Server Virtualization Architecture and Implementation

By Jeff Daniels

Abstract

Virtual machine technology, or virtualization, is gaining momentum in the information technology community. While virtual machines are not a new concept, recent advances in hardware and software technology have brought virtualization to the forefront of IT management. Stability, cost savings, and manageability are among the reasons for the recent rise of virtualization. Virtual machine solutions can be classified by hardware, software, and operating system/containers. From its inception on the mainframe to distributed servers on x86, the virtual machine has matured and will play an increasing role in systems management.

Introduction

Virtualization in the enterprise is catching on across the country. Hardware vendors are packaging systems tuned to support virtual machines, and software vendors are developing virtual server management tools for migrations, performance, and high-availability. Customer IT organizations have defined a virtualization strategy and have begun deploying virtualized data centers.

The virtual machine concept has been around for years. The recent revolution in virtualization technology, hypervisors, and paravirtualization has allowed servers using the popular x86 architecture to operate efficiently and effectively with virtual machines.

Virtual machine technology is an enabler for service-oriented architectures, isolated secure systems, and flexible deployment.

This paper describes the virtual machine from its inception in the 1960s to present day virtual machines. Various types of virtualization will be discussed, as well as the associated costs and benefits of using virtual machines. Information from this paper should outline the basics of virtualization and offer key concepts when implementing virtualization technology.

What is a Virtual Machine?

A virtual machine (VM) is an abstraction layer or environment between hardware components and the end-user. Virtual machines run operating systems and are sometimes referred to as virtual servers. A host operating system can run many virtual machines and shares system hardware components such as CPUs, controllers, disk, memory, and I/O among virtual servers [8].

A "real machine" is the host operating system and hardware components, sometimes described as "bare metal," such as memory, CPU, motherboard, and network interface.

The real machine is essentially a host system with no virtual machines. The real machine operating system accesses hardware components by making calls through a low-level program called the BIOS (basic input/output system).

Virtual machines are built on top of the real machine core components. Goldberg describes virtual machines as "facsimiles" or a "hardware-software duplicate of a real existing machine" [4, 5]. Abstraction layers called hypervisors or VMMs (virtual machine monitors) make calls from the virtual machine to the real machine. Current hypervisors use the real machine hardware components, but allow for different virtual machine operating systems and configurations. For example, a host system might run on SuSE Linux, and guest virtual machines might run Windows 2003 and Solaris 10.

Virtual machine monitors and hypervisors are similar to "emulators." Emulation is a "process whereby one computer is set up to permit the execution of programs written for another computer" [9]. Hypervisors offer a level of efficiency, in that emulators translate every instruction or system call to the CPU, memory, and disk.

Hypervisors have specialized management functions that allow multiple VMs to co-exist peacefully while sharing real machine resources. Mallach concludes the differences are largely semantic because both hypervisors and emulators require I/O requests, memory mapping, and logical memory schemes [10].

Virtual Machine History

Virtual machines have been in the computing community for more than 40 years. Early in the 1960s, systems engineers and programmers at MIT recognized the need for virtual machines. In her authoritative discourse, "VM and the VM Community: Past, Present, and Future," Melinda Varian [17] introduces virtual machine technology, starting with the Compatible Time-Sharing System (CTSS). IBM engineers had worked with MIT programmers to develop a time-sharing system to allow project teams to use part of the mainframe computers. Varian goes on to describe the creation, development, and use of virtual machines on the IBM OS/360 Model 67 to the VM/370 and the OS/390 [17]. Varian's paper covers virtual machine history, emerging virtual machine designs, important milestones and meetings, and influential engineers in the virtual computing community.

In 1973, Srodowa and Bates [15] demonstrated how to create virtual machines on IBM OS/360s. In "An Efficient Virtual Machine Implementation," they describe the use of IBM's Virtual Machine Monitor, a hypervisor, to build virtual machines and allocate memory, storage, and I/O effectively. Srodowa and Bates touch on virtual machine topics still debated today: performance degradation, capacity, CPU allocation, and storage security.

Goldberg concludes "the majority of today's computer systems do not and cannot support virtual machines. The few virtual machine systems currently operational, e.g., CP-67, utilize awkward and inadequate techniques because of unsuitable architectures" [6].

