RED HAT ENTERPRISE LINUX:
YOUR SOLARIS ALTERNATIVE
TABLE OF CONTENTS

3 INTRODUCTION
4 BUSINESS REQUIREMENTS
  4 Availability of Applications
  4 Strength of ISV Support
5 APPLICATION MIGRATION CONSIDERATIONS
6 PERFORMANCE
6 CUSTOMIZATION
6 TOOLS
7 BENCHMARKS
  8 Proof of Concept
  8 Bottom Line: Performance
8 RELIABILITY AND STABILITY
9 AVAILABILITY AND SCALABILITY
9 SECURITY
  10 Capabilities
  11 Bottom Line: Security
12 TOTAL COST OF OWNERSHIP
12 FEATURE COMPARISON
  12 Quick Compare
  13 Detailed Comparison of Selected Features
  13 Filesystems and Volume Managers: ext3, ext4, XFS vs. UFS, ZFS
  16 DTrace vs. SystemTap
17 CONCLUSION
INTRODUCTION

There were two primary reasons that IT professionals previously chose the Sun™ SPARC® platform to power their IT infrastructures: the performance of the hardware and the robustness of the Solaris™ operating system. As price, performance, and reliability of industry standard x86_64 servers have increased to the point where they can meet and exceed these features, the reasons to continue buying SPARC hardware have become less and less compelling. Similarly, Linux®, and in particular, Red Hat® Enterprise Linux, has emerged as the operating system of choice to leverage the benefits of an open, industry-standard architecture.

Selecting an operating system for your IT infrastructure has long-term consequences. The selection process must take into account not only the technical features of the current operating system, but the ability for the operating system to enable and support your future business requirements. With the future of Sun and its SPARC architecture in question, Sun’s customers are now realizing that they need to investigate alternative platforms for their enterprise IT workloads. The most widely available alternative to SPARC is the x86-based architecture. Processor designs, chipsets, and systems availability from a broad array of vendors significantly opens up the hardware selection process.

Just as a migration from SPARC means moving away from Sun hardware lock-in for greater choice, moving to a x86-based architecture also opens up the choice of the operating system. Red Hat, in collaboration with numerous partners like Intel, AMD, HP, IBM, Dell, and Sun, develops Red Hat Enterprise Linux to provide the operational flexibility to deploy across a broad range of industry-standard servers.

The enterprise business requirements determine the operational requirements for their IT infrastructure, which, in turn, defines the requirements regarding reliability, availability, and supportability. Many of these operational requirements translate into technical feature requirements of available products and drive the decision of which combination of products provides the best fit for the needs of present and the foreseeable future. Typically, the solution that fulfills these business requirements consists of multiple components: hardware, operating system, middleware, and applications. All of these components must work seamlessly together. In the case of Sun, one needs to consider the commercial viability of the vendor, in addition to technical feature sets.

Ten years ago, the early adopters realized significant benefits in starting the migration from Solaris and SPARC to Linux and x86. Today, the rest of the information technology community is building on the success of the early adopters and accelerating the migration to Red Hat Enterprise Linux. This whitepaper will review some key features and benefits of both Solaris and Red Hat Enterprise Linux. We will also explain why Red Hat Enterprise Linux is an ideal solution for enterprises that are facing Sun and SPARC’s uncertain future, whether you are considering launching a new IT production environment or migrating a legacy environment to a next-generation platform.

FACTORS THAT INFLUENCE OPERATING SYSTEM CHOICE

Aside from the concern that your current platform vendor may not be in business next year, there are more typical reasons for selecting a new operating system. Common reasons include new projects in your enterprise or an event that drives a mandatory migration away from an existing platform. For example, you may be faced with the end-of-life of platform hardware or software, or concerns about general platform longterm viability and serviceability.
BUSINESS REQUIREMENTS

Business requirements for an IT infrastructure are all about the applications. Businesses depend on their applications for themselves and their customers. The applications’ performance and availability, in turn, depend on the platform on which they are deployed. Other business requirements, such as performance (often defined as response time, reliability, and stability), availability, scalability, security, and total cost of ownership (TCO) are also of high priority in decision-making. More abstract requirements, like openness, reputation of the vendor, and the probability of the vendor ending product support, are often not formally specified, but play an important role in making long-term platform decisions.

Availability of applications

The availability of an application on an operating system is a primary consideration. Having your application readily available on Red Hat Enterprise Linux and/or Solaris x86, is the best situation.

Strength of ISV support

The support of a large number of independent software vendors (ISVs) is a good indication of a vibrant ecosystem around a platform. This ecosystem of software and services suggests both the long-term viability of the platform as well as the presence of network effects (the additional benefit to each user when a critical mass of users has adopted the platform). Solaris/SPARC was a popular platform and enjoyed strong ecosystem support, while Solaris on x86 has been slow to garner a similar level of support. In contrast, Red Hat Enterprise Linux has established itself with not only the large and vibrant community of open source developers, but also a rapidly growing number of ISVs in its broad ecosystem. These ISVs represent not only niche applications in vertical markets, but also broad-based applications that the companies build their foundations on, such as databases, business intelligence (BI), and enterprise resource planning (ERP).

The strength of support expressed by developers is just as important. Application developers often have a preferred platform on which they write and test the code, while reserving other platforms as cross-compilation or porting targets. This preference can have a meaningful effect on software quality, with support expertise favoring the preferred developer platform. By virtue of its low cost and pervasive hardware support, Red Hat Enterprise Linux is often the preferred development platform, while Solaris on x86 is widely regarded by developers as a secondary platform for porting. Most applications that run on Solaris 10 on x86 today were actually developed on another platform, such as

New Projects

IT organizations that decide they need to deploy a totally new application get the opportunity to decide whether to stay with the existing application delivery platform (hardware and operating system) or switch to something new. In choosing to adopt a new technology, you should always consider that while the selected technology may be new, the personnel will be the same. The implication being that, choosing a new technology for application delivery that is radically different from your existing environment could require extensive retraining of your existing staff or hiring new staff members who are experienced with the technology.

