Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds (T. Ristenpart, Eran Tromer, Hovav Schacham and Stefan Savage CCS 2009)

Evripidis Paraskevas (ECE Dept. UMD)
04/09/2014

"Hey! You! Get off of my cloud

Don't hang around 'cause two's a crowd "

Rolling Stones

Outline

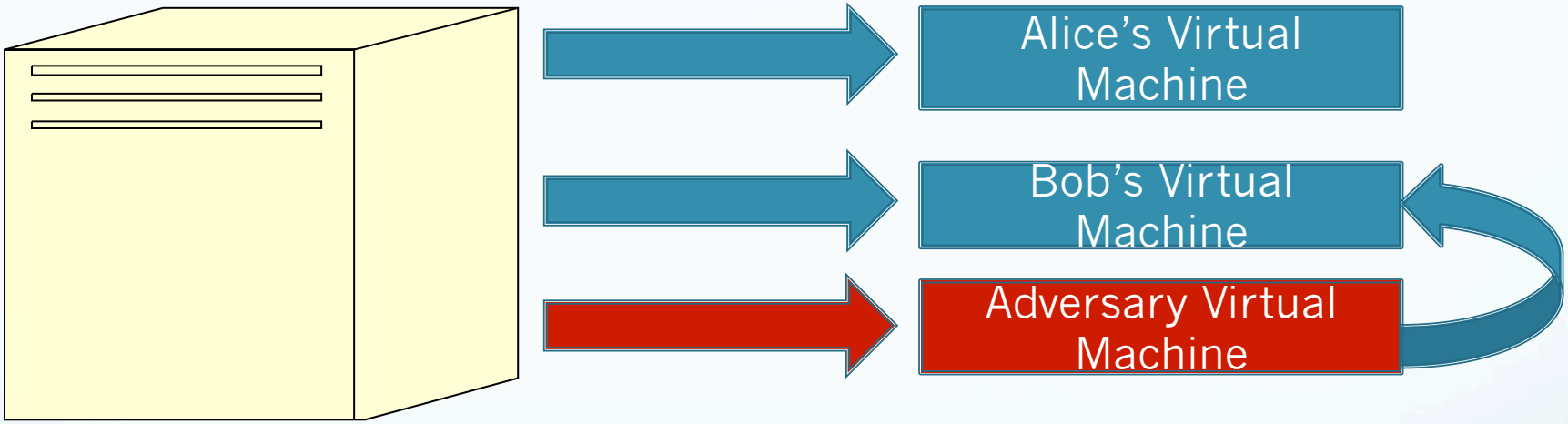
- Problem Definition and Motivation
- Threat Model
- Outline the attack
- Discuss the Feasibility and the Consequences of the attack

Cloud Computing

- What is the cloud?
  - The new infrastructure for hosting data and deploying software and services
  - Software, Platform, Infrastructure as a Service (SaaS, PasS, IaaS)
- Benefits
  - Cost Savings
  - Scalability
  - Flexibility
- New Threats
  - Trust relationship between customer and cloud provider
  - Multi-tenancy (security threat)

Multi-tenancy

- Multiplexing VMs of disjoint customers upon the same physical hardware (e.g. your instance is placed on the same server with other customers)



- New Risks
  - Side Channels Exploitation
  - Vulnerable VM isolation mechanisms
  - Lack of control who you are sharing the space with

Motivation

- Explore the threats of multi-tenancy in cloud computing
- Provide experimental results of the impact of these threats using a real cloud service provider (Amazon EC2) as a case study

Attack

- The attack considered requires two main steps:
  - *Placement* - Place a malicious VM on the same physical machine as that of the victim
  - *Extraction* - extract confidential information for the victim via a side channel attack

Threat Model

- Provider and infrastructure are trusted
- Attacker
  - Not affiliated with the provider (third-party user)
  - Can run many instances in the cloud
  - His instances might be placed on the same physical hardware as potential victims
  - Might manipulate shared physical resources

What Can Be Achieved By the Attacker?

- Can one determine where in the cloud infrastructure an instance is located?
- Can one easily determine if two instances are co-resident on the same physical machine?
- Can an adversary launch instances that will be co-resident with other user's instances?
- Can an adversary exploit cross-VM information leakage once co-resident?

