Energy Attack on Server Systems

Zhenyu Wu, Mengjun Xie† and Haining Wang
Department of Computer Science
College of William and Mary

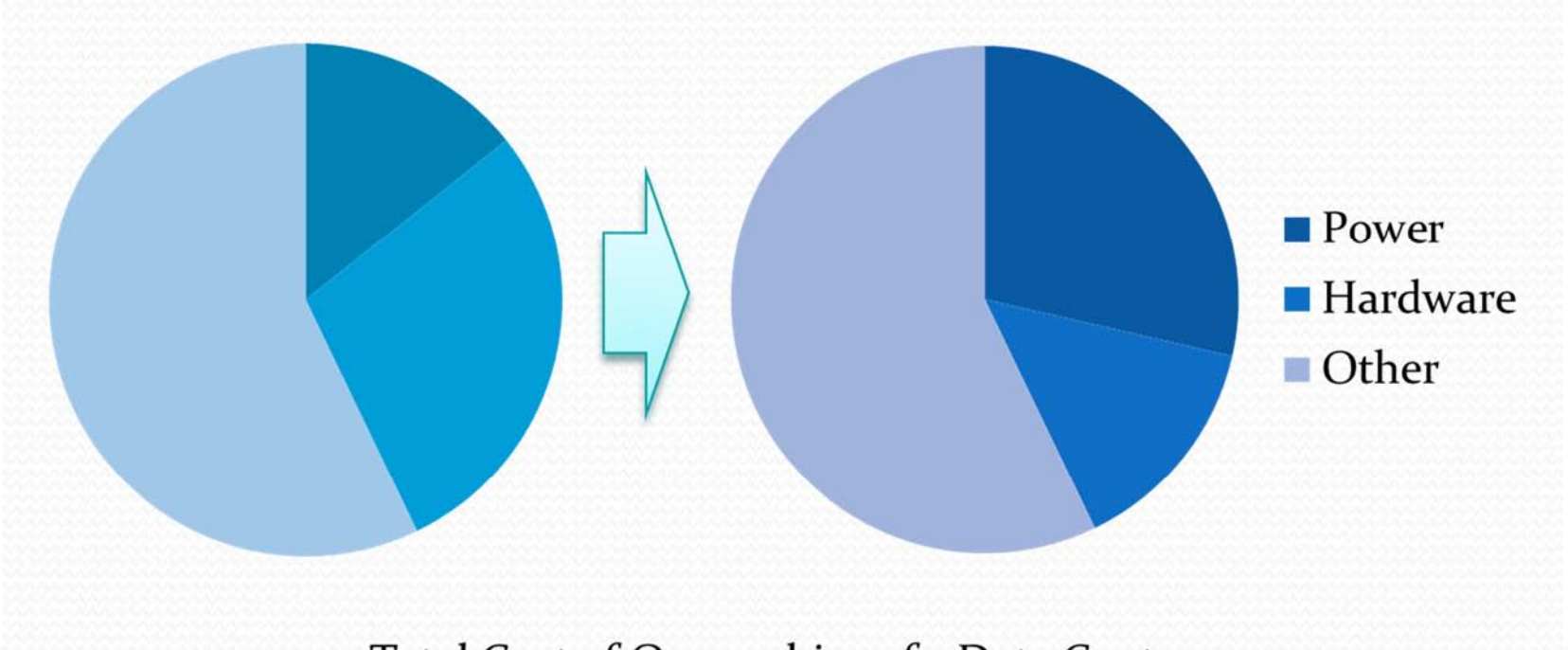
Presenter: Zhenyu Wu

The College of
WILLIAM & MARY

† Currently affiliated with
University of Arkansas at Little Rock

Energy and Green Computing

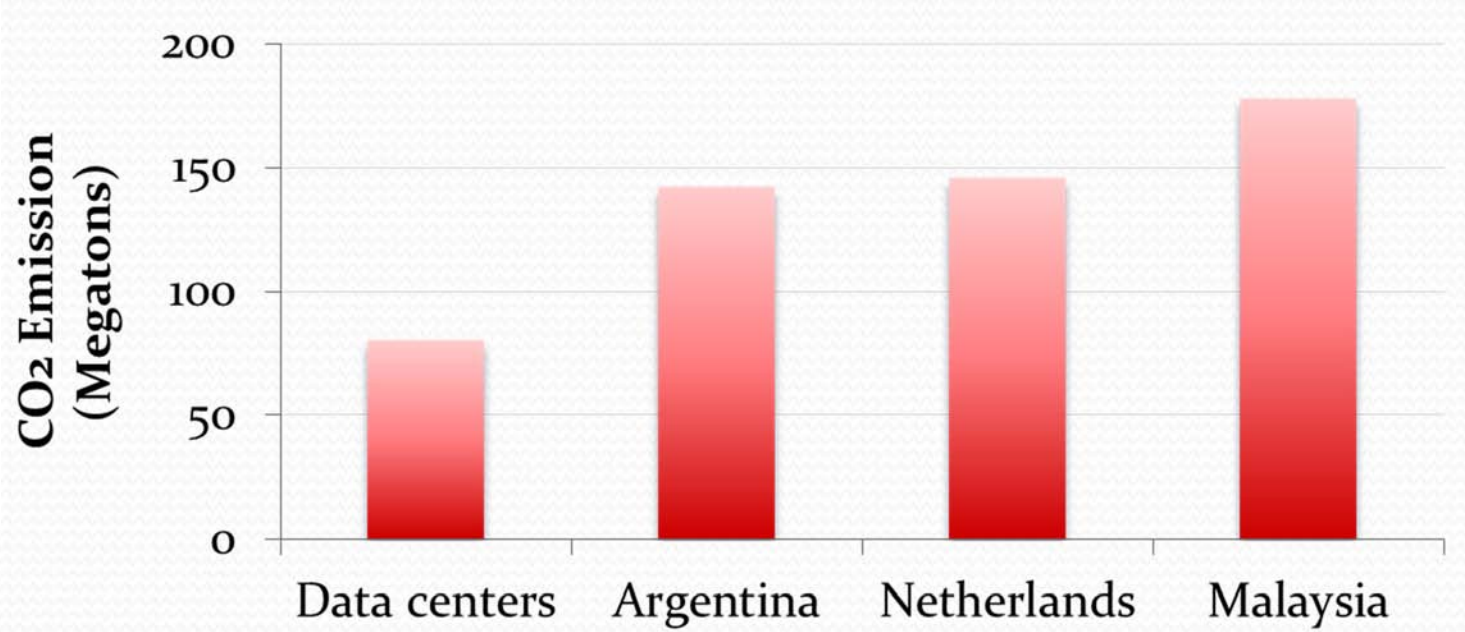
- Energy cost has become a major factor in the total cost of ownership (TCO) of large-scale server clusters



