data can be easily transferred between the environments (as long as the transfer is consistent with the system security policies).

To do this well is not easy. Security and performance are often at odds, and the virtual machine needs both. A virtual machine has to perform a lot of the work that the hardware did and do it in such a way as not to jeopardize the overall security of the system. Virtual machine technology is not new, but the INTEGRITY Operating System implementation, called Padded Cell™, is. Existing commercial virtual machines run in supervisor mode (on Intel Architecture, this is also known as "Ring 0"), which means that they have direct control of all the physical resources (CPU time, memory, devices) of the computing hardware. A bug or security flaw in this code can therefore have disastrous consequences. In addition, these virtual machines were not designed with security in mind and therefore cannot provide a high level of confidence that the system security policies can not be compromised. A better approach, adopted by

Padded Cell, is to run the virtual machine as a user-mode application and use the separation kernel to control the hardware and run the virtual machines. If the virtual machines run in user mode, then only the virtual PC environment running under the control of the virtual machine application is at risk if there are any bugs or security vulnerabilities in the virtual machine itself. Essentially, the virtual machine must then only have the same security assurance level as the operating system it is hosting—in the case of Windows and Linux, this is a relatively low level of assurance. Figure 2 depicts the layering of virtualization technology to support guest operating system environments.

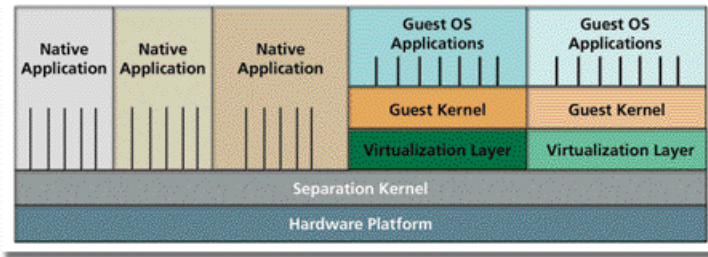


Figure 2—Hybrid legacy and native application architecture using separation kernel and virtual machines.

Thin Client Applications

The INTEGRITY Operating System was designed to support secure access to multiple remote servers, running at varying criticality levels. The INTEGRITY Operating System provides thin client communications applications, such as a VNC (Virtual Network Computing) client. The remote desktop is displayed in its own secure window by the secure window manager. The thin client paradigm is ideal for certain environments, such as ship-board command-and-control systems, because each operator uses a low-cost PC to control one or more, physically-isolated, powerful servers. A dedicated thin client may not even have a hard disk or any other hard media, saving precious space and power. In addition, a thin client implementation provides mobility to its users. For example, a weapons officer aboard ship can use his access card to login to an INTEGRITY powered workstation in his personal quarters, leave the room, and then use his card to access the same environment from another thin client workstation in the control room. Thin clients are easier to maintain, administer, and physically secure. Dedicated thin client workstations are cheaper than full-featured desktop computers, reducing overall network and system costs while still providing the same high assurance con-

trol over information at varying classification and sensitivity levels.

Organizations can develop their own custom graphical thin client interfaces, taking advantage of Green Hills Software-provided middleware such as X-windows servers, OpenGL, Java, and POSIX.1 API that

has been certified conformant with INTEGRITY (<http://get.posixcertified.ieee.org/register.html>).

Secure Device Drivers and System Services

The INTEGRITY Operating System manages the multiplexing of devices to ensure that each partition appears, during its own execution, to be running on its own PC despite the fact that the partitions are running concurrently. The INTEGRITY separation kernel handles the scheduling of these concurrent functions and enables a variety of shared devices, including the mouse, keyboard, and video monitor, to be accessed securely by multiple users and processes across multiple security domains. Typically, sharing hardware resources, such as the PC's PCI bus, creates possible covert timing channels which could be exploited by a Trojan horse in a critical partition to send information to a partition that is not authorized for that information. In addition to the major goal of separation and virtualization, the INTEGRITY Operating System is designed to mitigate these covert channels. The INTEGRITY Operating System accomplishes these goals by isolating each virtual environment from direct hardware access and by enforcing a strict resource partitioning policy.

The INTEGRITY Operating System provides multiple physical Ethernet interfaces to support communication with multiple networks running at varying security levels. The TCP/IP stack itself is also partitioned to securely handle each network interface without risking any illicit data transfer between them. Using the fundamental MILS security policies enforced by the separation kernel, this networking architecture guarantees that, even

in the midst of a network saturation attack on an unclassified network interface, the critical network interfaces are guaranteed to have the processing time and bandwidth they require. An alternative to using dedicated networking interfaces is to employ the Partition Communication System (PCS), a software implementation that provides partitioning of multiple levels of information across a network.