AMAZON ELASTIC COMPUTER CLOUD (EC2)

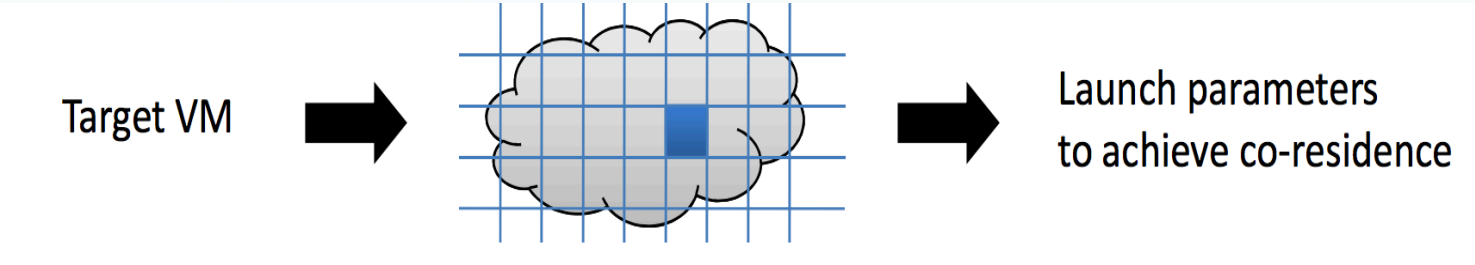
- Scalable, pay-as-you-go compute capacity in the cloud
- Customers can run different operating systems within a virtual machine
- Different regions available
  - US, EU, Asia
- Regions split into availability zones
  - In US: East (Virginia), West (Oregon), West (Northern California)
- EC2 Instances:
  - m1.small m1.large m1.xlarge c1.medium c1.xlarge

EC2 Highlights

- Xen (Virtual Machine Monitor)
- Domain0 (Dom0) privileged virtual machine configured to route packets for its guest images and reports itself as a hop in traceroutes
- Each account can run 20 VM instances
- When an instance is launched, it is assigned to a single physical machine for its lifetime
- Each instance assigned internal and external IP address and domain name

CLOUD CARTOGRAPHY

- Instance placing is not disclosed by Amazon but needed in order to carry out the attack
  - Map the EC2 service to understand where instances are placed



- Hypothesis
  - Different availability zones and instance types correspond to different IP address ranges
- Evaluation
  - Survey public servers on EC2 to map internal addresses to public addresses
  - Review addresses assigned to a large number of launched instances