Mandated Migration

A mandatory migration of an application workload from one hardware and/or operating system platform to another can be triggered by many different factors. Typical triggers could include:

- Existing hardware that is no longer supported.
- The upgrade to the newer hardware of the same architecture is considered too expensive or just as complex as a migration to another platform; therefore, the choice of operating system while keeping the applications is open again.
Solaris 10 on SPARC or even Linux on x86, and then ported to Solaris 10 on x86.

APPLICATION MIGRATION CONSIDERATIONS

The decision to migrate from a SPARC-based server platform to one based on the x86 architecture requires not only switching to a different operating system, but also a different version of the application or, in many cases, a port of an in-house application to the new operating system.

For that reason, the choice of which operating system to deploy on x86 should also take into consideration the cost and effort required to migrate such an application to the new target operating system. To minimize the time and effort needed for such a migration, the newly proposed operating system and build environment should be as similar as possible to the original SPARC environment. Given this requirement for ease of migration, we can eliminate all versions of Microsoft® Windows® from these discussions. While it is technically possible to port an application from Solaris SPARC to Windows, the costs included in deploying such a radically different software environment can't be justified when there are easier and more affordable alternatives available.

Solaris 10 on x86-based platforms would appear to be an ideal alternative to SPARC-based platforms for migration of applications, but because of its limited support on non-Sun x86 platforms and lack of ISV support, Solaris on x86 can't be considered as a serious longterm alternative to Solaris on SPARC.

Sun has attempted to remedy these shortcomings somewhat by enabling the user to run Red Hat Enterprise Linux binaries on top of Solaris x86. Unfortunately, the Solaris Containers for Linux Applications (SCLA) feature lacks support for Red Hat Enterprise Linux 5.1

Since Solaris is considered to be the flavor of UNIX® closest to Linux, porting applications from Solaris to Red Hat Enterprise Linux on the x86 architecture in many cases requires only minor changes to the source and high-level changes to the build environment (makefiles, directory paths, compiler, and linking switches).

This ease of migration from Solaris on SPARC to Red Hat Enterprise Linux on x86, combined with Red Hat's breadth of independent hardware vendor (IHV) and ISV support, make Red Hat Enterprise Linux the ideal migration target platform for homegrown and commercial applications.


- Existing hardware is no longer sufficient. The upgrade to the larger configuration is considered too expensive or just as complex as a migration to another platform; therefore, the choice of operating system while keeping the applications is open again.

- Dissatisfaction with the existing environment without any satisfactory guidance by the vendor.

- Application lifecycle. The current application workload may be at the end of its commercial life or may not take advantage of the latest hardware if upgraded.

In each of these scenarios, a balance between business requirements and technical features of the chosen components must be found.

In any workload migration situation, the initial goal is almost always to re-host the existing application on the new platform. In cases where the desired application isn't available on the new operating system/hardware combination, the choice of a new operating system is heavily influenced by other factors, such as ease of application and data migration to the newly proposed platform.
PERFORMANCE

Making the decision to switch from Solaris on SPARC to servers running Red Hat Enterprise Linux on x86 doesn’t mean accepting lower performance or scalability. Proof of this can be found on Red Hat’s website, where you can read the success stories of customers that have made the switch and have seen great performance improvements for their applications. Red Hat Enterprise Linux not only provides good out-of-the-box performance on x86, but also enables even greater performance via customization.

Red Hat Enterprise Linux also provides the tools needed for performance monitoring and tuning as part of the operating system distribution. While it is hard to replace the actual performance data gathered by running your own application on your hardware, proof of Red Hat Enterprise Linux’s ability to deliver great performance can also be found by reviewing its rank in various industry-standard benchmarks. More details on these benchmarks are found later in this paper.

NOTE: Although there are many Solaris vs. Red Hat Enterprise Linux performance comparisons on blogs, websites, and other opinion-dense forums, these accounts usually fail to provide substantiated evidence. In reviewing these statements, verify that the Linux kernel tested is version 2.6 and that version of Solaris tested is version 10, as there have been significant gains in raw performance since version 2.4 of the Linux kernel and version 9 of the Solaris operating system, and that the hardware is identically configured across tests.

CUSTOMIZATION

The fitness of an operating system for a particular purpose can be significantly improved by creating a customized version of that operating system that has been optimized for your production environment. By virtue of its modularity and openness, Red Hat Enterprise Linux is particularly well-suited for such custom builds. Not only is the platform configurable into server, desktop, and appliance roles with the different installation and compilation options, but you can enhance application performance by minimizing the number of modules loaded into the kernel, as well.

OpenSolaris™, Sun’s open source version of Solaris, is also open to customization. However, the Solaris kernel is not designed with the same degree of modularity as Linux, affording less flexibility in optimizing it for specific workloads or applications. To illustrate this further, consider the wide-ranging use of custom Linux builds in devices and appliances, from smart phones to supercomputers, where each build is optimized for a particular use case. Red Hat Enterprise Linux is supported on systems from the low-power Atom® processor to the latest Intel and AMD processor on the x86 architecture. The devices for which OpenSolaris has been adapted are much fewer in number.

Additionally, the support of OpenSolaris custom builds is dependent upon specialized support agreements with Sun. And while Solaris has been fully supported in production for years with limited possibilities to customize, Sun has only recently begun offering full support plans for OpenSolaris in production environments.

TOOLS

The release of Solaris 10 represented a significant update to the operating system, adding new features to many aspects of the platform. One of the significant new features was DTrace, a dynamic tracing framework for troubleshooting systemic problems in real-time on production systems. DTrace gives system
administrators and programmers the ability to follow a thread of execution from the application down to the operating system and back up again. DTrace makes it easy to find bottlenecks in application code, as well as environmental configurations while the application is in production, lowering the effort of improving its performance. Because DTrace is provided with the system, and the Solaris kernel includes a number of predefined markers or probe points, with many of the predefined probes and scripts, Solaris can provide immediate and easy visibility to any process or library across kernel and user boundaries.

DTrace has been instrumental in finding bottlenecks and driving performance gains in a wide range of applications, from financial to Web 2.0.

