Virtualization Technologies and Cloud Security: advantages, issues, and perspectives

Parmanand Prabhat, Dept. of Computer Science & Engineering, Himalayan University, Itanagar, Arunachal Pradesh, India.
Dr. Syed Umar Professor, Dept. of Computer Science & Engineering,HMKS & MGS College of Engineering India.

Abstract. Virtualization technologies allow multiple tenants to share physical resources with a degree of security and isolation that cannot be guaranteed by mere containerization. Further, virtualization allows protected transparent intro- spection of Virtual Machine activity and content, thus supporting additional con- trol and monitoring. These features provide an explanation, although partial, of why virtualization has been an enabler for the flourishing of cloud services. Nev- ertheless, security and privacy issues are still present in virtualization technol- ogy and hence in Cloud platforms. As an example, even hardware virtualization protection/isolation is far from being perfect and uncircumventable, as recently discovered vulnerabilities show. The objective of this paper is to shed light on cur- rent virtualization technology and its evolution from the point of view of security, having as an objective its applications to the Cloud setting.

Key words: Virtualization, Security, Cloud.

1 INTRODUCTION

Virtualization technologies allow multiple tenants to share physical resources with a degree of security and isolation that cannot be guaranteed by mere containerization. Further, virtualization allows protected transparent introspection of Virtual Machine activity and content, thus supporting additional control and monitoring. These features provide an explanation, although partial, of why virtualization has been an enabler for the flourishing of cloud services. Nevertheless, security and privacy issues are still present in virtualization technology and hence in Cloud platforms. As an example, even hardware virtualization protection/isolation is far from being perfect and uncircumventable, as recently discovered vulnerabilities show. The objective of this paper is to shed light on current virtualization technology and its evolution from the point of view of security, having as an objective its applications to the Cloud setting.

2 Technology Background

Server virtualization is quickly becoming the preferred deployment model for corporate data centers, as companies look to tap into the benefits of managing servers on a software level. Switching to virtualization means that the workloads happening on servers are not tied to a specific piece of physical hardware and that multiple virtual workloads can occur simultaneously on the same piece of machinery. The immediate benefits of virtualization include higher server utilization rates in the data center and lower costs, but there are more sophisticated advantages as well.

2.1 VirtualizationFrameworks

The essential characteristics of the most widespread virtualization environments are summarized in Table 1. It is worth noting that all present hypervisors support full vir- tualization (also hardware-assisted virtualization in the following), as it offers relevant performance and isolation benefits. In fact, hardware virtualization allows the CPU to detect and possibly block unauthorized or malicious access to virtual resources. Never- theless, no virtualization framework is immune to bugs. The virtualization platform can be an additional attack surface.

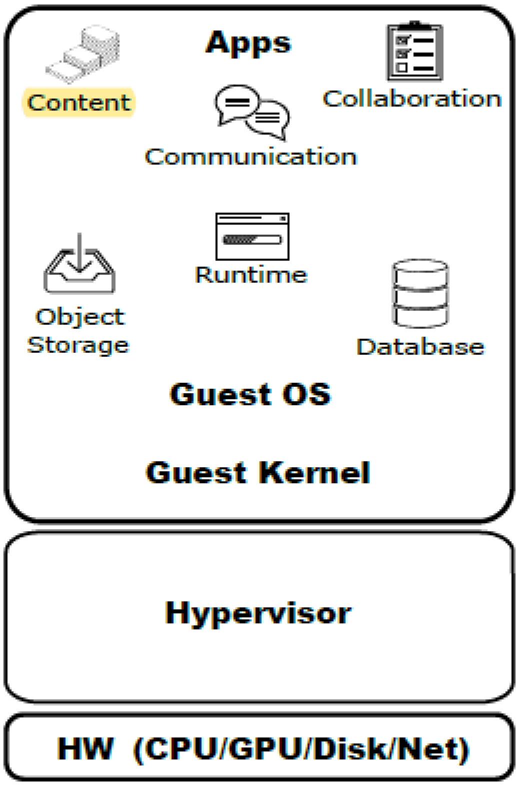

Fig. 1. Cloud layers and Virtualization

2.2 CPU Virtualization

CPU virtualization emphasizes performance and runs directly on the processor whenever possible. The underlying physical resources are used whenever possible and the virtualization layer runs instructions only as needed to make virtual machines operate as if they were running directly on a physical machine.

CPU virtualization is not the same thing as emulation. With emulation, all operations are run in software by an emulator. A software emulator allows programs to run on a computer system other than the one for which they were originally written. The emulator does this by emulating, or reproducing, the original computer’s behavior by accepting the same data or inputs and achieving the same results. Emulation provides portability and runs software designed for one platform across several platforms.