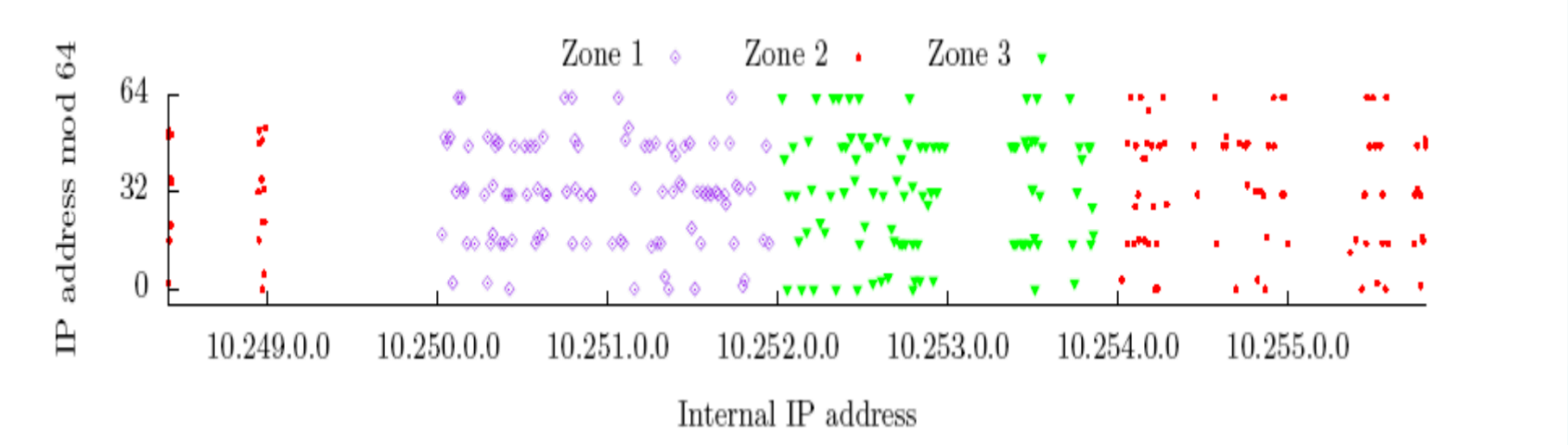
Survey Public Servers on EC2

- Create a set of public EC2-based web services
- Use WHOIS to identify distinct IP address prefixes associated with EC2 (/17, /18, /19)
- Use external probes to find responsive Ips
- Translate all responsive public IPs into internal IPs using the EC2's internal DNS
  - 14054 unique internal IPs

Instance Placement Parameters

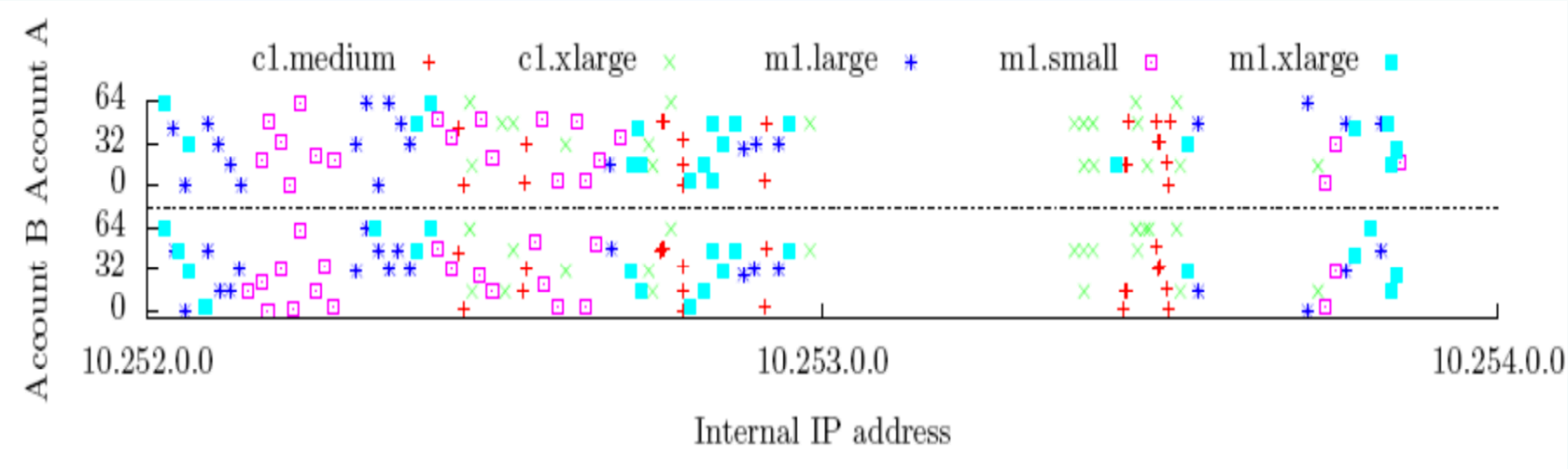
- EC2's internal address space is cleanly partitioned between availability zones
  - 20 instances launched for each of the 15 availability zone instance type pair
- How about the instances?
  - 20 instances of each type launched
  - Two accounts used (A & B)
  - B's instances launched 39 hours after terminating A's

Results (1)



The figure shows that different availability zones correspond to different IP address ranges

Results (2)



Some instance types correspond to same IP address ranges.