The open source alternative to DTrace in the Linux environment is SystemTap, a project that started with code contributions from IBM, Red Hat, Intel, Oracle, North Carolina State University, Stanford University, and others. Intended to provide system tracing and performance monitoring capabilities comparable to those in DTrace, SystemTap uses kprobe technology and extends it from the kernel into user space via the utrace kernel API. Like DTrace, SystemTap provides a library of “probe handlers” to allow for end-to-end tuning, without requiring kernel programming and probe skills from the user. Unlike DTrace, SystemTap allows advanced users to probe any data structure or function inside the kernel and provides for advanced language constructs such as conditionals and loops. Recent releases of SystemTap have been extended to allow support for DTrace markers in the userspace applications.

An in-depth comparison of DTrace and SystemTap is available later in this whitepaper.

BENCHMARKS

While Solaris 10 holds many performance records on industry-standard benchmarks on the SPARC platform, Solaris 10 on the x86 platform has a formidable competitor in Red Hat Enterprise Linux.

For example, consider the SPECweb2005 benchmark for evaluating the performance of web servers using three distinct workloads for banking, e-commerce, and support.

- The highest score of 71,045 was achieved on a Red Hat Enterprise Linux 5.3 (2.6.18-128.el5) system using Rock Web Server v1.4.7 (x86_64), Rock JSP/Servlet Container v1.3.2 (x86_64), in April 2009.
- As of April 2009, the top ten published SPECweb2005 results are based on Red Hat Enterprise Linux.
- Solaris 10 for the x86 platform is noticeably absent from these benchmarks.

For further proof of the ability of Red Hat Enterprise Linux to enable high-performance transaction processing, you need only to look at the TPC-C results. In August of 2008, Red Hat Enterprise Linux was used to host a DB2 database and achieved 1.2 million transactions per minute on the TPC-C benchmark at a cost of $1.99/tpmC. This was the first tpmC result on an x86 server that exceeded the one million tpmC result.

The TPC-H benchmark is used to measure a platform’s ability to host decision support systems that examine large volumes of data, execute queries with a high degree of complexity, and provide critical business intelligence.

For the 300GB version of the TPC-H benchmark, the top ten results are dominated by Red Hat Enterprise Linux. Solaris does not make the list.

5 See also press.redhat.com/2009/04/16/performance-benchmarks/
Red Hat Enterprise Linux has taken the number one position in a variety of enterprise application benchmarks as well. In November 2008, Red Hat Enterprise Linux achieved a record-setting score on the 2-Tier SAP SD benchmark by showing it could sustain a processing load of 5156 concurrent user connections and in turn delivering the best 24-core performance on x86_64 servers and beating Solaris x86 by more than 10 percent on comparable hardware.

Red Hat Enterprise Linux demonstrated that it is also an ideal platform for hosting Java-based applications. In February 2009, Red Hat Enterprise Linux running on a 96-core Xeon®-based server from NEC achieved the best SPEC JavaBlackBelt (JBB) score on a x86_64 server with 2,150,260 business operations per second (bops).

**Proof of Concept**

It is highly recommended that you run your own benchmarks, using your own combination of operating system, middleware, and application(s) on your preferred hardware, to get the best understanding of the performance you can expect. Often, benchmark configurations might offer the best performance, but not necessarily the best stability for your environment. To help you find your ideal balance, make sure that when you’re running a proof a concept in-house, use the try-and-buy option available by nearly all software vendors. In case of the operating systems discussed here, both vendors, Red Hat and Sun, have such offers.

**Bottom Line: Performance**

Red Hat Enterprise Linux has gained rapid and widespread adoption in many industries, due in part to the many success stories verifying its price-to-performance gains over legacy platforms. The ability to optimize Red Hat Enterprise Linux deployment for specific one-off applications, such as those popular in the financial services industry, has been critical to this success. Red Hat Enterprise Linux platforms have shown both best-in-class and overall leadership results across several performance benchmarks.

Given the prevalence and high ranking results of Red Hat Enterprise Linux in industry-standard benchmarks, and the notable absence of or, in some cases, the poor results delivered by Solaris x86 in these same benchmarks, Red Hat Enterprise Linux is the obvious choice over Solaris for hosting an x86 workload.

**RELIABILITY AND STABILITY**

Having established that your chosen applications can run on Red Hat Enterprise Linux or Solaris and deliver the performance you need, your attention is likely to rest immediately on the important issue of robustness. Operating systems have a reputation for robustness that is based on their perceived reliability and stability. When unpacked from its subjective shell, reliability is often measured in terms of the mean time between failure (MTBF) and the mean time to repair (MTTR), which factor into an overall percentage of uptime. Measuring and comparing the reliability of platforms can be difficult, because hardware choice and environmental conditions affect the results drastically. Stability is often characterized by the rate at which defects are found and fixed in the system, which can be equally difficult to track.

Most surveys by industry analysts show that CIOs, IT managers, and system administrators generally consider Red Hat Enterprise Linux to deliver the reliability needed for business-critical workloads and an evolution behind UNIX (HP-UX, AIX, and Solaris) for mission-critical applications. Industry-standard hardware running Red Hat Enterprise Linux has reached a level of maturity where you can configure fault-tolerant systems that are strong enough to match UNIX systems on proprietary hardware.

Red Hat Enterprise Linux kernel and other core operating system components have a well-deserved reputation for running months, even years at a time, without crashing, freezing, or needing to be rebooted.
AVAILABILITY AND SCALABILITY

In most mission-critical environments, systems are typically configured in clusters with redundancy and failover between nodes. Such high-availability clusters are closely associated with the operating system and are sometimes extensions of the platform.

The other motivation to use clustering is the need to provide single systems with significantly better performance. Scale up, using bigger boxes with more processors, and scale out, using more boxes in a single image cluster, are the two standard solutions to this challenge. In many cases, however, third-party applications supplement or supplant the operating system clustering features to provide a large, enterprise-grade solution.