When CPU resources are overcommitted, the ESX/ESXi host time-slices the physical processors across all virtual machines so each virtual machine runs as if it has its specified number of virtual processors. When an ESX/ESXi host runs multiple virtual machines, it allocates to each virtual machine a share of the physical resources. With the default resource allocation settings, all virtual machines associated with the same host receive an equal share of CPU per virtual CPU. This means that a single-processor virtual machines is assigned only half of the resources of a dual-processor virtual machine.

2.3 GPU Virtualization

Virtualization technology for applications and desktops has been around for a long time, but it hasn't always lived up to the hype surrounding it. Its biggest failing: a poor user experience.

And the reason why is simple. When virtualization first came on the scene, GPUs — which are specialists in parallel computing — weren't part of the mix. The virtual GPU, aka vGPU, has changed that.

On a traditional physical computing device like a workstation, PC or laptop, a GPU typically performs all the capture, encode and rendering to power complex tasks, such as 3D apps and video.

With early virtualization, all of that was handled by the CPU in the data center host. While it was functional for some basic applications, CPU-based virtualization never met the native experience and performance levels that most users needed.

3 Virtualization Security Issues

Virtualization technologies underlying Cloud computing infrastructure themselves con- stitute vulnerable surface. In a Cloud scenario, we can observe the following major security challenges

privileged user access: access to sensitive data in the Cloud has to be restricted to a subset of trusted users (to mitigate the risk of abuse of high privilege roles);
– **lack of data/computation isolation**: one instance of customer data has to be fully isolated from data belonging to other customers;
– **Reliability/availability**: the Cloud provider has to setup an effective replication and recovery mechanism to restore services, should a security issue occur;

hypervisor: the hypervisor is the software element sitting in between the host and guests to allow mediated access to physical resources. This layer should be trans- parent to a non-privileged user running into the guest. Unfortunately, its presence cannot be fully hidden. As such, an attacker can exploit hypervisor vulnera- bilities to gain access to both the host system and other guests.

Hypervisors also provide emulation capabilities for missing hardware elements. However, this is a potential attack surface, as demonstrated by Ray and Jason.
– **pivoting**: users can often login into specific services hosted by a VM. Once in- side, the attacker could also exit the virtual machine she accessed, to damage the underlying physical system and/or sibling VMs.
– **Migration**: virtual machines can be moved over different hosts for load balancing or disaster recovery. This "migration" is performed by copying the VMimage over the network. An attacker can potentially eavesdrop data and perform a man in the middle attack if the channel is not encrypted.
– **resource allocation**: virtualmachines are usually executed on-demand at run-time, thus making the resource allocation and management process as dynamic as pos- sible. Resource sharing can thwart the security of the host system as well as of its virtual machines. In fact, negligence in cleaning resources before releasing them to others can lead to severe data leakage. As an example, data written by a VM into volatile or persistent storage can be accessed by others who have access to the same elements.
The above attacks show how virtual machines and the physical machines hosting them can be thwart by attackers targeting the host or just the virtual machine. Some mitigating approaches can be as follows:
– **host side**: vulnerabilities in the implementation of the hypervisor can somewhat be mitigated by frequently updating the hypervisor to reduce 0-days vulnerability window;
– **networkmonitoring**:monitoring and analyzing internal communications between sibling guests can help; nevertheless, malicious network behavior is difficult to detect by means of traditional intrusion detection systems and intrusion prevention systems;
– **encryption**: to mitigate such migration attacks, encryption of the data in transit can be used; nevertheless, this proves quite demanding on performance, and con- sequently on costs.
– **on allocation**: this attack can be dealt with by carefully deleting/cleaning resources either persistent or volatile that have been previously assigned to other VMs.

3.1 Co-Location issues

Co-location of virtual machines by different tenants on the same physical host is par- ticularly frequent in Cloud computing. Virtual resources assigned to a tenant might get hacked by other virtual resources assigned to different tenants that are co-located within the same physical machine. Co-location can lead to different issues as follows:
– **information leakage**: by reusing the same physical hardware to allocate virtual resources, tenants might be able to exploit forensic tools to recover sensitive data fromprevious tenants;
– **performance degradation**: malicious tenants co-located in the same physical host might be able to make an uneven/widely varying use of computational power with high cpu-intensive co-located virtual machines with the final goal of degrading victim's performances;
– **service disruption**: malicious tenants sharing physical resources with their victim might be able to lead the hardware to unexpected behaviors thus causing a service disruption against the victim.