Goldberg proposes the "Hardware Virtualizer," in which a virtual machine would communicate directly with hardware instead of going through the host software. Nearly 30 years later, industry analysts are excited about the announcement of hardware architectures capable of supporting virtual machines efficiently. AMD and Intel have revealed specifications for Pacifica and Vanderpool chip technologies with special virtualization support features.

The 1980s and early 1990s brought distributing computing to data centers. Centralized computing and virtual machine interest was replaced by standalone servers with dedicated functions: email, database, Web, applications. After significant investments in distributed architectures, renewed focus on virtual machines as a complimentary solution for server consolidation projects and data center management initiatives has resurfaced [14].

Recent developments in virtual machines on the Windows x86 platform merit a new chapter in virtual machine history. Virtual machine software from Virtuozzo, Microsoft, Xen, and EMC (VMWare) has spurred creative virtual machine solutions. Grid computing, computing on demand, and utility computing technologies seek to maximize computing power in an efficient, manageable way.

The virtual machine was created on the mainframe. It has only recently been introduced on the mid-range, distributed, x86 platform. Technological advancements in hardware and software make virtual machines stable, affordable, and offer tremendous value, given the right implementation.

Types of Virtualization

Virtual machines are implemented in various forms. Mainframe, open source, paravirtualization, and custom approaches to virtual machines have been designed over the years. Complexity in chip technology and approaches to solving the x86 limitations of virtualization have led to three different variants of virtual machines:

1. software virtual machines (see Figure 1), which manage interactions between the host operating system and guest operating system (e.g., Microsoft Virtual Server 2005);
2. hardware virtual machines (see Figure 2), in which virtualization technology sits directly on host hardware (bare metal) using hypervisors, modified code, or APIs to facilitate faster transactions with hardware devices (e.g., VMWare ESX); and
3. virtual OS/containers (see Figure 3), in which the host operating system is partitioned into containers or zones (e.g., Solaris Zones, BSD Jail).

A simple UNIX implementation called *chroot* allows an alternate directory path for the root file system. This creates a "jail," or sandbox, for new applications or unknown applications. Isolated processes in chroot are best suited for testing and applications prototyping. They have direct access to physical devices, unlike emulators.

Sun Microsystems' "Solaris Zones" technology is an implementation of chroot, similar to the FreeBSD jail design, with additional features. Zones allow multiple applications to run in isolated partitions on a single operating system [16]. Each zone has its own unique process table and management tools that allow each partition to be patched, rebooted, upgraded, and configured separately. Distinct root privileges and file systems are assigned to each zone.

Microsoft Corporation's Virtual Server 2005 is a new virtual machine manager in the market. After acquiring virtual machine technology from software vendor Connectix in 2003, Microsoft introduced the Virtual Server 2005 product, which runs on a Windows 2003 host and, predictably, supports Windows guest operating systems only. At the time of publishing this paper, Virtual Server is lim-

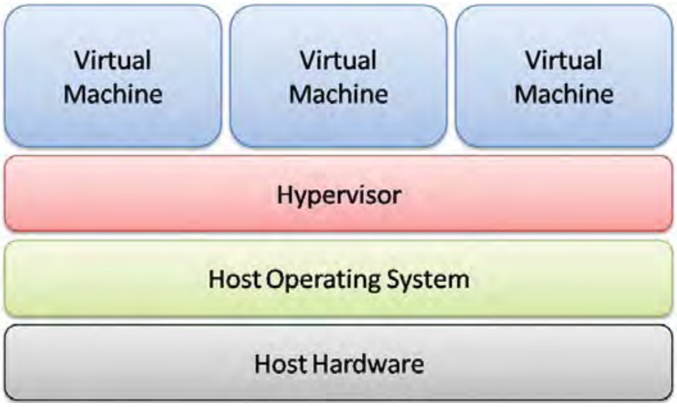


Figure 1: Software virtual machines.

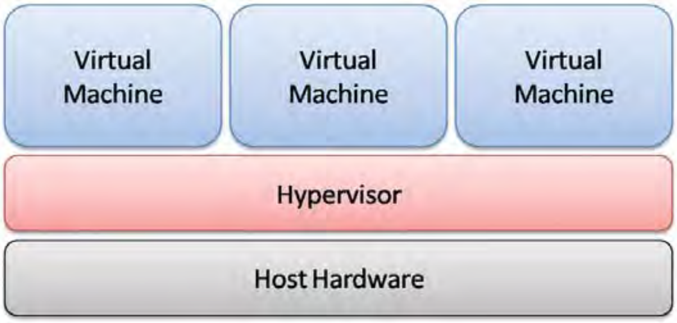


Figure 2: Hardware virtual machines.