Sun offers, with SolarisCluster™, a high-availability cluster product that does not include a cluster filesystem. If needed, a cluster filesystem can be purchased from Veritas™ and will be supported by both Sun and Veritas. Supported SunCluster configurations are predominantly found on hardware from Sun. Support when using hardware from other vendors is extremely limited. Scalability clustering is not available for Sun on the operating system level. Selected applications, like Oracle, offer scalable clustering on the application level. Sun offers scale up solutions only for its SPARC hardware platform.

Historically, Red Hat Enterprise Linux has enjoyed a strong reputation for scaling out well to large clusters of networked servers. In addition to scientific clustering packages like Beowulf (developed at NASA and maintained as a commercial open source product by Penguin Computing), some of the most popular options for high-availability and scalability-oriented clusters for commercial workloads are included in Red Hat Cluster Suite. Cluster Suite is built into the Red Hat Enterprise Linux Advanced Platform solution and includes Red Hat Global File System (GFS).

Cluster Suite provides the foundation for building high availability clusters by managing cluster quorum, service liveness checks, and service availability. Cluster Suite is built on the foundation provided by OpenAIS toolset, which provides the critical infrastructure for the Service Availability Forum (SAF) APIs. In addition to the cluster foundation, Cluster Suite includes resource agents for common infrastructure applications like Oracle and SAP databases, SAP systems, Apache webserver, and provides an easily extensible frameworks for integrating any application into a highly available configuration.

Building on the Cluster suite foundation, GFS is an open source, POSIX-compliant cluster filesystem that runs on Red Hat Enterprise Linux servers attached to a storage area network (SAN). It works on all major server and storage platforms supported by Red Hat. One of the most popular cluster filesystems for Linux, GFS allows Red Hat Enterprise Linux servers to simultaneously read and write to a single shared filesystem on the SAN, achieving high performance and reducing the complexity and overhead of managing redundant data copies. Red Hat GFS has no single point of failure, is incrementally scalable to tens of Red Hat Enterprise Linux servers, and works with all standard Linux applications.

SECURITY

Operating systems that cannot isolate security attacks and limit damage are no longer viable for business-critical use. Traditionally, operating system security has been a military-grade solution with high costs in the areas of usability and performance, in addition to the cost of the implementation itself. Exact definitions of different security levels can be found online at the Common Criteria portal.  

6 www.commoncriteriaportal.org/
The Common Criteria for Information Technology Security Evaluation (Common Criteria or CC) and the companion Common Methodology for Information Technology Security Evaluation (CEM) are the technical basis for an international agreement, the Common Criteria Recognition Agreement (CCRA).

The CCRA is an internationally recognized standard used by governments and businesses worldwide to determine the level of security and assurance of IT products. For operating system vendors, Common Criteria certification at Evaluation Assurance Level (EAL) 4 or higher has long been the minimum security assurance level required by most government and military agencies. Historically, the operating system vendors that wanted to supply these customers had developed a separate and trusted variant of their mainstream operating system to achieve this level of certification and offered it at a higher price.

Red Hat Enterprise Linux and Solaris have departed significantly from this tradition, combining military-grade security with open source transparency in a mainstream commercial operating system.

Red Hat Enterprise Linux 5.1 has received certification at EAL4+ on SGI servers, and Red Hat Enterprise Linux 5 has received certification at EAL 4+ on HP and IBM servers. Red Hat Enterprise Linux is the only commercial operating system to receive Common Criteria security certification for the broadest set of protection profiles at the highest level of assurance. In fact, since the EAL 4+ certifications are obtained in combination with specific hardware configuration, Red Hat Enterprise Linux 5 has been the most scrutinized operating system with the number of EAL 4+ certifications.

Solaris 10 Release 11/06 Trusted Extensions has received certification at EAL 4+. Solaris 10 Release 11/06 and Release 03/05 have also received EAL 4+.

Capabilities

Both Red Hat Enterprise Linux and Solaris benefit from the contributions of a diverse open source community around security and have well-deserved reputations for being resistant to attacks and intrusions when configured correctly. Red Hat Enterprise Linux is unique as a general purpose operating system that can address multi-level security (MLS) requirements that are traditionally met by only military-grade trusted operating systems.

The open source development model is frequently credited with strengthening the security features of Linux by virtue of its open review process. Although not all projects enjoy this quality of scrutiny, Linux has received better code and fewer bugs as a consequence. Another result of the active community is a robust set of security mechanisms, cryptographic libraries, and trusted utilities available on and used by Red Hat Enterprise Linux for host, network, and application security.

Red Hat Enterprise Linux offers a full range of access control mechanisms, including Discretionary Access Control (DAC), Role-based Access Control (RBAC), and Mandatory Access Control (MAC). Supplementing the traditional DAC implementation by the kernel, Linux Security Modules (LSM) is a lightweight framework with hooks into the kernel that enable various access control mechanisms to be loaded as kernel modules.

One such module, initiated by the US National Security Agency (NSA) and developed together with Red Hat, is Security-Enhanced Linux (SELinux). SELinux implements a flexible MAC mechanism called type enforcement, which associates each subject and object with a type identifier and allows rules governing type-based access to be defined in a policy file loaded into the kernel at boot time. Because the policy is not hard-coded in the kernel, SELinux provides strong mandatory security in a form that system administrators can adapt to a wide variety of security goals reliably and flexibly. Red Hat Enterprise Linux enables SELinux by default with a MAC policy that provides containment around network-facing processes. An administrator can deploy a more fine-grained multi-level security scheme by loading a customized SELinux policy.
Solaris 10 provides new frameworks for containment (zones), user rights management (roles and authorizations), and process rights management (privileges). Solaris Trusted Extensions, a layered product introduced in Solaris 10 Release 11/06 and since integrated as a part of Solaris 10, extends these frameworks by adding sensitivity labels. Trusted Extensions provide a MAC policy base that implements multi-level security to conform to the Common Criteria Label Security Protection Profile (LSPP). Unlike previous versions of Trusted Solaris™ 8, Trusted Extensions in Solaris 10 are based on Solaris Zones.