Total Cost of Ownerships of a Data Center

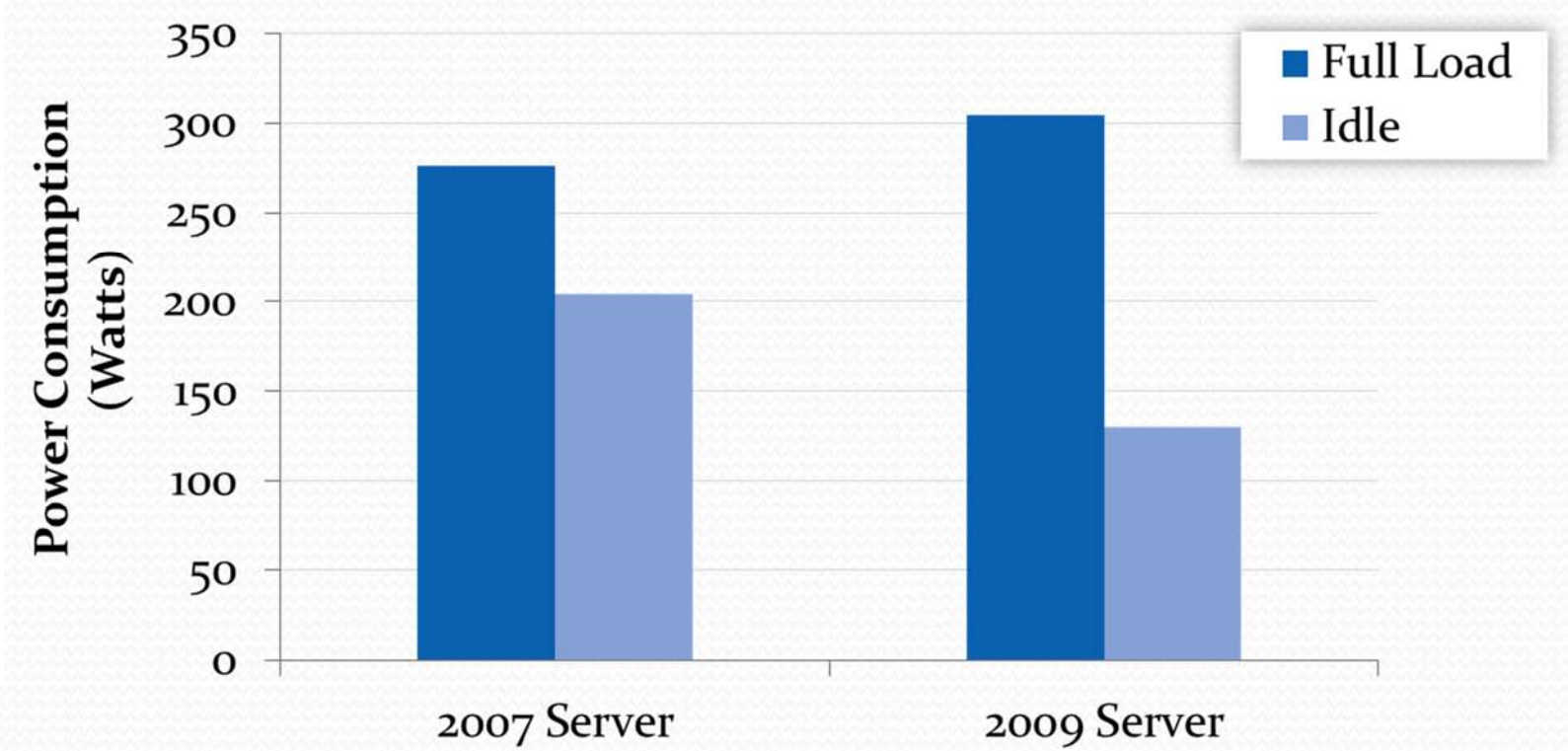
Energy and Green Computing

- Millions of tons of carbon-dioxide are generated in order of power data centers
  - Two Google searches = boiling a cup of coffee
