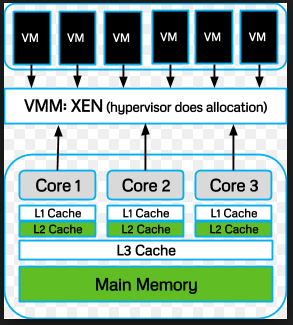
ABSTRACT

Virtualization in Cloud computing is making a virtual image of the storage devices or servers or network resources, so that they can be used in multiple machines at the same time.A virtual machine is the software implementation of any computing device, machine or computer that executes the series of instructions or programs as an actual machine.Security of hypervisor in cloud computing solves the attacks that allow a malicious virtual machine (VM) to compromise the Hypervisor.As well as techniques used by malicious VMs to steal more than their allocated share of physical resources, and ways to bypass the isolation between the VMs by using side-channels and prime- trigger-probe method to steal data are also discussed.

INTRODUCTION

Virtualization is used to run multiple virtual machines (VM) on a single hardware platform.This has the benefit of reducing costs for the customer and allowing the customer to quickly change server provisioning to meet the demand on their web services. But it requires multiple VM tenants who do not trust each other to share the same physical hardware where the data may be theft.The hypervisor is the piece of software that manages resources and isolates the VMs from each other. If a malicious VM is able to break free from the isolation that the hypervisor is supposed to guarantee, or compromise the hypervisor itself, then all the other VMs on the same physical machine are vulnerable it may steal data and can slow down the other VMs. Once co-resident, the malicious VM can attempt to compromise the hypervisor, or steal data using a side-channel.



This should be very concerning for any organization using the cloud that processes and stores sensitive customer data such as credit card information or social security numbers, and for organizations that perform highly proprietary work on the cloud. A part of the problem is that, as a customer using the cloud, you are not aware of who else is co-resident on the same physical hardware as your VMs.

Techniques discussed in this will show how it is possible for someone to get their VMs co-resident on the same physical hardware with their target VMs by using sidechannels. Once co-resident, the malicious VM can attempt to compromise the hypervisor, or steal data using a side-channel, or slow down the target VM by stealing physical resources.

**LITERATURE SURVEY**

**[1]Andrew R.Riddle;Soon M Chung “A Survey on the Security of Hypervisors in Cloud Computing”**

* This include attacks that allow a malicious virtual machine (VM).
* Techniques used by malicious VMs to steal more than their allocated share of physical resources (Xen,side-channel,co-residency)
* Security requirements and architectures for hypervisors to successfully defend against such attacks

**[2]S. Yu, X. Gui, J. Lin, X. Zhang, and J. Wang, “Detecting VMs Co-residency in the Cloud: Using Cache-based Side Channel Attacks,” Elektronika Ir Elektrotechnika, 19(5), 2013, pp. 73– 78.**

* In this the VMs Co-residency Detection Scheme via cache-based side channel attacks (VCDS) to get the location of the specified VM. Using load pre-processor based on cubic spline interpolation.
* VCDS computes the co-residency probability to describe VMs co-residency quantitatively. The experimental results show that VCDS improves the true detection rate.

**[3]B. Ding, Y. He, Y. Wu, and J. Yu, “Systemic Threats to Hypervisor Non-control Data,” Information Security, 7(4), 2013, pp. 349–354.**

* To secure hypervisors mainly focus on code or control-data integrity, without paying attention to non-control data integrity.
* Several types of non-control data within the Xen hypervisor

†Privilege related data

† Resource utilisation related data

† Security policy related data

**[4] J. Szefer, E. Keller, R. Lee, and J. Rexford, “Eliminating the Hypervisor Attack Surface for a More Secure Cloud,” Proc. of 18th ACM Conference on Computer and Communications Security, 2011, pp. 401–412.**

* We eliminate the hypervisor attack surface by enabling the guest VMs to run natively on the underlying hardware while maintaining the ability to run multiple VMs concurrently.

(i) pre-allocation of processor cores and memory resources

(ii) use of virtualized I/O devices

**[5] P. Karger, “Multi-level Security Requirements for Hypervisors,” Proc. of 21st Annual Computer Security Applications Conference (ACSAC 2005), 2005, pp. 267–275.**

* This paper will describe are the requirements and implications of a multi-level secure mode of operation as was defined many years ago in DoD Directive 5200.28 and in the implementing manual.
* In this mainly there are

(i) virtual machines Co-residency attacks : using side- channel mechanism.

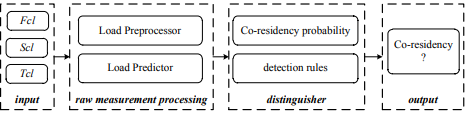
(ii) Prime-trigger-probe method.

(i)**CACHE-BASED SIDE CHANNEL ATTACKS :VMCD’S**

* Before detailing, we divide VMs into three roles in the attack scenario. They are malicious VM, victim VM and noisy VM, respectively. Malicious VM belongs to the attacker, which probes and analyzes cache activities. Victim VM is the attack target. Noisy VM resides in the same host with the malicious VM, which interferes with the attack.
* Commonly, cache-based side channel attacks consist of two major steps:

(1) measuring the cache load and

(2) analyzing the cache load to extract the private information.



* Here,It consists three cache load measurements sets

1.The first set is obtained before accessing the services of victim VM, marked as First cache load set, short for Fcl=(fcl1,fcl2,…, fcln)

2. The second set is probed when victim VM is doing the computation produced by accessing the service, marked as Second cache load set, short for Scl=(scl1, scl2,…,scln)

3. The third set is probed after the end of the service computation, marked as Third cache load set, short for Tcl=(tcl1, tcl2, …, tcln).

VCDS consists of two major modules which are raw measurement processing and distinguisher.

Working of CACHE-BASED SIDE CHANNEL ATTACKS :VMCD’S:

* Step 1: Load preprocessor takes as its inputs Fcl, Scl and Tcl, and returns Fcl1 , Scl1 and Tcl1 which respectively represent the load trend of Fcl, Scl and Tcl.
* Step 2: Load predictor takes as its inputs Fcl1 and Scl1 respectively, and returns Fcl2 and 2 1 Scl , which are the estimated caches load in the near future for Fcl1 and Scl1 . Meanwhile, choosing specified elements from Scl1 and Tcl1 to construct 2 2 Scl and Tcl2
* Step 3:Obtain 3 Fcl , 3 1 Scl , 3 2 Scl and 3 Tcl according to 2 Fcl , 2 1 Scl , 2 2 Scl and 2 Tcl , based on the normal cloud model [8]. Fcl3 includes twofold information of the load value and its membership degree. And the same to 3 1 Scl , 3 2 Scl and Tcl3 .
* Step 4: Distinguisher takes as its inputs Fcl3 , 3 1 Scl , 3 2 Scl and Tcl3 , and returns the result based on the co-residency probability computing algorithm and the detection rules