Solaris Zones is a soft partitioning technology (unlike hard partitions that provide electrical isolation) that creates multiple, virtual environments within a single instance of the operating system. When used in combination with another Solaris 10 feature, Solaris resource management, Zones become components of Solaris Containers, which create a virtual environment with fixed resource boundaries around workloads. This effectively protects applications against intrusions and limits the damage to the container in the event of a successful attack.

Yet another feature carried over from Trusted Solaris into Trusted Extensions in Solaris 10 is process rights management. Process rights management enables processes to be restricted at the command, user, role, or system level. Solaris 10 implements process rights management through privileges, which restrict processes to the minimum capabilities that the processes require. This restriction is called the principle of least privilege. On a system that implements least privilege, an intruder who captures a process has only the privileges granted to the process, limiting the power that could be abused.

Because the Solaris kernel was designed to enforce policy based on process privileges, Solaris 10 has a single security policy hard-coded into the kernel. It does not support multiple security policy configurations. Unlike the LSM framework, which allows SELinux to be dynamically loaded into the kernel, no mechanism exists to load alternate policies or disable the privilege policy in Solaris 10. For some cases, this might be considered an inflexible design for policy enforcement.

Bottom Line: Security

Both Red Hat Enterprise Linux and Solaris offer security capabilities that can protect business applications from a wide range of intrusions and attacks, with Solaris perceived as the benchmark against which to measure. But, Solaris’ Trusted Extensions policy can be considered inflexible because it is hard-coded in the kernel, while Red Hat Enterprise Linux offers considerable flexibility by supporting user-defined SELinux security policies, although defining a security policy can be a complex task.

TOTAL COST OF OWNERSHIP

Total cost of ownership (TCO) is a highly complex amalgamation of multiple costs. It includes the initial purchase costs, the deployment costs, the production costs, and the decommissioning cost. The TCO savings a customer can expect to achieve by switching from a legacy SPARC platform to an x86 platform comes primarily from the savings associated with hardware acquisition, power and cooling costs, conditioning, floor space, administrators, and service and support contracts. However, the number of features that are supported as part of the Red Hat Enterprise Linux subscription versus the Solaris x86 subscription give a cost advantage to Red Hat Enterprise Linux. The total cost of integrating a new Windows platform into an existing Solaris or UNIX infrastructure, plus developing or hiring the skills to support that platform, make the option of switching to a Windows server environment prohibitively more expensive than switching to Red Hat Enterprise Linux on x86.
# FEATURE COMPARISON

## QUICK COMPARE

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>RED HAT ENTERPRISE LINUX 5</th>
<th>SOLARIS 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware platforms supported</td>
<td>• x86: 32-bit and 64-bit&lt;br&gt;• Itanium2&lt;br&gt;• POWER&lt;br&gt;• IBM System z</td>
<td>• SPARC&lt;br&gt;• x86: 32-bit and 64-bit</td>
</tr>
<tr>
<td>Storage support</td>
<td>• Support of 6,347 configurations for connectivity to EMC Symmetrix storage.&lt;br&gt;• Support of 10,534 configurations for connectivity to EMC CLARiiON storage.</td>
<td>Support of only 32 configurations for x86-based servers from other hardware vendors than Sun to connect to EMC Symmetrix and CLARiiON.</td>
</tr>
<tr>
<td>Development model</td>
<td>Red Hat Enterprise Linux is an open source operating system, using the same source tree for all supported platforms.</td>
<td>• Solaris 10 is a single-source operating system, optimized to run multiple applications on multiple platforms.&lt;br&gt;• OpenSolaris is an open source operating system based widely in part on Solaris 10. Not all features of Solaris 10 are available in OpenSolaris. While OpenSolaris has an active community, until now very few contributions from the community have made it back into Solaris 10.</td>
</tr>
<tr>
<td>High-availability clustering</td>
<td>The Red Hat Enterprise Linux 5 Advanced Platform subscription includes support for Red Hat Cluster Suite, which enables users to create multi-node high-availability clusters for maximum up time of commercial and internally-developed applications.</td>
<td>Sun ships SolarisCluster high-availability software with Solaris 10, but requires an additional annual subscription payment for support. SolarisCluster software is certified on most Sun hardware configurations, but less than 20 non-Sun hardware configurations.</td>
</tr>
<tr>
<td>Cluster filesystem</td>
<td>Included in Red Hat Cluster Suite is the open-source cluster filesystem Red Hat Global File System (GFS). GFS allows systems to simultaneously read and write to a single shared filesystem on the SAN, achieving high performance and reducing the complexity and overhead of managing redundant data copies. It has no single point of failure, is incrementally scalable from 16 to 32 Red Hat Enterprise Linux servers, and works with all standard Linux applications. It scales to file system sizes of up to 8EB.</td>
<td>• None available for Solaris from Sun, but a cluster filesystem can be purchased from Veritas.&lt;br&gt;• Sun’s clustered filesystem product, Lustre is only available for Linux and targeted for High Performance Compute (HPC) clusters rather than HA clusters.</td>
</tr>
</tbody>
</table>
Red Hat Enterprise Linux 5: Your Solaris Alternative

Some features in the quick comparison are more interesting than others. The following discusses those considered more interesting for the purposes of this whitepaper.

**Filesystems and Volume Managers: ext3, ext4, XFS vs. UFS, ZFS**

All enterprise operating systems ship with a default filesystem that the vendor is willing to support and typically a next generation or alternative filesystem as well, so customers can have early access and input into filesystem features and functions that are under development or one that provides different performance characteristics. Occasionally these alternative filesystem technologies prove to be compelling enough to the user community that the vendor decides to actually offer support plans for them and promote them to the level of default filesystem for the operating system.

To a large extent, the success or failure of an operating system is dependent on the filesystem that it deploys with. The operating system depends on the file system to maintain data security, integrity, availability, capacity, and expandability, while delivering the highest possible throughput possible. In addition to those requirements, the management features and interfaces to the filesystem need to be as intuitive and as logical as possible for the system administrator in order to maximize his efficiency in performing routine filesystem management tasks.

As such, they are a keystone regarding security and administrative comfort for each operating system.

