Marc Staimer, Dragon Slayor Consulting

W H I T E P A P E R

Think All Distros Offer the Best Linux DevSecOps Environment?

Think Again!

Think All Distros Provide the Best Linux DevSecOps Environment? Think Again!

Introduction

DevOps is changing. Developing code with after the fact bolt-on security is dangerously flawed. When that bolt-on fails to correct exploitable code vulnerabilities, it puts the entire organization at risk. Security has been generally an afterthought for many doing DevOps. It was often assumed the IT organization's systemic multiple layers of security measures and appliances would protect any new code from malware or breaches. And besides, developing code with security built in, adds tasks and steps to development and testing time. More tasks and steps delay time-to-market. Multi-tenant clouds have radically changed the market. Any vulnerability in a world with increasing cyber-attacks, can put millions of user's data at risk.

Those legacy DevOps attitudes are unsound. They are potentially quite costly in the current environment. Consider that nearly every developed and most developing countries have enacted laws and regulation protecting personally identifiable information or PII[1]. PII is incredibly valuable to cybercriminals. Stealing PII enables them to commit many cybercrimes including the cybertheft of identities, finances, intellectual property, admin privileges, and much more. PII can also be sold on the web. Those PII laws and regulations are meant to force IT organizations to protect PII. Non-compliance of these laws and regulations often carry punitive financial penalties.

The practice of bolting-on security to DevOps after development should be retired. And yet, it's still the rule versus the exception. Consider that exploitative code vulnerabilities may not be specific to the developed code, rather the underlying operating systems, compilers, or tools utilized in developing the code. DevOps code too frequently slows down development. Deadlines are missed. Too many IT organizations make the cold calculation to leave the security activities to the end of their development cycles. That is becoming increasingly precarious. It only takes one exploited vulnerability to ruin the executives', stockholders', employees' and especially the developers' day. When that happens, pandemonium breaks out. Everyone drops whatever they're doing to fix the situation as rapidly as possible.

The cybercrime problem is getting worse, not better. The rise of ransomware has accelerated this trend. A new organization will fall victim to ransomware every 14 seconds[2]; 1.5 million phishing sites are created every month[2]; and ransomware attacks have increased more than 97 percent in the past two years[2]. It's a rapidly growing problem. Ransomware targets system and software vulnerabilities exploiting them as soon as they're publicly known. They become publicly-known as soon as a vulnerability patch is released.

The moment an OS vulnerability patch is released the clock starts. The onus of whether or not it can be implemented before the cybercrooks can exploit it falls on the IT administrators. This ratchets up the stress and one of the reasons why DevOps has become more security conscious morphing into DevSecOps.

DevSecOps integrates security operations and practices with software as it is developed. The goal is to rapidly deliver high-quality, highly secure products. DevSecOps focus is to ensure security built into the development process from the start. The development team is taking on the responsibility in delivering more secure code and services instead of relying on others to secure their work after completion. Success depends on close alignment between application development with significant improvements in monitoring, alerting, automation, patching, upgrading, and deployment outcomes.

DevSecOps is a very high priority for Oracle and it shows in all Oracle Databases, applications, middleware, and especially the Oracle Linux distribution. This paper demonstrates how Oracle's emphasis on security makes its Linux distribution the leader and most suitable for DevSecOps.

[1] For healthcare it is referred to as private health information (PHI) or electronic PHI (EPHI)
[2] Ransomware Statistics and Comparitech

Table of Contents

Known Linux Security Issues and Oracle Linux Answers

Open source Linux has rapidly become the predominant UNIX derived OS. It now greatly exceeds[3] Microsoft Windows servers deployments. That popularity has made it a target for cybercriminals of all types. Make no mistake, there are definitely Linux exploitable vulnerabilities. The open source community is committed to fixing those vulnerabilities. The problem is that applying Linux patches for those vulnerabilities is a non-trivial process.

