User-based CPU Verification Scheme for Public Cloud Computing

Huanyang Zheng, Kangkang Li, Chiu C. Tan, Jie Wu
CIS, Temple University

Outline

- Motivation
- Monitoring Architecture
- Algorithm Details
- Evaluation

Motivation

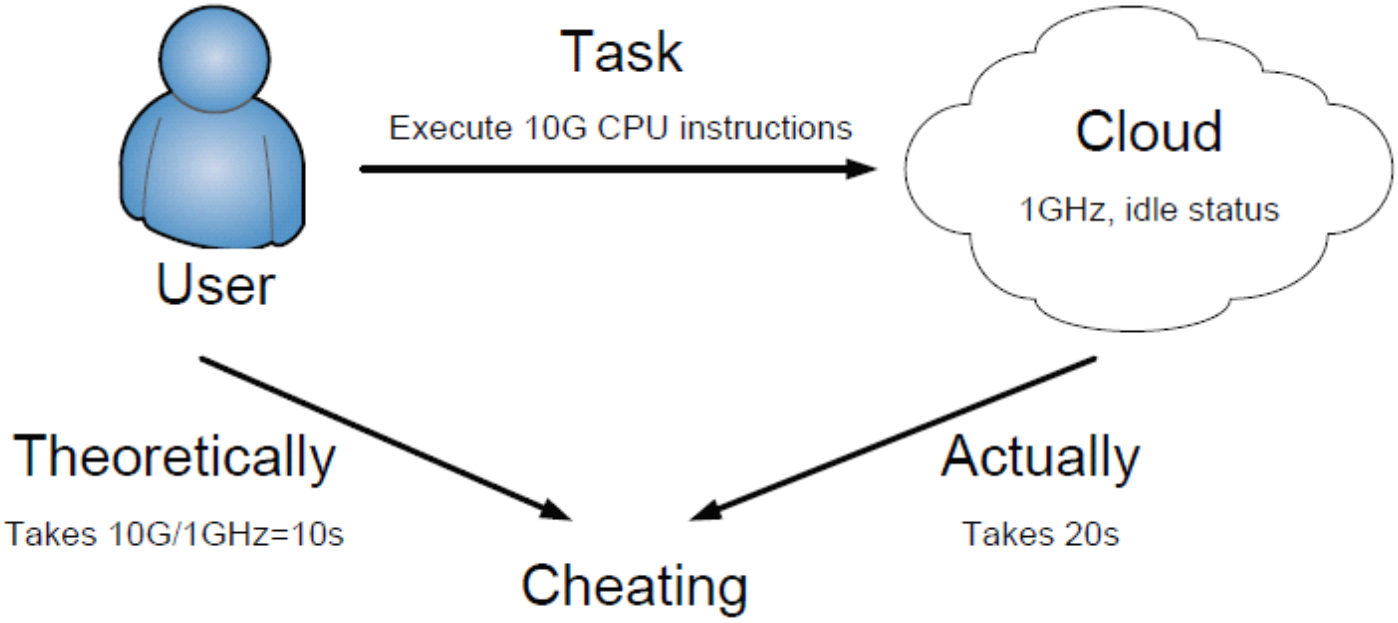
CPU Verification in the cloud system is difficult,

but necessary for the users.

- A malicious cloud allocates overfull VMs to a PM, as to save the operation costs of additional VMs.
- Errors in the VM migration code or algorithm, and the heterogeneity of hardware also result in the fluctuation of the CPUs.

