© 2014 IJIRT | Volume 1 Issue 5 | ISSN : 2349-6002

VIRTUAL MACHINE SECURITY

Sonarka Maini, Saurabh Jakhmola
Information Technology
Dronacharya College Of Engineering

***Abstract*- Computer security is a chronic and growing problem, even for virtual machines as evidenced by the seemingly endless stream of software security vulnerabilities. Security research has produced numerous access control mechanisms that help improve system security; however, there is little concensus on the best solution. Virtualization plays a major role in helping the organizations to reduce the operational cost, and still ensuring improved efficiency, better utilization and flexibility of existing hardware. "Virtualization is both an opportunity and a threat -says Patrick Lin, Senior director of Product Management for VMware". This paper presents a literature study on various security issues in virtualization technologies. Our study focus mainly on some open security vulnerabilities that virtualization brings to the environment. We concentrate on security issues that are unique for virtual machines. The security threats presented here are common to all the virtualization technologies available in the market, they are not specific to a single virtualization technology. We provide an overview of various virtualization technologies available in the market at the first place together with some security benefits that comes together with virtualization. Finally we provide a detailed discussion of several security holes in the virtualized environment.**

***Index Terms*- Virtualization, Security , Operating System**

I. INTRODUCTION

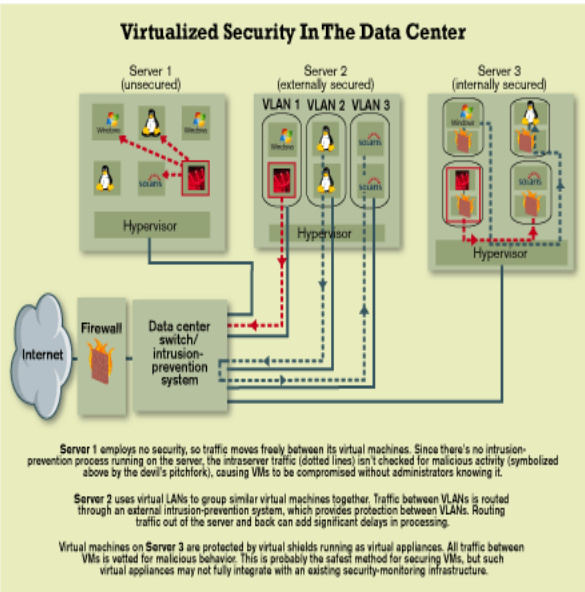
Security is a chronic and growing problem: as more systems (and more money) go on line, the motivation to attack rises. Linux is not immune to this threat: the “many eyes make shallow bugs” argument not withstanding, Linux systems do experience a large number of software vulnerabilities.

Security is a very broad concept, and so is the security of a system. All too often, people believe that a system is way more secure that it in practice is, but the biggest problems is still the human factor of he users; the possibility of careless or malicious users are commonly overlooked.

Virtualization essentially introduces a level of indirection to a system to decouple applications from the underlying host system. This decoupling can be leveraged to provide important properties such as isolation and mobility, providing a myriad of useful benefits. These benefits include supporting server consolidation by isolating applications from one another while sharing the same machine, improved system security by isolating vulnerable applications from other mission critical applications running on the same machine, fault resilience by migrating applications of faulty hosts, dynamic load balancing by migrating applications to less
loaded hosts, and improved service availability and administration by migrating applications before host maintenance so that they can continue to run with minimal downtime.
In non-virtual environment, the applications running on the machine can see each other, and in some cases can even communicate with each other, whereas in virtual environment the programs running in one guest machine are isolated from the programs running in another guest machine, in other words guest machines "provide what appear to be independent coexisting computers" to their running programs. The degree of isolation should be strong enough that the vulnerabilities in one virtual machine should not affect either the virtual machines or the underlying host machine.