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>RED HAT ENTERPRISE LINUX 5</th>
<th>SOLARIS 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next Generation filesystems</td>
<td>ext4 is a modern journaling filesystem that allows for an unlimited number of subdirectories, a maximum filesystem size of 1EB with maximum file size of 16TB, extents for mapping sequences of blocks together with multi-block allocation, and delayed allocation to optimize allocation and reduce overhead.</td>
<td>ZFS is an attempt to reinvent the filesystem by including features that are usually part of storage management software. Its main advantage is its scalability up to 256 quadrillion zettabytes. Still a newcomer to the filesystem market, there is little evidence of ZFS in production environments. ZFS is also embroiled in a patent dispute with NetApp, which may further hamper its widespread adoption.</td>
</tr>
<tr>
<td>Dynamic tracing</td>
<td>SystemTap is a joint effort of Red Hat, IBM, Intel, Hitachi, and Oracle, and can be used in production systems. It is implemented as a pre-compiled module and offers full control structures to access and collect data while tracing.</td>
<td>DTrace is a dynamic tracing facility that can be used in production systems and is implemented as part of the kernel. It allows for the tracing of compiled programs as well as many scripted language programs, including Java.</td>
</tr>
<tr>
<td>Security certification</td>
<td>EAL 4+</td>
<td>EAL 4+</td>
</tr>
</tbody>
</table>

**Detailed Comparison of Selected Features**

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>RED HAT ENTERPRISE LINUX 5</th>
<th>SOLARIS 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next Generation filesystems</td>
<td>ext4 is a modern journaling filesystem that allows for an unlimited number of subdirectories, a maximum filesystem size of 1EB with maximum file size of 16TB, extents for mapping sequences of blocks together with multi-block allocation, and delayed allocation to optimize allocation and reduce overhead.</td>
<td>ZFS is an attempt to reinvent the filesystem by including features that are usually part of storage management software. Its main advantage is its scalability up to 256 quadrillion zettabytes. Still a newcomer to the filesystem market, there is little evidence of ZFS in production environments. ZFS is also embroiled in a patent dispute with NetApp, which may further hamper its widespread adoption.</td>
</tr>
<tr>
<td>Dynamic tracing</td>
<td>SystemTap is a joint effort of Red Hat, IBM, Intel, Hitachi, and Oracle, and can be used in production systems. It is implemented as a pre-compiled module and offers full control structures to access and collect data while tracing.</td>
<td>DTrace is a dynamic tracing facility that can be used in production systems and is implemented as part of the kernel. It allows for the tracing of compiled programs as well as many scripted language programs, including Java.</td>
</tr>
<tr>
<td>Security certification</td>
<td>EAL 4+</td>
<td>EAL 4+</td>
</tr>
</tbody>
</table>
The default filesystem delivered with Red Hat Enterprise Linux is ext3, which added journaling capabilities to ext2. Two alternative filesystems available to Red Hat Enterprise Linux users are ext4 and XFS. Ext4 (the next generation of ext3) improves ext3’s fsck performance on large files as well as provides better delete performance by reorganizing some of the on-disk semantics of data storage. Currently available as a Technology Preview, the ext4 filesystem is on track to becoming a fully supported component of the Red Hat Enterprise Linux 5 and eventually the default filesystem in Red Hat Enterprise Linux 6. An alternate filesystem, XFS, is also available under beta program for Red Hat customers. XFS has proven to be very popular with users looking for fast performance when streaming or deleting large data files. Red Hat plans to move XFS from beta to full support.

As filesystems can only support the underlying storage that is being presented to them, there is also a need to reduce the number of filesystems on a server. Red Hat Enterprise Linux provides this functionality with the Logical Volume Manager (LVM). LVM provides the capability to aggregate underlying storage, whether from local disks or remote LUNs, and create volumes that remove disk and LUN boundaries. When deployed in conjunction with Cluster Suite, LVM provides a complete clustered volume manager with online resizing capabilities under GFS.

Solaris has never included a full-functioned volume manager, and the default filesystem for Solaris for many years, UFS (UNIX file system), depended on third-party tools for this functionality. In an attempt to support much larger filesystems, increase data integrity, and reduce the customer’s dependency on third-party filesystem management tools, Sun introduced the ZFS (Zetabyte File System) as an alternative to UFS in early versions of Solaris 10.

Ext3, XFS, and ext4 have evolved from classic filesystems. The intention behind ZFS was to overcome the restrictions of classic filesystems without losing compatibility. It integrates storage management features into the filesystem. It is up to discussion if this integration makes sense or if the classic separation of storage management and filesystem offers better control and less confusion for administrators. Classic IT environments with separate groups being responsible for hardware and system administration will, in most cases, prefer a separation between storage management as part of hardware administration and filesystems as part of system administration, to help keep responsibilities clearly defined and access rights as simple as possible. Small-scale IT environments where hardware administration and system administration are not separate functions might see some advantages to the integrated approach. However, such environments often have budget constraints that will not allow them to take advantage of all those features, due to constraints of the underlying hardware.

Due to the major conceptual changes in ZFS as compared to traditional filesystems, Sun published its own Solaris ZFS Administration Guide, even with ZFS being “designed to be robust, scalable, and simple to administer.”

New features and concepts of ZFS, and their counterparts in traditional filesystems, are:

- Pooled storage: Storage pooling allows a ZFS filesystem direct access to multiple disks without the additional control of a volume manager. Additionally, it allows for dynamic growing and shrinking of filesystem size when multiple filesystems share the same storage pool. To restrict filesystem size, explicit quotas have to be introduced.

- Traditional filesystems, like ext3, ext4 and XFS, offer the capability to grow as long as the underlying volume set offers enough space. This gives significantly better control over the use of storage space.
Transactional semantics: Each change of the content of a ZFS filesystem is treated like a transaction. This is managed by copy-on-write semantics and provides consistent filesystems without corruptions. Traditional filesystems offer multiple approaches to achieve the same level of consistency. On the lowest level, fsck corrects corruptions after they’ve occurred. On higher levels, journaling is offered either with the log or journaling section included in the data section, or with its own log or journaling section. In both cases, all changes to the content of the filesystem are logged and, in case of inconsistent data, it is possible to get data back to the last consistent state. The separation of data and the log or journal in the filesystem ensures that the journal stays consistent, since metadata is only added to the journal, while use data can be overwritten as needed. Transactional semantics are the more elegant solution, but traditional filesystems offer more granularity regarding the need of consistency.