A large number of research results have highlighted the actual existence of co- location vulnerabilities. Such papers show that completely preventing tenants from sharing the same physical resources is practically unfeasible (due to rising costs). A viable solution [3] might be an attribute-based approach where tenants can express constraints over both virtual and physical resource allocation. Tenants would be able to indicate an high data sensitivity, thus requesting to avoid co-location. In this way, co-location will not be allowed for virtual resources working on high sensitive informa- tion thus lowering the chance of data leakage. As a consequence, virtual resource cost would be increased. This could be an acceptable trade-off in most sensitive scenarios.

3.2 Randomness and Virtualization

Cloud providers usually deploy identical VM clones when needed to satisfy request load. As such, it can happen that very similar (oftentimes the very same) images are used for different tenants. As a consequence, the internal random pool for clone VMs is most probably the same/very similar for different VMs. An adversarymight exploit this weakness and try to guess the value of VMcryptographic keys. In order to address such issue, the Cloud or Service providers should try to increase the number of events fed to the entropy pool of VMoperating systems as soon as they are deployed, so as to provide an adequate level of security.

3.3 Container Security

Container security is the protection of the integrity of containers. This includes everything from the applications they hold to the infrastructure they rely on. Container security needs to be integrated and continuous. In general, continuous container security for the enterprise is about:

- Securing the container pipeline and the application
- Securing the container deployment environment(s) and infrastructure
- Integrating with enterprise security tools and meeting or enhancing existing security policies

Containers are popular because they make it easy to build, package, and promote an application or service, and all its dependencies, throughout its entire lifecycle and across different environments and deployment targets. But there are still some challenges to container security. Static security policies and checklists don't scale for containers in the enterprise. The supply chain needs more security policy services. Teams need to balance the networking and governance needs of containers. Build and runtime tools and services need decoupling.

A
Container security

By building security into the container pipeline and defending your infrastructure, you can make sure your containers are reliable, scalable, and trusted.

3.4 Unikernel Security

In this "modern" era of software development, the spotlight has bounced from virtual machines on clouds, to containers on clouds, to, currently, container orchestration… on clouds. As the "container wars" rage on, leaving behind multiple evolutionarily (or politically) dead-end implementations, unikernels are on the rise. Unikernels are applications built as specialized, minimal operating systems. While unikernels originated as an academic curiosity in the 90s, the modern crop are primarily focused on running as lightweight paravirtualized guests… on clouds.

3.5 Virtualization and Spectre/Meltdown

In the last several weeks, many of you have likely heard about the new security threat that involves the ability to exploit common features of modern CPUs. These attacks, known as “Meltdown” and “Spectre” can impact both bare metal and virtual servers. Red Hat Virtualization has added the “IBRS Family” of CPUs to the supported Cluster CPU type as a means to help protect against the IPRS and IBPM attacks that would result in guest attacks.

first step towards protecting your Red Hat Virtualization environment is to update all components to the latest version. RHV and/or RHEL hosts should be updated, Red Hat Virtualization Manager should be updated, and all guests should be updated.

The feature outlined below is available starting in Red Hat Virtualization 4.1.9 with the use of Intel Nehalem and newer CPUs, when the appropriate microcode is applied to the host(s). After updating the environment and then the Red Hat Virtualization Cluster CPU type to use a IBRS CPU Type (Spectre Variant 2 protection), all VMs in that cluster need to be stopped & started.

4 Virtualization Benefits for Security

Despite the compatibility challenges and fears of future hypervisor flaws, those interviewed still see tremendous potential for virtualization technology to improve security and manageability

Beaird, for example, has been able to improve his patch testing and deployment process through virtual systems. "A huge benefit for us has been the ability to test patches in a duplicate environment without the need for a separate dedicated hardware environment, which, for companies my size, isn't always feasible," he said, crediting VMware's VMotion technology for much of the improvements.

4.1 VirtualMachine Monitoring

VMWare provides the most comprehensive solution for server virtualization today. ManageEngine Applications Manager provides comprehensive performance metrics to monitor your VMware ESX/ESXi servers and their guest virtual machines, and helps you ensure they are performing well at all times. Applications Manager connects with VMware ESX/ESXi servers through APIs and determines the health status as well as the performance of host servers and their corresponding virtual machines.

With out-of-the-box reports, graphical views, alarms, thresholds and comprehensive fault management capabilities, administrators can maximize ESX server uptime and ensure that the guest virtual machines of the ESX/ESXi servers are running at peak performance.