The Linux Vulnerability Patching Problem

The reason patching Linux vulnerabilities is non-trivial is because it's normally disruptive. Disruptive processes require scheduling. Few applications can tolerate an outage during business hours. Most IT organizations schedule disruptive processes such as Linux vulnerability patching for a weekend sometime within a 90-120-day timeframe. Verizon 2020 DBIR reports only half of the vulnerabilities are patched within three months after discovery and when the patch is released. In other words, the patching is delayed. It's put off so the different stakeholders, applications, servers, hypervisors, storage, networking, etc., have time to coordinate their efforts. When that scheduled weekend rolls around, (and that's assuming it hasn't been deferred, which is far too common), the first 24 hours is when all disruptive patch processes are implemented. The next 24 hours is reserved to back out the patches that didn't work or caused problems. These processes are labor intensive and error prone.

Per Ponemon Institute and ServiceNow IT cloud services, interviews with more than 3,000 cybersecurity professionals worldwide[4] determined 48%[3] experienced minimally one and possibly more data breaches within a two-year period with 57% surveyed ascribed the breach to vulnerabilities that had a patch that had not been applied.

When a Linux vulnerability patch is released it documents all of the vulnerabilities and security holes that were fixed. This starts the race between those who implement the patch and those attempting to exploit the vulnerability. The patch release documentation informed the cybercriminals where to strike. They know they have a window of opportunity to exploit the vulnerabilities before the patch is implemented.

The cybercriminals reverse-engineer those vulnerabilities in weeks or even days. They assume most IT organizations take months, often many months to apply the patch. The 2019 Verizon Data Breach Investigations Report validates that assumption. They found patching tends to be untimely and generally incomplete. Their research showed IT organizations patch fewer than 40% on average their affected vulnerable systems within 30 days of vulnerability disclosure. Their research confirmed that patching is a disruptive process, requires extensive multi-department coordination, and is manually labor-intensive. The Ponemon Institute and ServiceNow report additionally found:

- 55% spent more time with manual processes than rapidly responding to vulnerabilities.
- 61% felt frustrated about their reliance on manual processes when patching vulnerabilities.
- 12 days on average was lost manually coordinating across teams for every vulnerability patched.
- 65% recounted difficulties triaging which vulnerabilities to patch first and which can wait.

Linux vulnerability patching is a massive security problem that urgently needs correction.

How Oracle Linux Remedies Problematic Linux Vulnerability Patching

Remedying this common demoralizing and problematic Linux security vulnerability patching requires making Linux OS patching "non-disruptive" and online. And that is exactly what Oracle does with Ksplice. Oracle's unique[5] Ksplice technology and service updates the kernels, hypervisors, and critical user space libraries without requiring a reboot or interruption. This means whenever any Linux OS patch is released,

[3] *IDC Worldwide Operating Systems and Subsystems Market Shares report* covering 2017, Linux had 68% of the market. Its share has only increased since then. Microsoft developer reveals Linux is now more used on Azure than Windows Server: https://www.zdnet.com/article/microsoft-developer-reveals-linux-is-now-more-used-on-azure-than-windows-server/

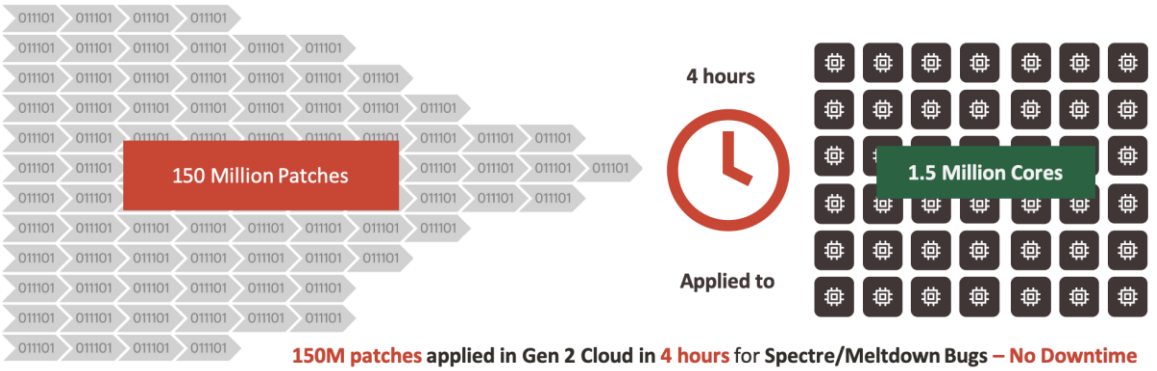
[4] Today's State of Vulnerability Response: Patch Work Demands Attention

