### 8271 discussion of: "Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds"

Stephen McCamant (Original paper: Thomas Ristenpart, Eran Tromer, Hovav Shacham, and Stefan Savage) University of Minnesota (Original paper: UC San Diego and MIT)

### Old and new topics in security

- Paper type 1: new idea, never been done before
  - Main contribution is novelty
  - Incentive to be first, maybe even a race

### Paper type 2: improvement in an

### already-busy area

- Contributions judged differentially
- Incentive to optimize

# Cloud threats, old and new Old: your system's regular vulnerabilities New but understood: need to trust cloud provider Focus here: attacks from cloud neighbors

### Case study: Amazon EC2

- Largest, highest-profile infrastructure cloud provider
- World-spanning data centers, instance sizes \$0.02-\$6.82 per hour
- Many instance types use Xen to multiplex one physical machine

### Ethical/legal sidebar

- Important for academic researchers to do things "by the book"
- Ethical obligations may be greater or less than legal ones
- 🖲 Here: CFAA, EC2 user agreement

### Placement and extraction

- Placement: get an instance on the same physical machine as the victim
- Extraction: given placement, get confidential info

### Network probing

- TCP traceroutes, port 80 and 443 scans, DNS resolution
- Instances have one name, but separate public and internal IP addresses

### Network mapping

- Internal addresses reflect topology
- Disjoint by availability region, clustered by instance type
- DomOs in an adjacent block

## Network-based co-residence checks DomO in traceroute (easiest) Close IP addresses Smallest packet round-trip times All found to have "effectively zero" false positives

### Hard disk usage channel

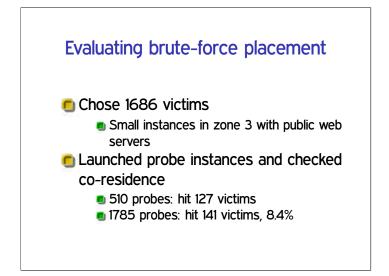
- Measure contention for hard disk (e.g., seek times) between VMs
- No attempt to optimize bandwidth: 0.0005 bits/sec (33 mins per bit)
- Why so slow?

### Covert channels and side channels "Covert channel": generally send and receiver cooperate One classification: storage channels, timing channels "Side channel": "sender" is passive victim

Can again include timing, also error messages, power usage, etc.

### Observed placement locality

- Sequential locality: new instance likely to use same machine as old dead one
- Parallel locality: instances started close in time more likely to share
- Non-locality: one account never given two instances on same machine



### Using locality

- Idea: use parallel locality, try to start probes soon after victim
  - Perhaps can trigger victim start, such as if it's based on demand
- About 40% coverage for 20 victims and 20 probes
- Also demonstrated against demos of commercial services

### Cache: Prime+Trigger+Probe

- 1. (Prime) Fill cache with my data
- Busy loop until preempted (recognize with TSC)
- 3. Measure time to re-read my data
- Must play tricks to defeat CPU pre-fetch
- Differential coding to resist noise

### Load and traffic estimation

- Check for co-residence using system load as a covert channel
- Estimate traffic load on co-resident web server

### Keystroke timing attack (classic)

- Fine-grained keystroke timing can reveal information about text typed
- Especially given per-user training
- Demonstrated in lab against passwords typed over SSH, without breaking crypto
  - 50× speedup over exhaustive search

### Keystrokes in Xen Lab installation with CPU pinning, otherwise idle; not real EC2 Threshold cache activity level More than idle, less than otherwise busy

- 5% false negatives, 0.3 false positives per second
- Timing resolution 13ms, enough for prior attacks

## Countermeasures: limited Randomize and isolate network structure Timing measurements still possible Block or add noise to covert channels Hard, and how to know you have them all? Avoid locality in placement algorithm Reduces but does not eliminate attacks

### Countermeasure: pay for isolation

- Pay extra to have machines all to yourself
- Argument: fair cost upper-bounded by cost of one physical machine
- Not implemented
  - Though compare: GovCloud