Monitoring Architecture

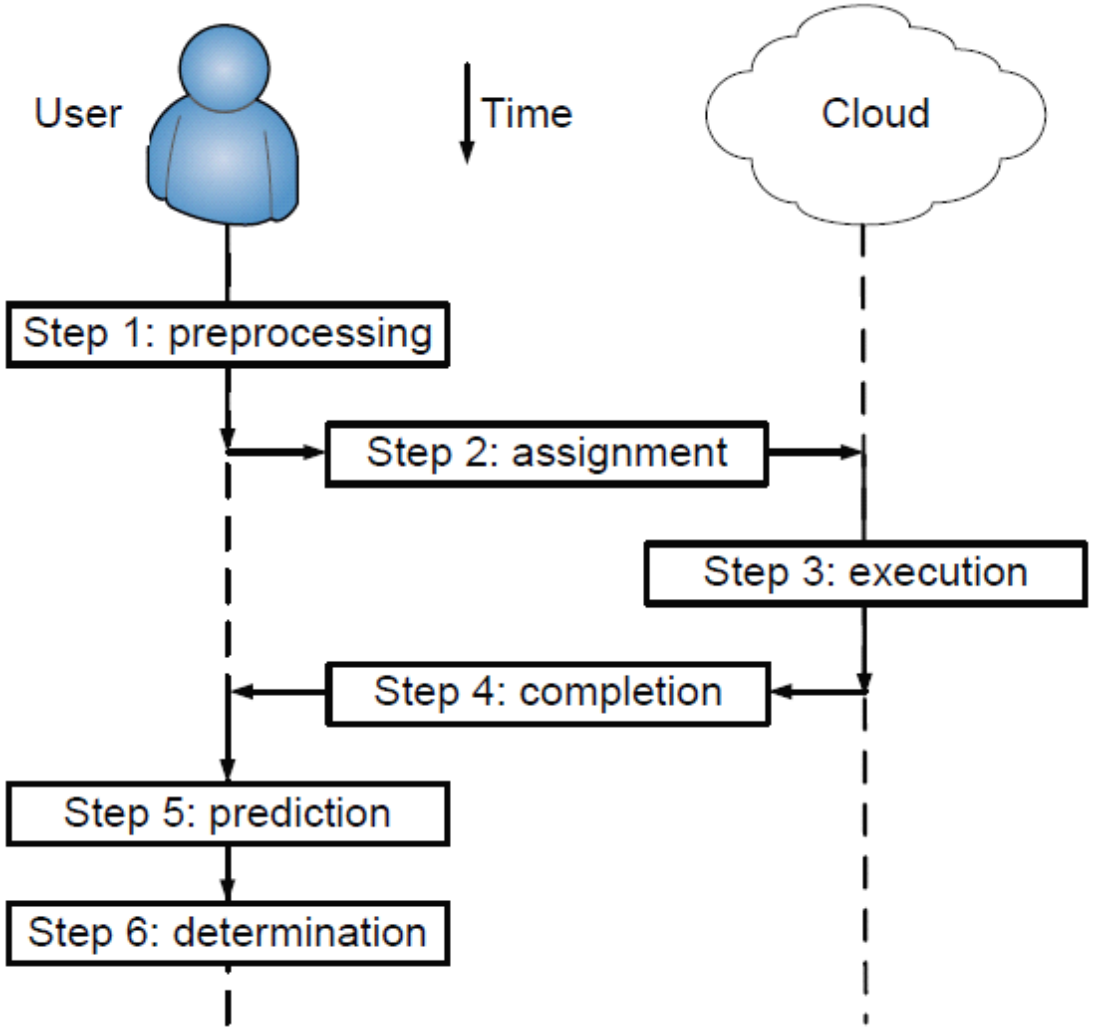
Assign a predefined task for the cloud, and check the difference between the theoretical execution time



Monitoring Architecture

Assum

- The ng
  any
- The
  tim
- Sm elay,
  are



Algorithm Details

The predefined task for cloud execution:

- Guarantee the execution of certain CPU instructions. For example, if the task is to calculate x=x+1 for 1000 times, the cloud can calculate x=x+1000 for one time instead, as to save calculation time.

Algorithm Details

The predefined task for cloud execution:

- Use a time-lock puzzle to guarantee execution.

Theorem 1 (Time-Lock Puzzle Theorem): Assume a large number b is relatively prime to a large composite number n, without factoring n; the quickest method to solve $b^{2^M} \bmod n$ (M is an arbitrary natural number) is to loop $b=b^2 \bmod n$ for M times (returns b as the outcome).

Algorithm Details

Theoretical task execution time:

- Stop everything and decide that the cheating detection is ***not*** rational, since the users buy the cloud for temporal computation, rather than doing cheating detection.

Algorithm Details

Theoretical task execution time:

- If our cheating detection program takes 40% CPU when running alone, and currently 80% CPU of the VM is taken off, then how much CPU would the detection program take ?
- Depends on the OS schedule. But in most OSs, it would take 40%/(40%+80%) = 1/3 CPU.

Algorithm Details

Cheating determination:

- The difference between the theoretical execution time and the actual execution time is larger than a certain threshold.