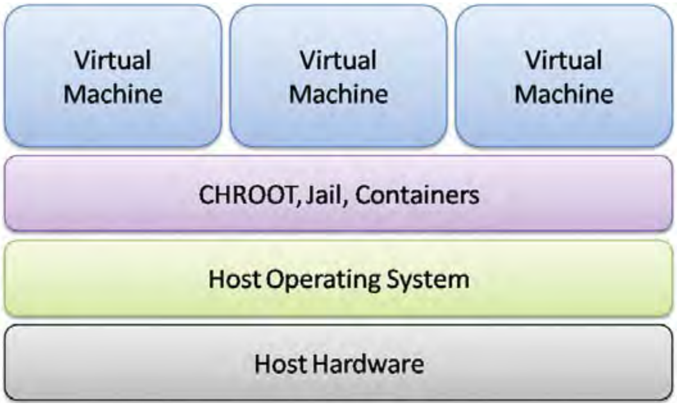


Figure 3: Virtual OS/containers virtual machines.

ited to running on single-processor hosts and cannot support symmetric multiprocessing (SMP).

SMP was introduced on RISC platforms, such as Sun Sparc and DEC Alpha chipsets, before being adopted on the x86 Intel Xeon and AMD Athlon processors. SMP allows multiple, identical chipsets to share one memory bank.

Instructions can be shared among the processors or isolated to a dedicated processor on the system. The system can share a workload, and with increased efficiency. A variation of SMP is AMD's Opteron technology, which allows dual-processor chips. The Opteron uses DDR SDRAM memory dedicated to each processor, as opposed to a single shared memory bank. The multiprocessing nature of numerous virtual machine guest servers on one host makes dual-core Opteron chips an attractive platform.

Paravirtualization is a variant of full operating system virtualization. Paravirtualization avoids "drawbacks of full virtualization by presenting a virtual machine abstraction that is similar but not identical to the underlying hardware" [18]. This technique allows a guest operating system to be "ported" through a special API (application programming interface) to run. The Xen paravirtualization research project, at the University of Cambridge, is a virtual machine monitor (hypervisor) that allows commodity operating systems to be consolidated and effectively mobilizes guests across physical devices. Xen currently supports only open source guest systems, though a Windows XP port is being developed. Denali is another paravirtualization implementation, but it requires significant modification to host system binaries and focuses on high-performance virtual machines.

EMC's VMWare technology is the market leader in x86 virtualization technology. VMWare ESX server uses a special hypervisor to "dynamically rewrite portions of the hosted machine code to insert traps wherever VMM intervention might be required" [1]. The VMWare solution is more costly, but it provides a robust management console and full-virtualization support for an array of guest operating systems including Solaris, Linux, Windows, and DOS.

Why Virtualization?

A recent Gartner survey revealed that "less than one-quarter of enterprises use virtual machines. However, more than 70 percent say they plan to test them in the near future" [12]. Data center floor space and rack space are prime real estate in computing environments. Cooling and electricity costs have risen in recent years. Infrastructure managers are looking to maximize the investment in existing computing power while keeping server sprawl and overhead costs in check.

Virtual servers generate hardware cost savings by allowing devices to be used to their full potential. Most distributed computing environments underutilize server capacity. Estimates for distributed, Windows-based servers indicate average capacity of 8 to 12 percent; UNIX servers use 25 to 30 percent of their capacity on average [3]. Virtual server technology unlocks unused capacity and allows the CPU, memory, disk, and controllers to be maximized for each physical device. Based on performance measurements, testing, estimates, and trial and error, any number of virtual servers can be added to a physical device, thereby increasing server utilization to sustainable levels. Instead of purchasing expensive servers with unused or excess capacity, a new virtual machine could be created for an application. Maintenance costs are avoided on the idle servers, and floor space is freed for virtual server hosts. A manageable growth plan can be created to add virtual servers, host servers, and related services.