[5] Red Hat has Kpatch and SUSE has kGraft which are somewhat similar to Ksplice, but only for the kernel and a small patch subset. Ksplice has a much broader range of patches including the ability to patch hypervisors and critical user space bits non-disruptively.

including patches for exploitable vulnerabilities, Ksplice enables those patches to be implemented quickly in a timely manner, without having to be scheduled or coordinated with anyone. No disruptions, no downtime, no scheduling, no coordination, and no extended windows of known exploitable vulnerabilities. In other words, a more secure Linux.

Oracle Linux Ksplice adds an additional level of security alerting after security patches are implemented with "Known Exploit Detection" on privilege escalations. Admins can narrow the alerts to specific privilege escalation attempts. It exclusively enables auditing and alerting for known privileged escalations.

No other Linux distribution, not IBM Red Hat, not SUSE, nor Ubuntu, solve this security problem. Oracle Linux is the first Linux distribution that truly remedies problematic Linux vulnerability patching.



Oracle Gen2 Cloud Spectre/Meltdown Non-disruptive Patch Proof Point

Another proof point comes from a very large airline utilizing Oracle Linux and Ksplice. Ksplice allowed them to meet stringent security requirements without having to "go through a lengthy change management process. That lengthy change management process commonly adds a ton of operating expenses" to the process. Sysadmin time plummeted from 54 to 7 hours per cycle while patched or updated servers continued to operate without a reboot. Ksplice also enabled the sysadmins to roll back patches or updates from any point in microseconds with a simple command.

What about those that claim hypervisors with Live Migration or vMotion can manage this problem with rolling patches. The short answer is they don't. The rolling patch process patches or upgrades a virtual machine (VM) with the Linux OS and the application, migrates users to the patched VM, moves on to patch the unpatched VM, then fails back the users to a patched VM. Rinse and repeat. This process is still disruptive necessitating scheduled downtime. It's still manual, labor-intensive, repetitive, and subject to frequent human errors. It does not increase the security of DevOps.

Oracle patch and update regression testing are designed to avoid adversely impacting existing environments. However, that does not relieve IT organizations of the responsibility of doing their own regression testing. Best practices still require those tests be run in their test environments before applying the patches to their production environments. Rolling back patches is another consistently error prone task. Not with Ksplice. And as Oracle Linux customers have discovered, Ksplice rolls back patches as easily as it applies them, once again non-disruptively without rebooting the machine.

The DevSecOps Security Ecosystem Deficit Problem

To truly be DevSecOps requires a very secure Linux ecosystem. The standard pushed by industry and governments, is the Common Criteria (CC) per the National Information Assurance Partnership[6] (NIAP) General Purpose Operating System Protection Profile (OSPP). It has become the industry standard.

Another DevSecOps Linux distribution issue is encryption. Sure, everyone has encryption. But their encryption is rarely National Institute of Standards (NIST) FIPS 140-2 validated. Many vendors claim HIPAA HITECH compliance, but have not been validated. FIPS 140-2 is not an encryption algorithm. It's a validation process by NIST third party contractors. This is critically important for the US healthcare and federal

[6] National Information Assurance Partnership

government markets. HIPAA HITECH requires all cryptographic modules must be NIST validated. FIPS 140-2 is the only approved validation process.