4.2 Semantic Introspection and Modeling VMBehavior

Monitoring key Cloud components that would be targeted or affected by attacks is vital in order to protect the VMs and the Cloud infrastructure . By either actively or pas- sively monitoring key VM components any possible modification to VMdata and code can be traced and recorded.
In fact, virtual machine introspection is a process that allows observing the state of a VMfrom outside of it. Syringe is one example of a monitoring system making use of virtualization to observe and monitor guest kernel code integrity from a privileged

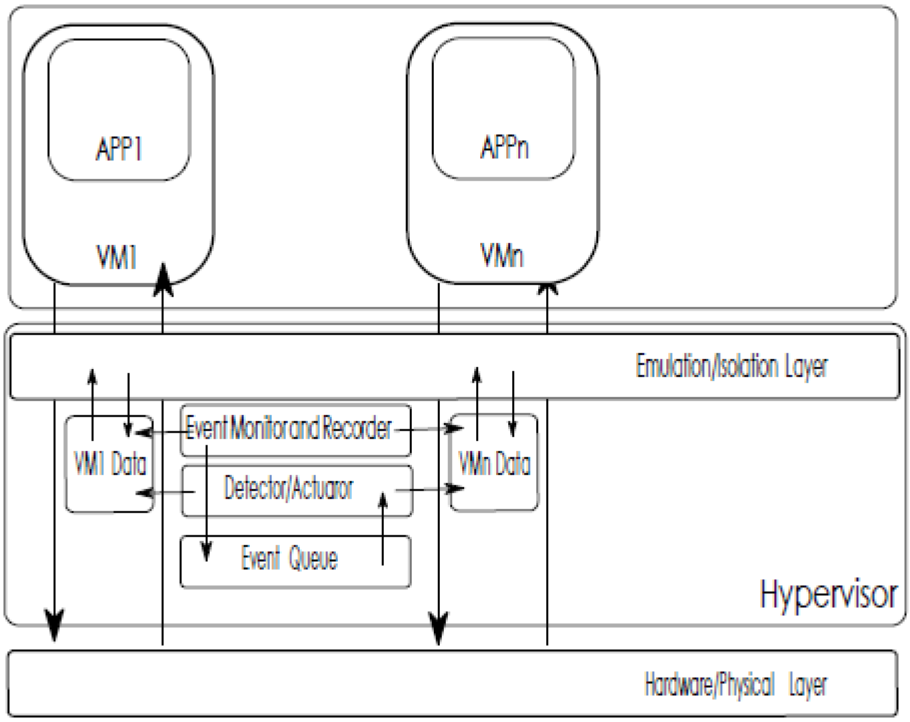

Fig. 3. Virtualization: Introspection Components

VMor fromthe VMM. However, it is quite simple for guest code to realize it is running inside a VMthat can potentially be a honeypot VM.

The approach depicted in Fig. 3 is an example of advanced transparent passive trac- ing and recording of VM events from the hypervisor.. Any relevant event or status change is recorded by an event interceptor and it is then stored in a pool of recorder warnings where the collected information is asynchronously evaluated (evaluator) and, if needed, a reaction is triggered (act) according to a chosen policy.

An interesting VM-introspection-based approach is CloRExPa providing vari- ous kinds of customizable resilience service solutions for Cloud guests, using execution path analysis. CloRExPa can trace, analyze and control live VMactivity, and intervened code and data modifications, possibly due to either malicious attacks or software faults. Execution path analysis allows the VMM to trace the VM state and to prevent such a guest from reaching faulty states, leveraging scenario graphs.

This trend towards semantic introspection of VM activity is a very active field also as regards mobile devices in the Cloud. This is the way to go for enabling control over possibly untrusted mobile Cloud nodes/applications. In fact, as will be detailed in the following for Bring Your Own Device (BYOD) untrusted devices, either they have to be banned altogether from the enterprise or enhanced semantics-aware introspection has to be put in place to prevent them from leaking sensitive information. Outside of the enterprise, semantic introspection allows legitimate users to regain control over their device internals. This approach will help detect and react to malware and to backdoors that are put in place even by trusted software or apps.

Themain problemwith introspection is that it requires knowing the internals and se- mantics of guest operating systems and running applications. This is especially difficult in case of closed-source OS and application such as inWindows and Mac environments. In fact, Windows OSes have always been the main target of malware that have exploited numerous bugs and vulnerabilities exposed by its implementations. Recent trusted boot technology plus additional integrity checks have rendered the Windows OS less vulnerable to kernel-level rootkits. Nevertheless, guestWindows VirtualMachines are becoming an increasingly interesting attack target. HyBIS is the only exam- ple of introspection system protecting present Windows OS Guests from malware and rootkits.