The cost to implement virtual machines has significantly decreased. Recent virtual machine monitors, hypervisors, and paravirtualization tools make it easy to create virtual machine instances, instead of developing virtual machine code. The 1980 paper "A Virtual Operating System" identifies two costs to implement virtual machines: cost to write virtual machine software and implementation costs. The estimated cost of labor to develop the initial virtual machine monitor was eight to ten person-months and an estimated four person-months to port the entire system [7]. With current virtual machine monitors, an engineer can have an Oracle 10g cluster hosted on Red Hat Enterprise Linux running within minutes—basically, the amount of time it takes to download the binaries.

While the development and implementation costs of virtual machines are significantly less today than in 1980, "A Virtual Operating System" touches on another benefit of virtual machines: migration costs. Traditional systems are tied to server or desktop hardware. The life expectancy of servers is typically three to five years. When server technology becomes obsolete, the data must be migrated to a new platform, and applications must be reconfigured in the new environment. Worse, if the equipment is leased or acquired under a capacity services agreement, large scale system migrations must occur at the end of the term in order to avoid contract penalties. Virtual machines make those transitions easier. VMWare offers a migration tool called P2V, physical to virtual machine, which helps engineers move from legacy environments to virtual servers. Platespin Ltd. offers a flexible suite of tools to automatically migrate between virtual and physical machines (and back again), dynamically reallocate disk space, network configuration, unique identifiers, and other configuration settings. In contrast to traditional standalone systems, migrating virtual machines from one host platform to another host platform is relatively simple, in terms of configuration, man-hours, and resources required.

Licensing

Virtual servers can provide opportunities for software consolidation and reduced licensing costs. A Forrester study concludes Windows licenses and maintenance costs total $5,800 per year. Adapting to new virtual machine technology, many vendors have changed their licensing models to a "cost per instance" model instead of the "cost per processor" model.

Saving licensing fees when migrating from physical to virtual servers may not be effective under the cost per instance model. For example, Microsoft recently announced its new licensing model, noting that "per-processor licensed software will be licensed by virtual processor when used in a virtual OS environment and by physical processor when run in physical OS environments" [12]. However, virtual servers offer the ability to consolidate similar systems and software packages on common platforms to recognize license cost savings.

Consolidation is a key driver for many organizations implementing virtual machine technology. Nearly 60 percent of IT managers are considering consolidation [11] projects. Consolidation efforts represent an attempt by IT management to capture cost savings by retiring or decommissioning legacy devices and standardizing support processes. Consolidation projects present the opportunity to minimize the number of physical devices as well as software licenses, various packages, and management tools.

Once legacy systems are migrated to a consolidated, virtual environment, standardized images can be built and cloned to ensure integrity. High availability systems and clustered environments can be quickly configured with virtual machines.

Strategic initiatives can start with standardized images as a launching pad for new applications builds. When physical hosts need to be retired or phased out, virtual machines can easily be moved to the new platform with little interruption. Products such as Virtual Motion and Xen can move virtual machines on the fly, with little or no user downtime.

Virtualization in the IT Strategic Plan

Virtual servers should be a component in any Information Technology Strategic Plan (ITSP). As organizations plan for technologies, roadmaps are developed in one, three, five, seven, and out years. For example, an ITSP might have biometric readers on a three year plan, while an enterprise resource planning (ERP) upgrade is on a five year outlook. Virtualization technologies will fall in the one to three year planning cycle for most organizations.

The goal of IT projects and initiatives is to create business opportunities or generate cost savings. Virtualization is a key component in several planning areas:

- expanding business lines, such as shared and dedicated hosting;
- faster deployment, time to market;
- increased standardization, leading to lower total cost of ownership;
- consolidation efforts; and
- increased utilization of computing capital.

There are various other possibilities where virtual server technologies could create opportunities or cost savings. As business goals are defined and objectives determined by the business, virtualization technologies should be considered as one of the ways IT can help meet those goals.

Enterprise architecture is "the organizing logic for business process and IT infrastructure capabilities reflecting the integration and standardization requirements of the firm's operating model" [13]. Enterprise architecture seeks to align business goals and organizational needs with technology. The idea is to plan, deploy, and manage technologies to meet business objectives. Similar to the IT strategic plan, virtualization technologies have their place in the enterprise architecture model.

Ross mentions two important concepts in her definition of enterprise architecture: *integration* and *standardization*. Virtual servers offer increasingly flexible methods of systems integration. Hot failovers, highly available systems, real-time relocation of virtual systems, dynamic reallocation of system resources, and even wide-area network disaster recovery (backup) are integrated with virtual servers. The "data-center in a box" concept is a physical device with many integrated virtual servers that performs data center functions such as routing, messaging, and directory services.