55 of 100 Account B instances had IP address assigned to Account A instance

Seems that user account doesn't impact placement

Determining Co-Residence

- Use the identified targets
- Network-based co-residence checks:
  - Matching Dom0 IP address
  - Small packet round-trip times
  - Numerically close internal IP address (within 7)
- Verify co-residence
  - Hard disk based covert channel on the 3 co-residence checks

Effective Co-Residence Check

- Compare internal IP addresses to see if they are close (within 7 internal IP addresses)
- If yes, perform a TCP SYN traceroute to an open port on the target and see if there is only a single hop (Dom0 IP)
- Very “quiet” check (little communication with the victim)

Exploiting Placement in EC2

- Place an attacker's instance on the same physical machine as a victim.
- Two strategies to achieve "good" coverage:
  - Brute-forcing Placement (launch many instances over a relatively long period of time)
  - Abusing Placement Locality (Target recently launched attacks)

Brute-Force Placement

- Infer likely victim zone and instance type from cloud map
- Launch many instances within a time frame and hope for co-residency
- Of 1686 target victims co-residence achieved with 141 victim servers (8.4% coverage)

Abusing Placement Locality

- Launch lots of instances right after the launch of victim's instance
  - Attacker may trigger a new instance launch by overloading the victim with requests (auto-scaling)
- Experiment
  - Single victim instance is launched
  - Attacker launches 20 instances within 5 minutes
  - Perform co-residence check
- 40% of the time the attacker launching just 20 probes achieves co-residence against a specific target instance

Results

	# victims v	# probes p	coverage
Zone 1	1	20	1/1
	10	20	5/10
	20	20	7/20
Zone 2	1	20	0/1
	10	18	3/10
	20	19	8/20
Zone 3	1	20	1/1
	10	20	2/10
	20	20	8/20

Trial	Account A	Account B	Total
Midday (11:13 – 14:22 PST)	2 / 5	2 / 5	4/10
Afternoon (14:12 – 17:19 PST)	1 / 5	3 / 5	4/10
Night (23:18 – 2:11 PST)	2 / 5	2 / 5	4/10

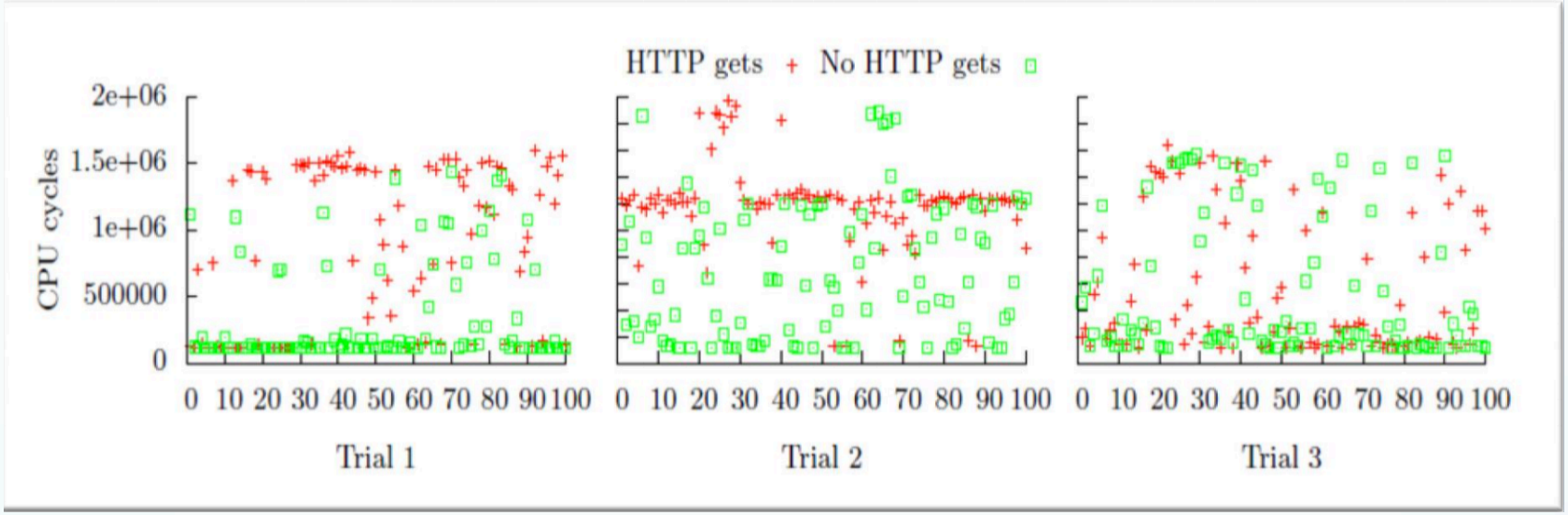
Figure 3: (Left) Results of launching p probes 5 minutes after the launch of v victims. The rightmost column specifies success coverage: the number of victims for which a probe instance was co-resident over the total number of victims. (Right) The number of victims for which a probe achieved co-residence for three separate runs of 10 repetitions of launching 1 victim instance and, 5 minutes later, 20 probe instances. Odd-numbered repetitions used Account A; even-numbered repetitions used Account B.