Validation is neither non-trivial nor inexpensive. If the encryption stack is not FIPS 140-2 validated, then per HIPAA HITECH[7] the stored electronic private health information (EPHI) is not in compliance. Many US Government agencies, state, and local governments also require the FIPS 140-2 validation for encryption.

The next frequently overlooked DevSecOps Linux security issue is keeping up with the latest hardware and open source security advancements. Security is an ongoing moving target. It's a game of offense versus defense. Cybercrime is a lucrative business raking in billions of US Dollars[8] each and every year with a lot of the profits being poured into R&D. Cybercrime is organized and at times, state sponsored. For example, North Korea cybercrime captures significant amounts of foreign currency for the notorious hermit nation[9].

There is never going to be perfect security. Security best practice is for IT organizations make it as hard as they can for the cybercriminals incenting them to find easier targets. This is why security pros emphasize layered defense in depth. It is the same strategy required for DevSecOps. Being able to leverage the latest hardware security enhancements such as AMD's EPYC Secure Memory Encryption (SME) makes it harder for the cybercriminals. Making sure the Linux distribution is utilizing the latest open source security enhancements such as Kata Containers, libvirt, QEMU, and Kubernetes security functions like OpenID Connect tokens based on OAuth 2.0, makes breaching security that much more difficult.

Awkwardly, nearly all Linux distributions haphazardly support both hardware and more importantly open source security advancements. Just like disruptive patching, it opens up potential vulnerabilities that can be exploited.

How Oracle Linux Addresses the DevSecOps Security Deficit Ecosystem Problem

Oracle is laser focused on security. This has led Oracle to a very proactive approach in adopting and supporting the latest hardware from AMD, Intel, and ARM. Oracle is also steadfastly committed to supporting the latest open source Linux security software including KATA Containers, libvirt, QEMU, and OpenID Connect tokens based on OAuth 2.0. In addition, Oracle provides Secure Boot, data encryption both in-flight, and at-rest.

Oracle goes beyond just supporting open source Linux advancements, it makes them as well. Oracle is an ongoing major contributor to the Linux development open source community. One example of those contributions is the widely acclaimed Oracle Clustered File System 2 (OCFS2). OCFS2 is a general purpose, extent-based clustered file system developed by Oracle and contributed to the Linux community. It's a highly effective open source, enterprise-class alternative to proprietary cluster file systems, providing both high performance and high availability. Another is DTrace. DTrace is a comprehensive dynamic tracing framework providing a powerful infrastructure enabling administrators, developers and service personnel to concisely answer arbitrary questions about the behavior of the Linux OS and user programs in real time. It greatly simplifies root cause analysis, critically important to DevSecOps during testdev to troubleshoot problems quickly and accurately. These are just some of Oracle's contribution. Oracle provides pre-release code on GitHub.

Oracle Linux is FIPS 140-2 validated. And Oracle Linux is now the only Linux distribution on the NIAP Product Compliant List[10].

Known Linux DevSecOps Performance Impact and Oracle Linux Answers

When the underlying Linux OS does the heavy lifting on performance, DevSecOps processes and applications have to do much less. Some may question how is performance optimization a DevSecOps

[7]Electronic PHI ("EPHI") has been encrypted as specified in the HIPAA Security Rule by "the use of an algorithmic process to transform data into a form in which there is a low probability of assigning meaning without use of a confidential process or key" (45 C.F.R. § 164.304, definition of encryption) and if such confidential process or key that might enable decryption has not been breached. To avoid a breach of the confidential process or key, these decryption tools should be stored on a device or at a location separate from the data that they are used to encrypt or decrypt. The encryption process identified below have been tested by NIST and judged to meet this standard.
Valid encryption processes for data at rest are consistent with NIST Special Publication 800-111, "Guide to Storage Encryption Technologies for End User Devices."
[8] Ransomware Statistics and Comparitech
[9] How Cybercrime Funds North Korea's Nuclear Program
[10] Oracle Linux certified under Common Criteria and FIPS 140-2

security issue? Security and performance go hand-in-hand. Security processes are frequently resource intensive. Greater performance enables more of that performance to be allocated to built-in security without reducing acceptable application performance. Oracle's 40-year experience as the world's dominant database has given Oracle insider knowledge and a unique perspective on how to optimize security and performance of critical workloads.