4.3 Finer-Grained Security

IT security aims to ensure the right people have access to the right resources and use them in the right ways. Making sure those are the only things that can happen is the "principle of least privilege," a cornerstone of enterprise security policy. Custom roles for Cloud IAM make that easier with the power to pick the precise permissions people need to do their jobs—and are now generally available.

Google Cloud Platform (GCP) offers hundreds of predefined roles that range from "Owner" to product- and job-specific roles as narrow as "Cloud Storage Viewer." These are curated combinations of the thousands of IAM permissions that control every API in GCP, from starting a virtual machine to making predictions using machine learning models. For even finer-grained access control, custom roles now offer production-level support for remixing permissions across all GCP services.

Security that’s built to fit

Consider a tool that needs access to multiple GCP services to inventory Cloud Storage buckets, BigQuery tables and Cloud Spanner databases. Enumerating data doesn’t require privileges to decrypt that data. While predefined roles to view an entire project may grant .query,. decrypt and .get as a set, custom roles make it possible to grant .get permission on its own. Since a custom role can also combine permissions from multiple GCP services, you can put all of the permissions for a service account in one place—and then share that new role across your entire organization.

5 Secure Enclaves and Virtualization

In Cloud computing environments, hardware resources are shared, and parallel com- putation widespread that can produce privacy and security issues when isolation is not enforced. In fact, the hypervisor is an important cornerstone of Cloud computing that is not necessarily trustworthy or bug-free. To mitigate this threat Intel and AMD in- troduced respectively SGX 3 and SEV 4, which transparently encrypt a vir- tual machines memory. Intel introduced the SGX hardware extensions to create a trusted execution environment (secure enclave or isolation container) within its CPUs. SGX claims runtime protection of a running process/VM even if the host OS and soft- ware components are malicious. Isolation containers are a primitive to minimize trusted software, leveraging trusted hardware and having a small performance overhead. This is a smart idea though present implementations (AMD SEV and Intel SGX) do still have some limitations, as we detail in the following.

5.1 Intel SGX

There is tremendous opportunity for application and solution developers to take charge of their data security using new hardware-based controls for cloud and enterprise environments. Intel Software Guard Extensions (Intel SGX) offers hardware-based memory encryption that isolates specific application code and data in memory. Intel SGX allows user-level code to allocate private regions of memory, called enclaves, which are designed to be protected from processes running at higher privilege levels. Only Intel SGX offers such a granular level of control and protection.

5.2 SGX Security Issues

Intel Software Guard Extensions (**SGX**) is a set of security-related instruction codes that are built into some modern Intel central processing units (CPUs). They allow user-level as well as operating system code to define private regions of memory, called *enclaves*, whose contents are protected and unable to be either read or saved by any process outside the enclave itself, including processes running at higher privilege levels. SGX is disabled by default and must be opted in to by the user through their BIOS settings on a supported system.

SGX involves encryption by the CPU of a portion of memory. The enclave is decrypted on the fly only within the CPU itself, and even then, only for code and data running from within the enclave itself. The processor thus protects the code from being "spied on" or examined by other code. The code and data in the enclave utilise a threat model in which the enclave is trusted but no process outside it can be trusted (including the operating system itself and any hypervisor), and all these are thus treated as potentially hostile.

6 Use Cases for Virtualization

Separate the computer/server from the hardware. The same disk image can be used across multiple types of hardware without having to install new drivers, and the machine can be migrated from physical machine to physical machine instantly.

More efficient resource use. A single physical server can be split into multiple machines/workloads and use the resources more efficiently. Less space is taken up and less power is used. Old physical servers can be virtualized to put them on newer hardware.

Fast deployment and snapshot ting. Deploying new machines can be nearly instantaneous, and machines can be snapshotted in time, allowing for rollback to any point in time.

Backup and transfer. Rather than backing up individual files, entire operating systems can be backed up or transferred, allowing for quick recovery. DR sites can easily mimic production more fully, and disk images can be stored on centralized storage such as SANs, NAS or other devices to improve performance and/or availability.

6.1 BYOD and Virtualization

Like it or not, enterprises have entered a “post-PC world,” where the network must accommodate new choices at every layer of the stack. These include traditional, mobile, and social applications and operating systems; various server architectures; and an array of mobile devices ranging from smartphones to tablets and other mobility tools. Cisco’s Internet Business Solutions Group (IBSG) conducted extensive research and analysis to uncover key insights about BYOD (“bring your own device”) and desktop virtualization trends in U.S. enterprises. The Cisco IBSG Horizons BYOD and Virtualization study surveyed 600 enterprises IT leaders from 18 industries. This paper offers an overview of the top 10 insights.