OS virtualization provides a _ne granularity of control at the level of individual processes or applications, which is more beneficial than the hardware virtualization abstraction that works with entire OS instances. For example, OS virtualization can enable transparent migration of individual applications, not just migration of entire OS

instances.This granularity migration provides greater exibility and results in lower overhead [21, 24]. Furthermore, if the operating system requires maintenance, OS virtualization can be used to migrate the critical applications to another running operating system instance. By decoupling applications from the OS instance, OS virtualization enables the underlying OS to be patched and updated in a timely manner with minimal impact on the availability of application services [26]. Hardware virtualization alone cannot provide this functionality since it ties applications to an OS instance,and commodity operating systems inevitably incur downtime due to necessary maintenance and security updates. On the other hand new security protection programs are also emerging in the market every now and then from different vendors, but most of these security solutions are mainly focused on hypervisor. Since hypervisor is a new layer between the host's OS and virtual environment, it creates new opportunities for the malicious programs. And more over, hypervisor is basically a software program, so it has all the traditional software bugs and the security vulnerabilities as any software have. One of such product that hits the market recently is SHype [4], a new secure hypervisor that binds security policies to the virtual environment. A good debate on recent security solutions can be found on [10].



However, virtual machine security is more than just deploying a secure hypervisor to the environment. Virtualization technologies are still evolving. Newer versions with added features are introduced before the security consequences of the older version has been fully studied. This work analyzes the general security threats in a virtual environment and suggests possible solutions for few of the mentioned threats. Understanding of virtualization technologies greatly helps to understand the security consequences that occur in the environment. The back ground of various virtualization technologies together with some security benefits offered by these virtualization technologies and finally analyze the security issues concerning virtualization.

II. RESEARCH METHODOLOGY

This paper is a literature survey that analyse various issues concerning security in virtual machine environment. This work provides an overview of security consequences arises in a virtualized environment. However this paper does not provide one perfect solution for all the described threats. But do provide an understanding of how these threats can be avoided while implementing virtualization.

III. VIRTUALIZTION CONCEPTS

OS virtualization isolates processes within a virtual execution environment by monitoring their interaction with the underlying OS instance. Similar to hardware virtualization [25], applications that run within the virtual environment should exhibit an effect identical to that demonstrated
as if they had been run on the unvirtualized system. In addition, a statistically dominant subset of the application interaction with system resources should be direct to minimize overhead.
We classify OS virtualization approaches along two dimensions, host-independence and completeness. Host-dependent virtualization only isolates processes while host-independent
virtualization also decouples them. The distinction is that host-dependent virtualization simply blocks or filters out the namespace between processes, while host-independent virtualization provides a private virtual namespace for the applications' referenced OS resources. The former does not support transparent application migration since the lack of resource translation tables mandates that the resource identifiers of an application remain static across hosts for a migrating process, which can lead to identifier

conicts when migrating between hosts. Examples of host-dependent virtualization include Linux VServers and Solaris Zones .

Host-independent virtualization encapsulates processes in a private namespace that translates resource identifiers from any host to the private identifiers expected by the migrating application. Examples of this approach include Zap and Capsules . We refer to this virtual private namespace as a pod, based on the terminology used in Zap.In terms of completeness, partial virtualization virtualizes only a subset of OS resources. The most common example of this is virtual memory, which provides each process with its own private memory namespace but doesn't virtualize any other OS resources. As another example, the FreeBSD
Jail abstraction provides partial virtualization by restricting access to the filesystem, network, and processes outside of the jail, but does not regulate SysV interprocess communication (IPC) mechanisms. While partial virtualization has been used to support tighter models of security by limiting the scope of faulty or malicious processes, it can be unsafe if there exist direct or indirect paths for processes inside the environment to access resources outside or even break out of the environment. The chroot environment in Unix is a notorious example of a filesystem partial virtualization mechanism that has serious security shortcomings

Complete virtualization virtualizes all OS resources. While commodity OSs provide virtualization for some resources, complete virtualization requires virtualization for many resources that are already not virtualized, including process identifiers (PIDs), keys and identifiers for IPC mechanisms such as semaphores, shared memory, and message queues, and network addresses.
Within this taxonomy of virtualization approaches, complete and host-independent virtualization provides the broadest range of functionality, which includes providing the
necessary support for both isolation and migration of applications. An additional distinction between the taxonomies is in the scope of the application with respect to the available
systems. Virtualization approaches that are host-dependent and/or partial provide benefits only on a single host, while complete, host-independent virtualization approaches provide the support for applications to exploit the available systems that are accessible to the entire organization. The remainder of this paper focuses on the demands of supporting this more general form of virtualization in the context of commodity OSs.