The Oracle Database is a fundamental underlying software infrastructure. Its performance affects and is itself affected by security. Making the database more efficient and performance optimized enables more security functions without impacting database application performance. The Oracle Linux OS is also a fundamental underlying software infrastructure. Oracle is applying that production proven 40-year experience to the Linux performance landscape.

Linux DevSecOps Performance Optimization Limitations

Red Hat, SUSE, and Ubuntu Linux distributions do not have Oracle's decades of database performance and security optimization experience. They cannot leverage knowledge they do not have. That means they leave it up to the application developers to optimize for performance. That's more work for the developers. This frequently slows down DevSecOps. And as stated previously, security often negatively affects application performance, which in turn further slows DevSecOps completion time. All Linux distributions take advantage of open source performance enhancements. Few distros emphasize application/database performance optimization.

How Oracle Linux Optimizes Performance for DevSecOps

Oracle specifically emphasizes performance optimization. Oracle Linux is the development standard at Oracle. It is the basis today for all Oracle products and services. The Oracle public cloud is built on Oracle Linux. Oracle engineered systems including Exadata, Exadata Cloud at Customer (ECC), Private Cloud Appliance (PCA), Private Cloud at Customer (PCC), Oracle Database Appliance (ODA), Oracle Zero Data Loss Recovery Appliance (ZDLRA), and Oracle Big Data Appliance (BDA) are all built on Oracle Linux. Most Oracle applications are developed on Oracle Linux. It therefore benefits Oracle significantly to have performance optimization built into Oracle Linux. And Linux performance optimization is a major focus for Oracle.

Oracle Linux has many available performance optimizations including:

- Enhanced memory performance. Oracle Linux seeks to better locate processes near its memory and to better place workloads that don't fit on a single NUMA node.
- SPECjbb® benchmark Java performance optimization[11] – delivers up to 3.6X performance improvement that helps eliminate lock contention.
- Accelerates performance for slower block devices with bcache that simplifies the use of SSDs as a block cache. This provides millions of IOPS on Storage Class Memory (SCM) and NVMe flash SSDs with a new scaled multi-queue block layer subsystem.
- Oracle Database and Oracle Linux engineering teams collaborate continuously on improvements and optimizations to boost database application performance. Some examples include:
    - Traditional inter-process communication (IPC) mechanisms tended to exhibit stability issues when subject to heavy loads. Oracle pioneered a new approach called Reliable Datagram Sockets (RDS). RDS is a low-latency connectionless protocol for delivering datagrams reliably to thousands of endpoints. RDS results in significantly fewer retransmissions. This is especially useful during times of peak processing because it greatly improves database performance on Linux. Oracle contributed the RDS code to the open source community. It is now part of the Linux kernel.
    - Oracle took advantage of Linux RDS simplifying Oracle Database code. They removed the now extraneous user code that had addressed the instability issues before RDS. This reduced database CPU consumption accelerating performance.

[11] Release Notes for UEK Release 4

- Extensive performance and scalability improvements to the process scheduler, memory management, file systems, and the networking stack. It's tuned to perform better and faster on x86 configurations with multiple CPU cores and large amounts of main memory.
- Optimized libraries and system calls improve Oracle Database queries performance.
- Developed Database Smart Flash Cache for Oracle Linux to accelerate IOs for read intensive database workloads. Database Smart Flash Cache empowers the database buffer cache to expand beyond System Global Area (SGA) in main memory to second-level cache that resides on a flash or SCM device. Because SCM is three to five times faster[12] reads than flash SSDs, which themselves are an order of magnitude faster HDDs, Database Smart Flash Cache significantly accelerates database performance. It does so only for the cost of the SCM or flash SSD.
- As previously discussed, Oracle contributes all Improvements to the Linux operating system upstream into the open source Linux community. This is so non-Oracle application workloads and DevSecOps can take advantage of these optimizations as well.