Algorithm Details

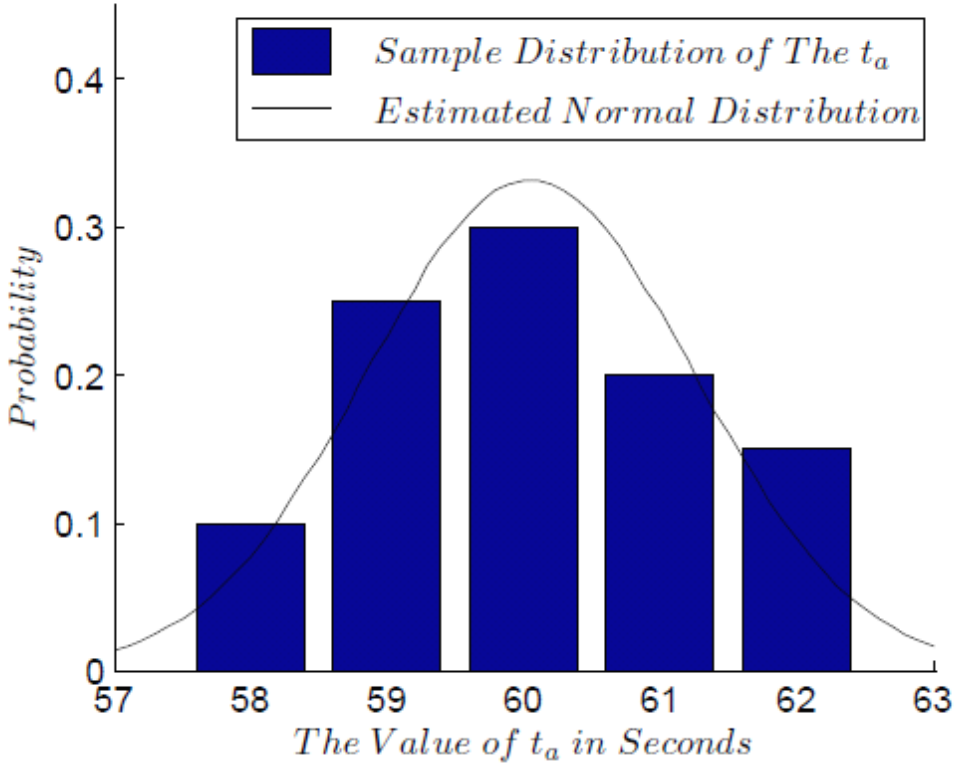
Small execution task vs. Large execution task

- The resources for cheating detection are limited.
- A smaller task means less precision in one round of cheating detection, but more rounds.
- A larger task means higher precision in one round of cheating detection, but fewer rounds.

Algorithm Details

Small execution task vs. Large execution task

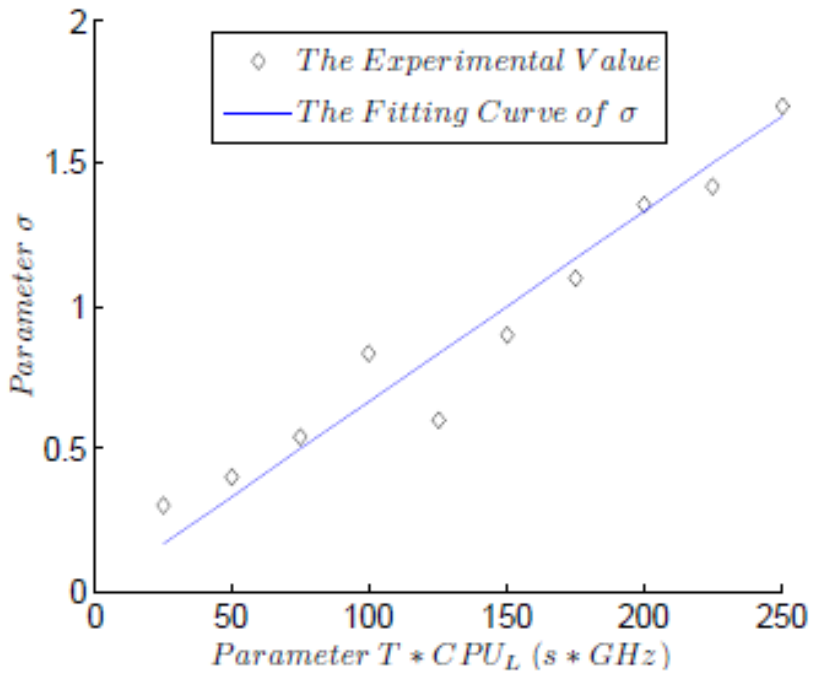
- Exper execution time i n.



Algorithm Details

Small execution task vs. Large execution task

- The dist proportional to the tc ask.



Algorithm Details

Small execution task vs. Large execution task

- Theoretical model shows that a smaller task and more detection rounds are better.
- However, the task cannot be infinitely small, since the interferences are no longer negligible.

Evaluation

System setup

- Based on Oracle VM VirtualBox, version 4.1.22.
- The virtualization technology is essentially the same as what it is in the cloud system.

Evaluation

Memory-intensive test

- Our detection method requires very small

$Memory$	t_t	μ of t_a	σ of t_a
$512MB$	60	62.4	8.27
$640MB$	60	60.5	1.29
$2GB$	60	60.2	1.20

Thank you !

Q & A