Much better coverage using time locality

Cross-VM Information Leakage

- Measuring cache usage
  - Time-shared cache allows an attacker to measure when other instances are experiencing computational load
- Load-based co-residence check
  - Co-residence check can be done without network-based technique
  - Use a priori knowledge about load variation
  - Induce Computational load (lots of HTTP requests) and observe

Results

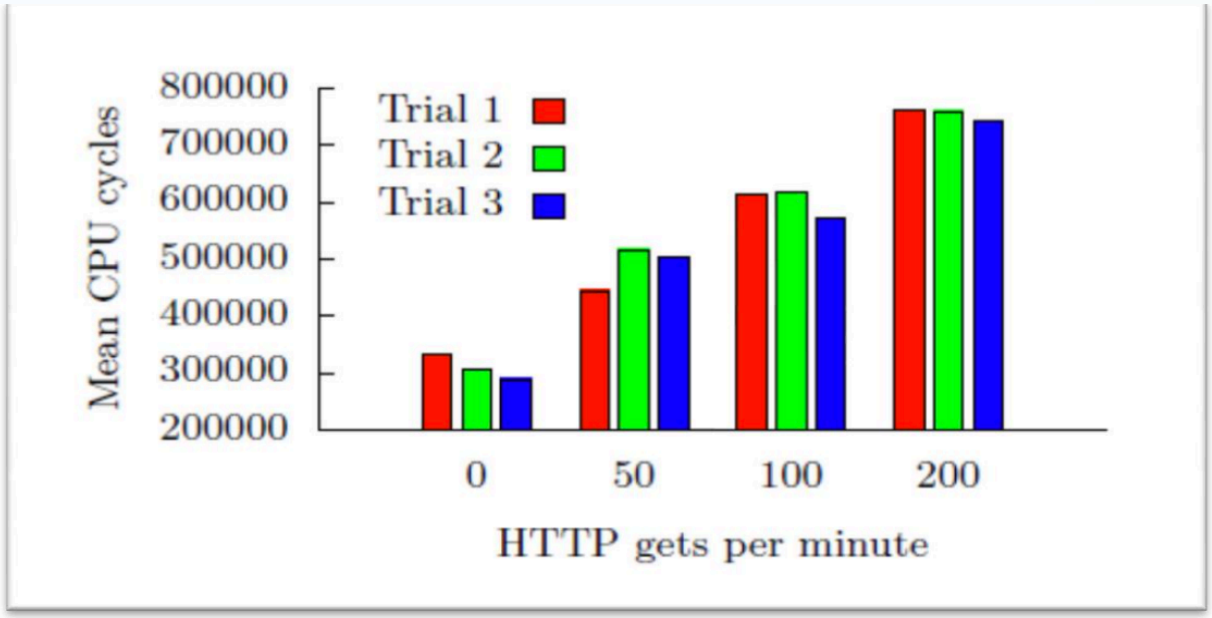


Instances in Trial 1 and Trial 2 were co-resident on distinct physical machines; instances in Trial 3 were not co-resident

Estimating traffic rates

- Load measurement might provide a method for estimating the number of visitors to a co-resident web server
- Perform 1000 cache load measurements in which
  - no HTTP requests are sent
  - HTTP requests sent at a rate of 50 per minute
  - 100 per minute
  - 200 per minute

Results



There is a clear correlation between traffic rate and load sample

Keystroke timing attack

- Cache load measurements used to mount a key stroke attack
- The goal is to measure the time between keystrokes made by a victim typing a password (in a SSH terminal)
- Inter-keystroke times if properly measures can be used to perform recovery of the password

Conclusion

- Attacker can achieve co-residence of a malicious VM with a victim's VM
- Shared Physical Infrastructure will cause problems (extract of sensitive information through side-channel attacks)
- Practical attack performed
- Countermeasures?? (small talk)

Thank you!!

Questions???