Virtual servers go a long way towards standardization for infrastructure operations. Servers can be commoditized using the "gold image" model, where a virtual machine with the latest compliant system configuration is used to build new servers, ensuring standardization and change control. This also reduces risk of misconfiguration or non-configuration of features that may occur due to human error when building and rebuilding physical systems. Common platforms serve as an enabler for business objectives and other enterprise architecture components. Initiatives such as ERP implementations and service-oriented architecture applications rely on infrastructure being available, standardized, and usable. Virtual server technologies can be used as a building block in standardization and integration in enterprise architectures.

Virtual Server Implementation

Implementation plans differ at every organization. What is applicable for one industry or business may not work for others. However, there are some common implementation *techniques* that transcend business lines.

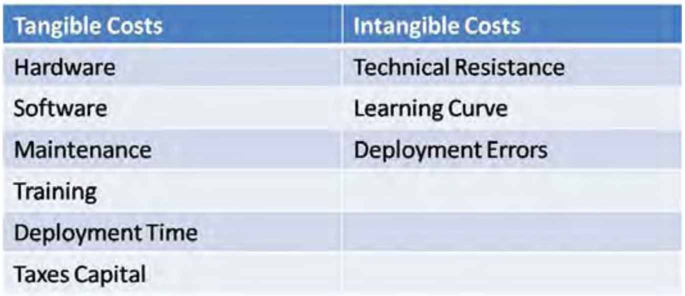
VMWare, a leading vendor of virtualization products, uses the VMWare Infrastructure Methodology (VIM): *assess*, *plan*, *build*, *manage*. The process considers the existing inventory of systems, creates a plan to "virtualize" the systems, install and configure the hosts, and manage the new virtual server infrastructure. Many organizations will follow these steps even if they are outside of the VIM methodology, but the figures, processes, and systems will be different.

Organizations tend to start using virtual servers in development systems, instead of production, to prove the new technology. Generally, the lower service levels and less criticality of development systems make an ideal choice to implement and evaluate the impact to the environment before going to production.

Teranet, an e-commerce and government systems integrator, offers a modified approach: perform the assessment, build a business case for all servers, perform a proof-of-concept, build a business case for all development and test servers, and, finally, deploy in phases. Using this implementation methodology, Teranet successfully deployed more than 270 virtual servers at a cost savings of over $3 million.

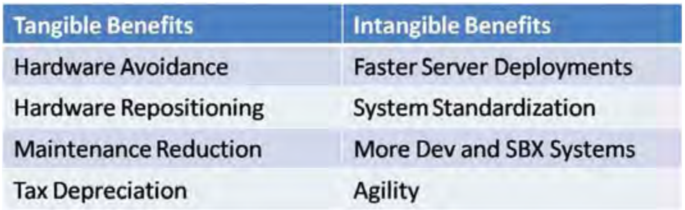
The phased approach was also used by Moen, the faucet manufacturer. Moen went through four phases of implementation, each integrating more virtualization technologies in the data center. In Moen's case, each phase had specific success criteria, such as cost avoidance, performance, cost reduction, and operating efficiencies [2]. The Moen team carefully evaluated success factors, process changes, and implementation goals following each phase. Moen also captured tangible and intangible costs and benefits throughout the implementation. The figures below show some of the types of costs and benefits identified by Moen.

Similar to the proof-of-concept approach, a pilot implementation is another way to "kick the wheels,“ so to speak. Pilots offer a quick win in many ways. Virtual server technology is proven during the pilot.



Tangible Costs	Intangible Costs
Hardware	Technical Resistance
Software	Learning Curve
Maintenance	Deployment Errors
Training	
Deployment Time	
Taxes Capital	

Figure 4: Moen tangible and intangible costs during implementation of virtual servers [2].



Tangible Benefits	Intangible Benefits
Hardware Avoidance	Faster Server Deployments
Hardware Repositioning	System Standardization
Maintenance Reduction	More Dev and SBX Systems
Tax Depreciation	Agility

Figure 5: Moen tangible and intangible benefits during implementation of virtual servers [2].

The pilot team will test-drive the systems and test functionality in an operational or small subset of systems. Pilots can promote virtualization success by sharing early wins with project management. Successful pilots allow users and project teams to gain valuable experience that will come in handy during full-scale production roll-outs.