It’s important to note that BYOD is merely the “tip of the iceberg” when it comes to gaining the full benefits of mobility. Many other elements also come into play (for example, cloud will assume a key role in fulfilling the promise of BYOD). However, BYOD and desktop virtualization are already having a significant impact on the enterprise — a trend that is certain to grow in the months ahead.

6.2 Virtualization and Smartphones

The vision of the *Internet of Things (IoT)* is coming closer to reality as a large number of embedded devices are introduced to our everyday environments. For many commercial IoT devices, ubiquitously connected mobile platforms can provide global connectivity and enable various applications. Nevertheless, the types of IoT resource-utilizing applications are still limited due to the traditional stovepipe software architecture, where the vendors provide supporting software on an end-to-end basis. This paper tries to address this issue by introducing the *Sensor Virtualization Module (SVM)*, which provides a software abstraction for external IoT objects and allows applications to easily utilize various IoT resources through open APIs. We implement the SVM on both Android and iOS and show that the SVM architecture can lead to easy development of applications. We envision that this simplification in application development will catalyze the development of various IoT services.

6.3 Future Research Directions

This paper critically reviews the literature on environmental valuation of ecosystem services across the range of global biomes. The main objective of this review is to assess the policy relevance of the information encompassed by the wide range of valuation studies that have been undertaken so far. Published and other studies now cover most ecosystems, with aquatic and marine contexts attracting the least attention. There is also a predominance of single function valuation studies. Studies valuing multiple functions and uses, and studies which seek to capture the ‘before and after’ states as environmental changes take place, are rare. By and large it is the latter types of analyses that are most important as aids to more rational decision taking in ecosystem conservation versus development situations involving different stakeholders (local, national and global). Aggregate (global scale) estimates of ecosystems value are problematic, given the fact that only ‘marginal’ values are consistent with conventional decision-aiding tools such as economic cost–benefit analysis.

7 Conclusion

Virtualization is at the heart of Cloud computing. Albeit more lightweight approaches such as Containerization and Unikernels exist, hardware-supported isolation mecha- nisms provide beneficial in many different scenarios where security requirements are relevant. Nevertheless, security vulnerabilities are still a major issue, as highlighted by recently discovered exploits. Enhanced virtualization approaches and more effective isolation and monitoring technologies, that can also leverage additional computing re- sources of recent CPUs and GPUs, are still in their infancy. Such advances, coupled with appropriate software counterparts, will possibly improve the integrity and security of resources in Cloud, server farms, and in mobile scenarios.

References

1. Brasser, F., Mu¨ller, U., Dmitrienko, A., Kostiainen, K., Capkun, S., Sadeghi, A.: Software grand exposure: SGX cache attacks are practical. CoRR abs/1702.07521 (2017)
2. Canlar, E.S., Conti, M., Crispo, B., Di Pietro, R.: Windows mobile livesd forensics. J. Netw. Comput. Appl. 36(2), 677–684 (Mar 2013)

3 .AMD: Secure virtual machine architecture reference manual. http://www.0x04.net/doc/ amd/33047.pdflast accessed 2018-02-02 (2005)

4. Baiardi, F., Maggiari, D, Sgandurra, D., Tamberi, F.: Transparent process monitoring in a virtual environment. Electr. Notes Theor. Comput. Sci. 236, 85–100 (2009), http://dx. doi.org/10.1016/j.entcs.2009.03.016
5. Bijon, K., Krishnan, R., Sandhu, R.: Mitigating multi-tenancy risks in iaas cloud through constraints-driven virtual resource scheduling. In: Proc. of the 20th ACM Symp. on Access Control Models and Technologies. pp. 63–74. SACMAT ’15, ACM, New York, NY, USA (2015)
6. Brasser, F, Capkun, S, Dmitrienko, A, Frassetto, T., Kostiainen, K.,Mu¨ller, U., Sadeghi, A.: DR.SGX: hardening SGX enclaves against cache attacks with data location randomization. CoRR abs/1709.09917 (2017)
7. Cazalas, J., McDonald, J.T., Andel, T.R., Stakhanova, N.: Probing the limits of virtualized

   Software protection. In: Proc. of the 4th Program Protection and Reverse Engineering Workshop. pp. 5:1–5:11. PPREW-4, ACM, New York, NY,USA (2014)