Checksum and self healing: ZFS offers checksumming and data recovery on the filesystem layer, which allows for transparency of the checksumming and data recovery to applications. Traditional filesystems working on top of volume sets will typically use checksums on blocks of data. Journal checksumming, together with nanosecond timestamps as are introduced in ext4, allow for a similar effect as checksumming on the filesystem layer.

Self-healing data is provided by ZFS by using varying levels of data redundancy, including mirroring and a variation on RAID-5. In traditional setups, this is part of the storage management level that provides volume sets to the filesystem.

Scalability: At this time, ZFS is more scalable than ext3, ext4, or XFS, allowing for up to 256 quadrillion zetabytes of storage since ZFS is 128-bit. It should be mentioned that, until now, there was no installation of such a large storage example to really test the scalability. Neither XFS and ext4 reach into those regions yet, but have been tested to their limits, which is 1EB for ext4. Rather than looking at theoretical limits, the filesystem size recommendation for most operational environments are limited based on the operations team’s recovery window.

ZFS snapshots: ZFS offers snapshots as a read-only copy of the filesystem. This is traditionally part of storage management. Today, storage management tools offer much more sophisticated levels of snapshots than ZFS in the hardware, and LVM offers simpler read-only copy and copy-on-write snapshots at the volume management layer.

Simplified administration: It is up for discussion if integrating storage management functions into the filesystem really simplifies administration. It allows the use of one command with more options, rather than using multiple commands with simpler options. It also removes the conceptual border between storage and filesystem management.

Additional to the features already mentioned in the comparison with ZFS, there are other new features of ext4. These include the resolution of the 32,000 subdirectory limit (an unlimited number of subdirectories is now supported), a maximum filesystem size of 1EB with a maximum file size of 16TB, extents for mapping sequences of blocks together with multiblock allocation, and delayed allocation to optimize allocation and reduce overhead. The use of extents with multiblock and delayed allocation, as well as the use of persistent preallocation, reduces the need for online defragmentation, which is planned for the next release of ext4.

Bottom line: Filesystems

While ZFS is a new take on the concept of filesystems, most the new features available in ZFS can be matched with the existing ext3 filesystem available in Red Hat Enterprise Linux or in the near future with ext4 or XFS, with the exception of scalability. Even with these new features, it is important to remember that
ZFS is primarily a Solaris-only component (ZFS will also be available on Mac OS X server version). If a user were to choose to deploy Solaris on an x86 platform as he transitions off of SPARC, with the explicit goal of utilizing ZFS, he would be severely limiting his choice of Solaris supported hardware platforms to deploy on.

Given ZFS’s brief history of commercial availability (less than five years), it’s understandable to see why most current Sun customers are taking a “wait and see” attitude and sticking with the more traditional UFS filesystem for their Solaris deployments. The fact is that many existing Solaris users and system administrators have yet to use or even be trained on ZFS.

For those users and system administrators that are planning on moving off of the Solaris/SPARC platform, minimizing the amount of change and being able to leverage the skills on hand are critical factors for a smooth transition. While some of these system administrators may see a transition from Solaris/SPARC to Solaris x86 as an opportunity to introduce ZFS into their infrastructure, the need for training on ZFS management would definitely slow down this transition and increase the expense.

Since many of the operating system features and functions in Solaris (including UFS filesystem management) map easily to those found in Red Hat Enterprise Linux, existing Solaris system administrators would find the transition to ext3 filesystem management on Red Hat Enterprise Linux a very straightforward and intuitive process.

While ZFS does have compelling features to offer, the limited availability of supported deployment platforms and its unproven track record in production environments quickly eliminate it as a compelling reason to select Solaris x86 over Red Hat Enterprise Linux in the transition from Solaris SPARC.

DTrace vs. SystemTap

SystemTap and DTrace are dynamic tracing tools that support debugging and profiling. Neither should be used indiscriminately in a production environment, but they can be used there as a superior monitoring tool. As it is a truism that the process of measuring will change the measured data, it is also true that the process of monitoring will change the monitored data. This should be kept in mind when enabling either tool.

DTrace was introduced in the Solaris Express 11/03 release and significantly enhanced in the Solaris Express 5/04 and 11/04 releases. It is marketed as “a comprehensive dynamic tracing facility that gives Solaris users, administrators, and developers a new level of observability into the kernel and user processes.”

Since DTrace is a feature of the Solaris kernel, it is covered under the same open source license, known as CDDL. Even though attempts are being made to port DTrace to Linux, the differences between GPL and CDDL licensing appear to prohibit the introduction of DTrace code into the Linux code base. In true open source fashion, Red Hat, IBM, Intel, Hitachi, Oracle, and others contributed code and expertise to implement SystemTap, which brings similar functionality to Linux without requiring extensive instrumentation.

Both tools share the same intent—supporting the user in optimization efforts by providing dynamic tracing. Both tools also share many ideas, concepts, and features. Most commonalities are based in the shared intent. Therefore, both support instrumentation of code, data collection, and scripting to help interpret the collected data. However, the actual implementation and the level of intrusion is different.

DTrace infrastructure is itself part of the kernel. SystemTap is implemented as a pre-compiled module that is loaded only when specific functionality is required. Both methods allow the respective tool to be enabled and

10 sourceware.org/systemtap/wiki/SystemtapDtraceComparison
disabled in a production system. Creating DTrace as part of the kernel allows for faster execution, but runs a higher risk of interference with other parts of the kernel when tracing code. Creating SystemTap as a pre-compiled module allows for easy addition as well as easy removal.

While both tools use scripting, only SystemTap has support for control structures in its scripting language. While this does not necessarily limit the capabilities of DTrace, it does require a different approach to make conditionals and loops work. Additionally, SystemTap allows for more complex reporting of first principles, iterations, and conditionals. Thread-local variable and speculative tracing are supported by both.