Summary

Virtual machines have enjoyed a special niche within the information technology community over the years. Systems engineers and developers have continued to support virtual machines and push innovation in new ways. Virtual machines are gaining wider acceptance due to new efficiencies, ease of use, and users' demands for flexibility. Hardware, software, and operating system (container) virtual server technology are among the various virtual machine implementations.

There is no "one size fits all" virtual machine solution. Rather, many options are designed around specialized approaches. Hardware advances such as the AMD Opteron dual-core processors are making it possible to build powerful servers to host guest operating systems. Intel's Vanderpool and AMD's Pacific next-generation architecture will allow more flexible virtual systems at the hardware level.

Data centers and IT management are implementing virtual server technology, often as part of a consolidation strategy. Cost savings in the areas of software license management, systems management, data center, and overhead costs, such as electricity, generators, and floor space are key benefits for consolidated virtual server environments. IT managers trying to contain server sprawl, standardize and control systems, and build strategic platforms see virtual machine technology as an enabler.

Virtual storage area networks and grid computing are taking virtual machines to new levels. Advanced technologies such as high-performance computing, grid computing, and service-oriented architectures with dynamic allocation of resources are complimentary solutions to virtual machines. From its inception on the mainframe to distributed servers on x86, the virtual machine has matured and will play an increasing role in systems management.

References

1. Barnham, P., Dragovic, B., Fraser, K. et al. Xen and the art of virtualization. 2003. In *Proceedings of the 19th ACM Symposium on Operating System Principles* (*SOSP'03*). 164–177.
2. Buchwald, R. 2005. Many happy returns: Techniques on how to identify VMware return on investment and how to use it to justify VMware expansion. *VMWorld*. Presentation SLN693 (Oct. 20).
3. Day, B. 2005. Identifying server consolidation cost savings. Forrester Research, Cambridge, MA.
4. Goldberg, R. P. 1971. Virtual machines—Semantics and examples. *IEEE Computer Society Conference*. 141-142.
5. Goldberg, R. P. 1971. Hardware requirements for virtual machine systems. In *Proceedings of the Hawaii International Conference on System Sciences*.
6. Goldberg, R. P. 1973. Architecture of virtual machines. Honeywell Information Systems, Inc., Billerica, MA.
7. Hall, D. E., Scherrer, D. K., and Sventek, J. S. 1980. A virtual operating system. *Comm. ACM 23*, 9.
8. Kreuter, D. 2004. Where server virtualization was born. *Virtual Strategy Magazine* (July 21).
9. Lichstein, H. A. 1969. When should you emulate? *Datamatlon 15*, ii. 205-210.
10. Mallach, E. G. 1972. Emulation: A survey. *Honeywell Comput. J. 6*, 4. 287-297.
11. ONStor, Inc. 2005. Top 10 requirements for effective server consolidation. **www.onstor.com**.
12. Park, A. R. and Gammage, B. 2005. Microsoft updates server licensing to enable virtualization. ID Number G00132810. Gartner Group, Stamford, CT.
13. Ross, J. W. 2007. Enterprise architecture as a strategy. Center for Information Systems Research, MIT Sloan-CISR.
14. Serjeant, A. 2005. Building a case for server consolidation. *VMWorld*. Presentation (Oct. 20).
15. Srodawa, R. J. and Bates, L. E. 1973. An efficient virtual machine implementation. In *Proceedings of ACM SIGARCH-SIGOPS Workshop on Virtual Computer Systems*.
16. Tucker, A. and Comay, D. 2004. Solaris zones: Operating system upport for server consolidation. Sun Microsystems, Inc.
17. Varian, M. 1997. VM and the VM community: Past, present, and future. Office of Computing and Information Technology, Princeton University, Princeton, NJ.
18. Whitaker A., Shaw, M., and Gribble, S. D. 2002. Denali: Lightweight virtual machines for distributed and networked applications. Tech. rep. 02-02-01. University of Washington.

Biography

*Jeff Daniels (***jeff.w.daniels@gmail.com***) is a doctoral candidate in digital communications at the Indiana State University in Terre Haute. He has authored papers and presented numerous international conference presentations on topics including virtualization, security, and systems architecture. He is the recipient of several awards, including the Lockheed Martin Pinnacle Award for Excellence and the Lockheed Martin President's Award, and holds a Master's degree from Rensselaer Polytechnic Institute and a BS degree from the University of Central Florida.*