The INTEGRITY Operating System provides built-in audit management services that enable authorized users and applications to access and manage the log of all security-relevant events, such as a cross-domain information transfer. Audit logs can be viewed in a securely-labeled window by authorized personnel and archived for future analysis.

INTEGRITY Desktop

The INTEGRITY Desktop labels each window with its appropriate criticality level, according to standard secure labeling techniques. The mouse can only be moved within the foreground window. This prevents unintentional mouse events from occurring in other security domains. When a user needs to interact with a different security domain, that domain's dedicated window must explicitly be brought to the foreground.

The INTEGRITY Operating System manages virtual screen buffers for partitions at varying security levels. No partition has direct access to the computer's graphics card. Rather, each partition is provided a virtual screen buffer that it uses for a virtual display. The INTEGRITY Desktop contains a secure agent responsible for forwarding screen information when appropriate to the physical video card. The INTEGRITY Desktop also manages and routes, in a secure fashion, input device requests (i.e. from the mouse).

In addition to a multi-level video screen, INTEGRITY Operating System supports multi-headed monitors. Multiple level environments can then be displayed on the different monitor screens, a requirement for some types of command-and-control systems where each monitor has a dedicated purpose.

Another example of secure attention sequence (SAS) is the use of ctrl-alt-delete to access the login/authentication screen. INTEGRITY's mandatory access control policies and high assurance implementation guarantee that this sequence cannot be spoofed or intercepted by user-mode applications. The SAS provides the user with a trusted path mechanism for logging in to and out of the INTEGRITY Operating System. The INTEGRITY Operating System also provides secure drivers and interfaces for smart cards and other access devices used in conjunction with the login dialog.

Information Transfer Agent

The INTEGRITY Operating System, as defined by configured security policies, separates information at varying sensitivity levels. Authorized users, however, will need a secure mechanism by which documents can be moved between partitions. Consider a user who needs to take data from an encrypted email sent across the Internet and then needs to use this data for critical operations. The Information Transfer Agent (ITA) ensures that no malicious code can be brought into the critical partition. ITA supports secure "cut and paste" which allows on-screen data in one security domain to be transferred to another window in a different security domain. Since policies regarding cross-domain data transfer are system-dependent and may require human intervention (either locally or from a remote location), the INTEGRITY Operating System supports a modular re-grading architecture, providing hooks for customization of its policy manager.

The ITA optionally provides a graphical user interface for moving documents between security domains. The user interface has a standard Windows look and feel (e.g. drag and drop document to a special "ITA" icon). When a document is moved from a non-critical partition to a critical partition, the same high assurance re-grader application running natively on INTEGRITY is invoked to authorize or deny, as appropriate, the transfer.

Conclusion

For 22 years, Green Hills Software has been developing and deploying innovative products in the areas of operating systems, tools, and security. The INTEGRITY Operating System builds upon the commercial success of the INTEGRITY separation kernel, bringing high assurance security to the command-and-control, IT, enterprise, and other security and safety-critical environments that have a requirement for a rich, robust computing environment while managing operations at varying levels of criticality.

On the military side, the DoD believes future combat success depends on building the Global Information Grid (GIG) that will network together assets—soldiers, aircraft, ships, tanks—so that information can be efficiently and accurately distributed across theaters of operation. The security ramifications of this plan are staggering. Think Internet. A single affected node on the GIG could spread like a cancer, placing missions and lives at risk.

Internal attacks are also a major concern; a backdoor can be targeted to inflict the worst possible damage at the worst possible time, and it is well known that intricately placed subversions often escape source code inspection. The U.S. did it in the 80s. Thomas Reed, former Secretary of the Air Force, recounts in his book “At the Abyss: An Insider’s History of the Cold War” how the CIA inserted a Trojan horse into a company’s control software that the Soviets were planning to use for the trans-Siberian gas pipeline. The sabotage eventually resulted in a three kiloton explosion. We should expect our enemies to attempt similar infiltrations.

High assurance solutions, such as the INTEGRITY Operating System, are needed to combat these security risks and provide a future of safer and more secure computing.



30 West Sola Street ▲ Santa Barbara, CA 93101 ▲ ph 805.965.6044 ▲ fax 805.965.6343 ▲ www.ghs.com

Green Hills Software, the Green Hills logo, and INTEGRITY are registered trademarks of Green Hills Software. All rights reserved. All other trademarks are the property of their respective companies. © 2005 Green Hills Software, Inc. v1 0405