Oracle Linux performance optimizations are seamless and transparent to DevSecOps saving steps, testing, and most importantly, time.

Linux DevSecOps Database Performance Optimization Limitations

Most applications and micro-services take advantage of a database. Choosing the right database can mean the success or failure of the DevSecOps project. Very few databases take advantage of Linux performance optimizations. Fewer yet leverage Intel or AMD hardware performance optimizations. This is often quite frustrating to developers as it forces them to try and eke out more performance from their application code or general infrastructure. Doing so increases the time to completion and frequently infrastructure costs.

How Oracle Linux Enhances Oracle Database Performance for DevSecOps

The most mission critical database for the past four decades has been the Oracle Database. It is the world's first comprehensive database[13] supporting the vast majority of popular analytics methodologies including: relational, key value, time series, JSON, XML, Object, document, spatial, graphic, and AI-machine learning. Oracle Databases don't just provide AI-machine tools. A broad set of plug-and-play algorithms are available meaning no programming or data scientists are required to take advantage of Oracle's AI-machine learning. And all of the different analytics capabilities can have access to the same data. Oracle multi-tenant container databases (CDBs) and pluggable databases (PDBs) enable multiple database types to utilize one physical data copy reducing storage infrastructure. Less storage infrastructure translates into much reduced tasks and workloads.

Oracle and Intel worked closely in collaboration to performance optimize and scale Oracle Database applications running on Oracle Linux. For example:

- Intel optimized CPU threading algorithms that enable the Oracle Database to improve NUMA scalability by taking advantage of Intel SIMD and AVX instructions.
- Oracle modified the Oracle Database to take advantage of multi-threaded Intel® IPP (Intel® Integrated Performance Primitives) library. This accelerates columnar compression/decompression as well as encryption operations.
- Oracle Databases use Oracle Hybrid Columnar Compression (HCC) with Oracle Linux. HCC gets 10-15X data reduction or more[14]. The best deduplication and compression data reduction algorithms

[12] SCM was supposed to be 10 X lower latency than Flash NVMe SSDs because 3D XPOINT has 10 X lower latency than MLC NAND Flash. Real world results per the storage vendors, Pure Storage, Dell, NetApp has shown the SCM controller on the drive reduces overall latency differences by 50 to 70% equating into 3-5 X lower latency.

[13] DB-Engines ranking. Digging into each DB makes it clear Oracle is currently the only DB that supports OLTP, OLAP, Data Warehousing, time series, document, key value, object, JSON, XML, spatial, and graphical DBs in the same DB, thus making it the 1st comprehensive DB.

[14] Per Oracle Database, Exadata, and ZFS documentation, HCC ranges up to 50X data reduction for archival data. For most data warehouses, 10-15X is what is commonly found per Oracle System Engineers reported by Dragon Slayer Consulting 2019.

only get 2-5X data reduction for Oracle Databases[15]. This enables the Oracle Database to run faster and consume less costly storage.

- Oracle Database applications compiled on Oracle Linux can get the best Linux application performance by utilizing Intel's optimized compiler.

This means applications based on the Oracle Database running on Oracle Linux will get the best performance versus any other distribution. And as previously mentioned, performance and security go hand-in-hand. A very large insurance company discovered this when they tested the Oracle Database 11G on both Red Hat and Oracle Linux. The hardware infrastructure was identical for both. They found a non-trivial 15 to 20% greater throughput performance on secure Oracle Linux[16].