8. Costan, V., Lebedev, I., Devadas, S.: Secure processors part i: Background, taxonomy for secure enclaves and intel sgx architecture. Foundations and Trends in Electronic Design Automation 11(1-2), 1–248 (2017)

9. Carbone, M., Conover, M., Montague, B., Lee, W.: Secure and robust monitoring of vir- tual machines through guest-assisted introspection. In: Research in Attacks, Intrusions, and Defenses Intl. Symp., RAID 2012. pp. 22–41 (2012), https://doi.org/10.1007/978-3- 642-33338-5_2

10. Chakrabarti, S., Leslie-Hurd, R., Vij, M., McKeen, F., Rozas, C., Caspi, D., Alexandrovich, I., Anati, I.: Intel software guard extensions (intel; sgx) architecture for oversubscription of secure memory in a virtualized environment. In: Proc Hardware and Architectural Support for Security and Privacy. pp. 7:1–7:8. HASP ’17, ACM, New York, NY, USA (2017)
11. Combe, T., Martin, A., Di Pietro, R.: To docker or not to docker: A security perspective. IEEE Cloud Computing 3(5), 54–62 (2016)
12. Dall, C., Nieh, J.: Kvm/arm: The design and implementation of the linux arm hypervisor. SIGARCH Comput. Archit. News 42(1), 333–348 (Feb 2014)

13. Costan, V., Lebedev, I.A., Devadas, S.: Sanctum: Minimal hardware extensions for strong software isolation. In: USENIX Security Symp. pp. 857–874 (2016)

14. Di Pietro, R., Franzoni, F., Lombardi, F.: HyBIS: Advanced introspection for effective windows guest protection. In: ICT Systems Security and Privacy Protection. pp. 189–204. Springer Intl. Publishing (2017)

15. Di Pietro, R., Lombardi, F., Signorini, M.: CloRExPa: Cloud Resilience via Execution Path Analysis. Future Gener. Comput. Syst. 32, 168–179 (mar 2014)

16. Di Pietro, R., Lombardi, F.,Villani, A.: CUDA Leaks: A detailed hack for cuda and a (partial) fix. ACMTrans. Embed. Comput. Syst. 15(1), 15:1–15:25 (Jan 2016)

17. Dowty, M., Sugerman, J.: GPU virtualization on vmware's hosted i/o architecture. SIGOPS Oper. Syst. Rev. 43(3), 73–82 (Jul 2009)

18. Dua, R., Raja, A.R., Kakadia, D.: Virtualization vs containerization to support paas. In: 2014 IEEE Intl. Conf. on Cloud Engineering. pp. 610–614 (March 2014)

19. Fernandes, D.A.B., Soares, L.F.B., Freire, M.M., Incio, P.R.M.: Randomness in virtual machines. In: 2013 IEEE/ACM 6th Intl. Conf. on Utility and Cloud Computing. pp. 282–286 (Dec 2013)

20. Gruss, D., Lettner, J., Schuster, F., Ohrimenko, O., Haller, I., Costa, M.: Strong and efficient cache side-channel protection using hardware transactional memory. In: 26th USENIX Security Symp. (USENIX Security 17). pp. 217–233. USENIX Association, Vancouver, BC (2017)

21. By Hertzsprung at English Wikipedia, C.B.S..: Execution rings. https://commons.wikimedia.org/w/index.php?curid=8950144

22. Gupta, V., Gavrilovska, A., Schwan, K., Kharche, H., Tolia, N., Talwar, V., Ranganathan, P.: GViM: GPU-accelerated virtual machines. In: Proc. of the 3rd ACM Workshop on System- level Virtualization for High Performance Computing. pp. 17–24. HPCVirt '09, ACM, New York, NY, USA (2009)

23. Hetzelt, F., Buhren, R.: Security analysis of encrypted virtual machines. SIGPLAN Not. 52(7), 129–142 (Apr 2017)

24. Hong, C.H., Spence, I., Nikolopoulos, D.S.: Gpu virtualization and scheduling methods: A comprehensive survey. ACMComput. Surv. 50(3), 35:1–35:37 (Jun 2017)

25. Intel: Intel virtualization technology specification for the ia-32 intel architec- ture. http://dforeman.cs.binghamton.edu/˜foreman/550pages/Readings/ intel05virtualization.pdflast accessed 2018-02-02 (2005)

26. Jason, G.: VENOM: Virtualized Environment Neglected OperationsManipulation. Available fromMITRE, CVE-ID CVE-2015-3456. (May 2015)