Probe execution is implemented as optimized native code in SystemTap, while DTrace depends on interpreted byte codes. While the interpreted byte code allows for easier interpretation and portability across release, the optimized native code keeps the performance impact low. Due to the nature of Solaris kernel ABI, DTrace has the additional advantage of supporting tracing of Java programs and several script languages. SystemTap depends on a new kernel interface, utrace, to provide similar functionality, tracing userspace applications.

While being developed in just a little bit more than half the time of DTrace, SystemTap has certain features DTrace does not match. These capabilities include:

- Probing arbitrary statements in code symbolically using debugging information; DTrace is limited to the ABI boundaries, like entry and exit markers.
- Symbolically extracting arbitrary data at probe points like any context-visible variable as preserved by the compiler; all data is visible in DTrace, but access to local variables may require the use of manual register offsets.
- End-user extensible probe library as script-based tapsets; DTrace does not offer anything similar.

Additionally, the number of available symbolic probe points in user space and in the kernel are significantly higher in SystemTap.

**Bottom line: Dynamic tracing**

While DTrace is a stable product with continuing development, SystemTap is a stable product with ongoing development to extend the full range of capabilities into the usespace with a large community of developers. This allows SystemTap development to have a faster reaction time to implement new features as they are requested, while DTrace has a limited advantage regarding code stability. Both have single features that the other does not possess: SystemTap focuses on the area of adaptability and easy access to full control structures, while DTrace shines with script and interpreted languages. Classic development environments will realize a major advantage from using SystemTap over DTrace, while teams using heavy scripting in their programming efforts may, at this time, still prefer DTrace.

**CONCLUSION**

As discussed, Red Hat Enterprise Linux and Solaris both provide the robust feature set one would expect to find in an enterprise-class operating system. However, Red Hat Enterprise Linux and its associated subscription offers distinct advantages over Solaris 10 in a server market dominated by the x86 architecture.

Red Hat Enterprise Linux’s advantages over Solaris fall into three categories—platform support, platform viability, and customer value—all of which are critical to users that are faced with the inevitable migration from SPARC to x86.
Platform support.

While Solaris is built to run on 100 percent of Sun's hardware and peripherals, its supported availability on non-Sun x86 hardware is limited. Adopting Solaris 10 for your x86 servers would result in limiting the choice of x86 server vendors and models. The same argument can be made with respect to ISVs. Solaris on SPARC may still have a viable (albeit shrinking) commercial application ecosystem, but Solaris on x86 is not a tier-one platform for any of the top enterprise ISVs. In the x86 server market, the two tier-one operating system platforms targeted by ISVs for their applications are Linux and Windows. The fact that Red Hat Enterprise Linux operations and management interfaces are similar to Solaris, much more so than Windows, makes Red Hat Enterprise Linux the smart choice when considering a move to an alternative operating system for Solaris workloads.

Platform viability.

With the future of Sun Microsystems and its SPARC processor platform in doubt, the future of Solaris itself is an unknown and should be a major concern for anyone still dependent on Solaris today. Regardless of any technical innovations that may arise in Solaris, reliance on a commercial operating system that has an unknown future can't be justified. Even though Sun has created and sponsored an open source version of Solaris called OpenSolaris, to date it has had little commercial adoption and the community is only a fraction of the size of the global Linux community.

A key indicator of platform viability is the frequency with which that platform is used in public benchmarks. In the world of performance benchmarks on the x86 architecture, Red Hat Enterprise Linux is a dominant contender and preferred platform, whereas Solaris 10 is not even a candidate.

While Solaris has enjoyed a dominant position on Sun's own hardware, Red Hat Enterprise Linux offers the advantages of the open source development model, backed by a large worldwide community, along with the professional support structure of a dedicated and trusted vendor.

Customer value.

One of the most valuable aspects of the technology acquisition process is freedom of choice or, rather, freedom from vendor lock-in. Red Hat Enterprise Linux has broad support from server and storage vendors as well as ISVs, which ensures that a customer who chooses Red Hat Enterprise Linux is able to protect and maintain that freedom. Solaris's limited availability and platform support severely curtail this freedom of choice for the customer.

Red Hat Enterprise Linux also packages more value into its distribution than Solaris. While Solaris customers receive and can run SolarisCluster software, they cannot get support for it unless they purchase an extra SolarisCluster subscription and, even then, there are less than 20 configurations of non-Sun hardware that Sun supports for Solaris on x86. Red Hat Enterprise Linux 5 also has the added value of providing virtual machine-based virtualization bundled as part of its distribution. With the Red Hat Enterprise Linux Advanced Platform solution, customers are offered unlimited virtual guests and built-in clustering and file system technologies. Sun has been talking about its xVM server for two years, but has yet to ship, and it will be incompatible with its existing VirtualBox technology once it does finally ship.

The flexibility to support multiple versions of an operating system and upgrade only when the need arises is a valuable operational requirement for any datacenter. The Red Hat Enterprise Linux subscription allows customers to upgrade and downgrade to newer or older versions of Red Hat Enterprise Linux at no additional charge and at any time. In contrast, Sun still charges license fees for Solaris 8 and 9, and has different support service pricing models for those releases.
The cost of staffing a datacenter with certified professionals also has to be taken into account when making a platform choice. With the overall rise in popularity of Linux training, and the Red Hat Certified Engineer® (RHCE®) certification in particular, most IT managers have a pool of certified Red Hat Enterprise Linux professionals to choose from as they build out their business. While Sun does offer Solaris systems administration certification, it is neither a well-known nor highly sought-after certification. The result is a small and shrinking number of individuals that are highly compensated for their Solaris skill set due to their scarcity. Indeed, these trained Solaris professionals are also more expensive than Red Hat Enterprise Linux administrators. In the case of a new environment, this is a significant advantage for Red Hat Enterprise Linux. In summary, Red Hat Enterprise Linux is the better choice for customers desiring increased flexibility, scalability, performance, and ease of use, and reduced TCO, both now and in the long-term.