Known Linux DevSecOps Deployment Bottlenecks and Oracle Linux Answers

Few ever enjoy rewriting or modifying code if they don't have to. Yet, far too frequently, code must be modified before being placed into production. One of the key reasons for this is the DevSecOps environment is different from the production one. Public clouds may be at a different Linux OS version than other public clouds, or on-prem versions. Each environment may be at a different implemented security patch level or have a completely different Linux distribution. Even though Linux distributions are generally compatible, there can be and are significant differences.

Linux DevSecOps Application Deployment Headaches

If the DevSecOps development environment is different from the production environments, code and processes will need to be altered for production and testing, slowing down final deployments. It gets more complicated in hybrid environments especially when the Linux distributions may not be completely compatible. Applications may not live up to expectations or service level agreements or may not work. It's all just one more DevSecOps headache.

How Oracle Linux Simplifies DevSecOps Application Production Deployments

As previously mentioned, Oracle has built its entire business on Oracle Linux. All applications developed on Oracle Linux will also run without modification on Oracle Exadata, Oracle Exadata Cloud at Customer, Oracle Database Appliance, Oracle Private Cloud Appliance, Oracle Private Cloud at Customer, and the Oracle public cloud. Develop once and run everywhere.

As to compatibility between Oracle Linux and Red Hat Linux, Oracle treats any and all incompatibilities as if it were a bug. In the 13 years Oracle has been distributing Oracle Linux there has never been an open incompatibility ticket[17].

Other Oracle Linux DevSecOps Advantages

Oracle is the first to deliver autonomous Linux operations in the cloud with Oracle Autonomous Linux:

- Automatic provisioning
- Automatic scaling
- Automatic tuning
- Automatic online updating and patching
- Automatic security monitoring and remediation

Autonomous Linux is currently only available in the Oracle Cloud. However, some of these autonomous capabilities can be replicated utilizing standard tools within Oracle Linux.

[15] Per Dell EMC, HPE, Pure Storage, Hitachi, NetApp, and several other storage vendors, deduplication of Oracle Databases tops out in a best-case scenario of 5X, although typically closer to 2.5X reported by Dragon Slayer Consulting 2019.
[16] Oracle Case Study
[17] Oracle OpenWorld September 2019 Larry Ellison Keynote at the 15:50 mark

Conclusion

DevSecOps is navigating new territory by building in security from the start. Doing so without compromising performance, or modifying code upon production deployment, is complicated and difficult. Oracle Linux makes it simple.

By any measure, there is no better Linux distribution for DevSecOps than Oracle Linux.

For More Information on Oracle Linux DevSecOps

Go to: Oracle Linux

Paper sponsored by Oracle. **About Dragon Slayer Consulting:** Marc Staimer, as President and CDS of the 22-year-old Dragon Slayer Consulting in Beaverton, OR, is well known for his in-depth and keen understanding of user problems, especially with storage, networking, applications, cloud services, data protection, and virtualization. Marc has published thousands of technology articles and tips from the user perspective for internationally renowned online trades including many of TechTarget's Searchxxx.com websites and Network Computing and GigaOM. Marc has additionally delivered hundreds of white papers, webinars, and seminars to many well-known industry giants such as: Brocade, Cisco, DELL, EMC, Emulex (Avago), HDS, HPE, LSI (Avago), Mellanox, NEC, NetApp, Oracle, QLogic, SanDisk, and Western Digital. He has additionally provided similar services to smaller, less well-known vendors/startups including: Asigra, Cloudtenna, Clustrix, Condusiv, DH2i, Diablo, FalconStor, Gridstore, ioFABRIC, Nexenta, Neuxpower, NetEx, NoviFlow, Pavilion Data, Permabit, Qumulo, SBDS, StorONE, Tegile, and many more. His speaking engagements are always well attended, often standing room only, because of the pragmatic, immediately useful information provided. Marc can be reached at marcstaimer@me.com, (503)-312-2167, in Beaverton OR, 97007.