IV. BACKGROUND

Virtualization was first developed in 1960's by IBM Corporation, originally to partition large mainframe computer into several logical instances and to run on single physical mainframe hardware as the host. This feature was invented because maintaining the larger mainframe computers became cumbersome. The scientist realized that this capability of partitioning allows multiple processes and applications to run at the same time, thus increasing the efficiency
of the environment and decreasing the maintainance overhead. By day to day development, virtualization technologies has rapidly attains popularity in computing, in fact it is now proven to be a fundamental building block for today's computing .

Although the main focus of this paper is to provide an overview of security vulnerabilities in a virtual environment. It is worth mentioning some of the security benefits that comes together with virtualization. Two primary benefits offered by any virtualization technology are 1.Resource sharing and 2.Isolation. Resource sharing - Unlike in non-virtualized environment where all the resources are dedicated to the running programs, in virtualized environment the VMs shares the physical resources such as memory, disk and network devices of the underlying
host. The resources are allocated to the virtual machine on request. Hypervisors plays a significant role in resource allocation.

Virtualization provides a facility of restoring a clean non infected environment even the underlying system is infected by malicious programs. Since, Virtualization provides an isolated environment this can be used for debugging malicious programs, and also to test new applications. Virtualization can be done in several ways. There are various virtualization technologies available in the market that helps to virtualize the environment. Depending on the needs

and goals of the organization, one virtualization technology is better than the other. This section gives an overview of some of the existing virtualization technologies.

V. SECURITY VULNERABILITIES IN VIRTUALIZATION

Most of security flaws identified in a virtual machine environment are very similar to the security flaws assoicated with any physical system. The following are some general flaws that are unique to the virtual environment.

VI. COMMUNICATION BETWEEN VM's OR BETWEEN VM'S AND HOST

One of the primary benefits that virtualization bring is isolation. This benefit, if not carefully deployed become a threat to the environment. Isolation should be carefully configured
and maintained in a virtual environment to ensure that the applications running in one VM dont have access to the applications running in another VM. Isolation should be strongly
maintained that break-in into one virtual machine should not provide access either to virtual machines in the same environment or to the underlying host machine. Shared clipboard in virtual machine is a useful feature that allows data to be transferred between VMs and the host. But this useful feature can also be treated as a gateway for transferring data between cooperating malicious program in VMs.In worst case, it is used to "exfiltrate data to/from the host operating system ". In some VM technologies, the VM layer is able to log keystrokes and screen updates across the virtual terminals, provided that the host operating system kernel has given necessary permission. These captured logs are stored out in the host, which creates an opportunity to the host to monitor even the logs of encrypted terminal connections inside the
VMs.
Some virtualization avoids isolation, in order to support applications designed for one operating system to be operated on another operating system, this solution completely exploits the security bearers in both the operating systems. This kind of system, where there is no isolation between the host and the VMs gives the virtual machines an unlimited access to the host's resources, such as file system and networking devices. In which case the host's file system becomes vulnerable .

VII. VM MONITORING FROM THE HOST

Host machine in the virtual environment is considered to be the control point and there are implications that enable the host to monitors and communicate with the VM applications
up running. Therefore it is more necessary to strictly protect the host machines than protecting distinctive VMs. Different virtualization technologies have different implications for the host machine to influence the VMs up running in the system. Following are the possible ways for the host to influence the VMs ,

- The host can start, shutdown, pause and restart the VMs.
- The host can able to monitor and modify the resources available for the virtual machines.
- The host if given enough rights can monitor the applications running inside the VMs.
- The host can view, copy, and likely to modify the data stored in the virtual disks assigned to the VMs.