27. Jia, L., Zhu, M., Tu, B.: T-vmi: Trusted virtual machine introspection in cloud environments. In: Proc. of the 17th IEEE/ACM Intl. Symp. on Cluster, Cloud and Grid Computing. pp. 478–487. CCGrid '17, IEEE Press, Piscataway, NJ, USA (2017)

28. Lengyel, T.K.: Malware collection and analysis via hardware virtualization, doctoral dissertations. 964 (2015), https://opencommons.uconn.edu/dissertations/964

29. Jian, Z., Chen, L.: A defense method against docker escape attack. In: Proc. of the 2017 Intl. Conf. on Cryptography, Security and Privacy. pp. 142–146. ICCSP '17, ACM, New York, NY, USA (2017)

30. Kaplan, D., Powell, J., Woller, T.: AMD memory encryption. White paper (2016), https://developer.amd.com/wordpress/media/2013/12/AMD_Memory_Encryption_ Whitepaper_v7-Public.pdf

31. Kauer, B.: Oslo: Improving the security of trusted computing. In: USENIX Security Sympo-

Sium. pp. 229–237 (2007)

32. Kocher, P., Genkin, D., Gruss, D., Haas, W., Hamburg, M., Lipp, M., Mangard, S., Prescher, T., Schwarz, M., Yarom, Y.: Spectre attacks: Exploiting speculative execution. ArXiv e-prints 1801.01203 (Jan 2018)
33. Lee, S., Shih,M., Gera, P.,Kim, T.,Kim, H., Peinado, M.: Inferring fine-grained control flow inside SGX enclaves with branch shadowing. CoRR abs/1611.06952 (2016)

34. Lee, R.B.: Hardware-enhanced access control for cloud computing. In: Proc. of the 17th ACM Symp. on Access Control Models and Technologies. pp. 1–2. SACMAT '12, ACM, New York, NY, USA (2012)
35. Lombardi, F., Di Pietro, R.: Secure virtualization for cloud computing. J. Netw. Comput. Appl. 34(4), 1113–1122 (Jul 2011)

36. Lipp, M., Schwarz, M., Gruss, D., Prescher, T., Haas, W., Mangard, S., Kocher, P., Genkin, D., Yarom, Y.,Hamburg, M.: Meltdown. ArXiv e-prints 1801.01207 (Jan 2018)
37. M, J.A.: Performance comparison between linux containers and virtual machines. In: Intl. Conf. on Advances in Computer Engineering and Applications. pp. 342–346 (March 2015)
38. Madhavapeddy, A., Mortier, R., Rotsos, C., Scott, D., Singh, B., Gazagnaire, T., Smith, S., Hand, S., Crowcroft, J.: Unikernels: Library operating systems for the cloud. SIGPLAN Not. 48(4), 461–472 (Mar 2013)
39. Lombardi, F., Pietro, R.D., Soriente, C.: Crew: Cloud resilience for windows guests through monitored virtualization. In: Proc. of the 2010 29th IEEE Symp. on Reliable Distributed Systems. pp. 338–342. SRDS '10, IEEE Computer Society,Washington, DC, USA (2010)
40. Martin, A., Raponi, S., Combe, T., Di Pietro, R.: Docker ecosystem vulnerability analysis. C omputer Communications 122, 30 – 43 (2018)

41. Manu, A.R., Patel, J.K., Akhtar, S., Agrawal, V.K., Murthy, K.N.B.S.: A study, analysis and deep dive on cloud paas security in terms of docker container security. In: 2016 Intl. Conf. on Circuit, Power and Computing Technologies (ICCPCT). pp. 1–13 (March 2016)
42. Maurice, C., Neumann, C., Heen, O., Francillon, A.: Confidentiality issues on a GPU in a virtualized environment. In: Financial Cryptography and Data Security, Lecture Notes in Computer Science, vol. 8437, pp. 119–135. Springer Berlin Heidelberg (2014)
43. Merkel, D.: Docker: Lightweight linux containers for consistent development and deployment. Linux J. 2014(239) (Mar 2014)
44. Pan, Z., He, Q., Jiang, W., Chen, Y., Dong, Y.: Nestcloud: Towards practical nested virtualization. In: Proc. of the 2011 Intl. Conf. on Cloud and Service Computing. pp. 321–329. CSC '11, IEEE Computer Society,Washington, DC, USA (2011)
45. Moghimi, A., Irazoqui, G., Eisenbarth, T.: Cachezoom: How sgx amplifies the power of cache attacks. In: Cryptographic Hardware and Embedded Systems, CHES 2017. pp. 69–90. Springer Intl. Publishing, Cham (2017)