And particularly, in general all the network traffic to/from the VMs pass through the host, this enables the host to monitor all the network traffic for all its VMs. In which case if a host is compromised then the security of the VMs is under question. Basically in all virtualization technologies, the host machines are given some sort of basic rights to control some actions such as resource allocations of the VMs running on top. But care should be taken when configuring the VM environment so that enough isolation should be provided which avoids the host being a gateway for attacking the virtual machine .

VIII. GUEST TO GUEST ATTACK

As mentioned it is important to prevent the host machine than the individual VMs. If an attacker gains the administrator privileges of the hardware then its likely that the attacker can break-in into the virtual machines. It is termed as guest-to-guest attack because the attacker can able to hop from one virtual machine to another virtual machine provided that the underlying security framework is already broken.

IX. EXTERNAL MODIFICATION OF A VM

There are some sensitive applications exists which rely on the infrastructure of the VM environment.

These applications running inside a virtual machine requires the virtual machine to be a trusted environment to execute that application. If a VM is modified for some reason, the applications can still be able to run on the VM but the trust is broken. Sudhakar and Andrew in their paper emaphasis more attacks on application virtualization. A best solution for this problem is to digitally sign the VM and validating the signature prior to the execution of this sensitive applications .

X. EXTERNAL MODIFICATION OF THE HYPERVISOR

As mentioned earlier in hypervisor is responsible for providing isolation between the guest machines. The VMs are said to be completely isolated or "self protected" only if the underlying hypervisor behaves well. A badly behaved hypervsior will break the security model of the system. There are several solutions exists for this problem, one of the recommended solution is to use secure hypervisor like SHype to ensure security in the hypervisor layer. Another solution is to protect the hypervisor from unauthorized modifications or enable the guest machines to validate the hypervisor.

XI. CONCLUSIONS

The paper has presented some of the security flaws in the virtual machine environment. Some of the threats presented here may be considered as benefits in some situations, but they are presented here so that proper care should be taken while designing and implementing the virtual environment.

Virtualization brings very little added security to the environment. One of the key issue is that everyone should be aware of the fact that virtual machines represent the logical instance of an underlying system. So many of the traditional computer threats apply the same to the virtual machines also. Another issue that makes the security consequences difficult to understand is that, there are so many different types of virtualization technologies available in the market. Each of it has it own merits and demerits, each virtualization deployment is different depending on the need for the virtualization. It is common that any single virtualization technology
will not provide shield to all the security issues arise. However, the key to create a good virtualization environment is to study carefully the environment that is to be virtualized,
the needs and goals of the organization, and taking into consideration all the possible security issues that puts the virtual machines at risk. Finally carefully design the virtual environment
with the help of correct virtualization technology that matches the goals.

Majority of the security issues presented here concerns the security of the host and the hypervisor. If the host or the hypervisor is compromised then the whole security model is
broken. Attacks against the hypervisor becoming more popular among the attackers realm . Therefore after setting up the environment, care should be taken to ensure that the hypervisor is secure enough to the newly emerging threats, if not patches has to be done. Patches should be done frequently so that the risk of hypervisor being compromised will be avoided.

Virtualization is a powerful solution to reduce the operational costs in today's computing but if done wrong it become as a threat to the environment. While implementing, exaggerate the security model to with stand the attacks. And as mentioned earlier keep monitoring for new developments that emerges in this field and continue to stay up to date.

REFERENCES

[1] http://www.symantec.com/avcenter/reference/Virtual_Machine_Threats.

[2] http://www.networkworld.com/weblogs/security/012014.html.

[3] VMware. *VMware security center*. http://www.vmware.com/support/security.html.

[4] R.P.Goldberg. Survey of virtual machine research. In *Computer*, volume 7, pages 34–35. IEEE, June 1974.

[5] http://www-128.ibm.com/developerworks/linux/library/l-linux-kvm/.