  - Global data center carbon emission (2007)



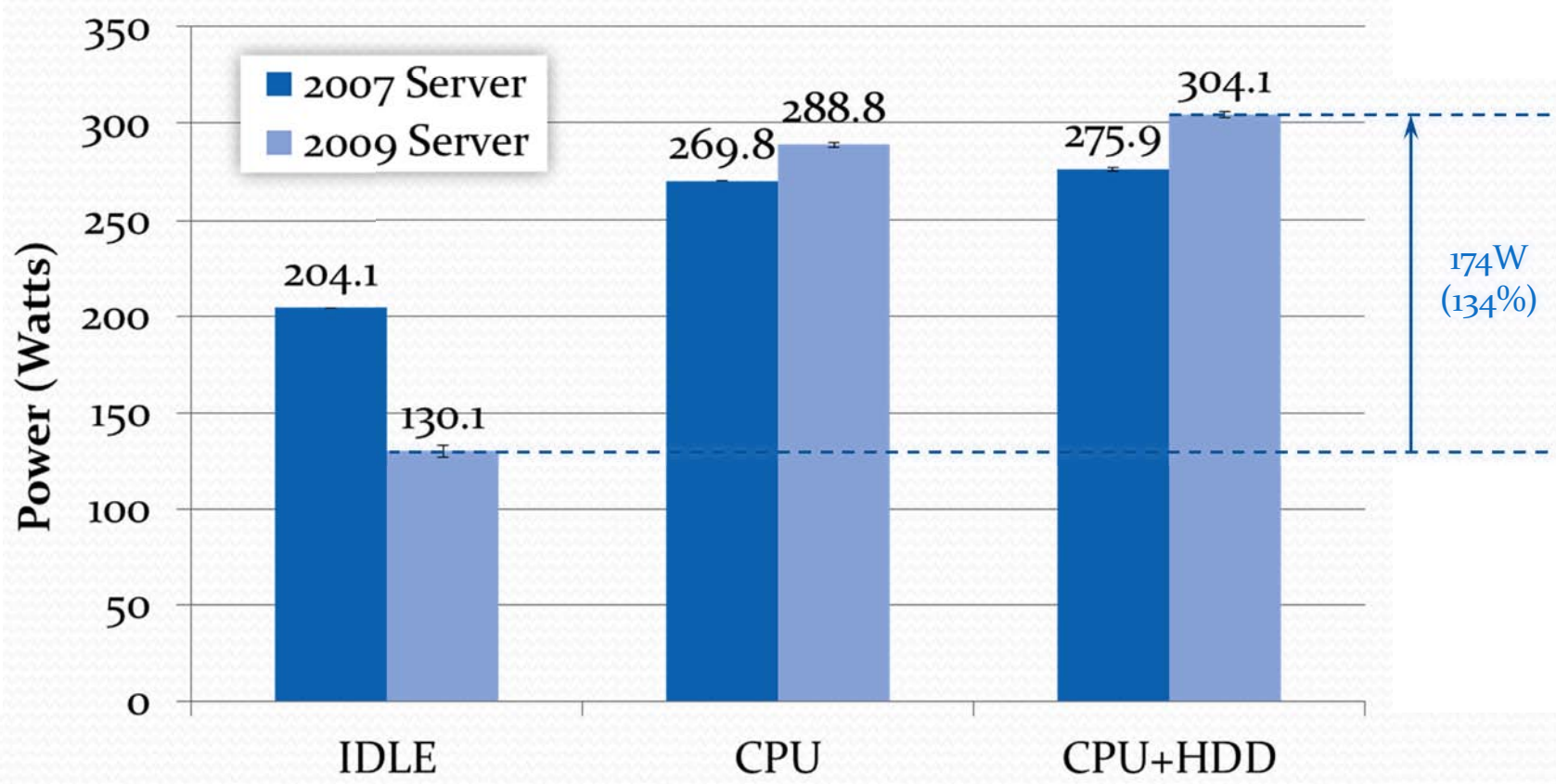
Energy Proportional Computing

- Aims to make servers consume energy proportional to its workload.



Energy Proportional Computing

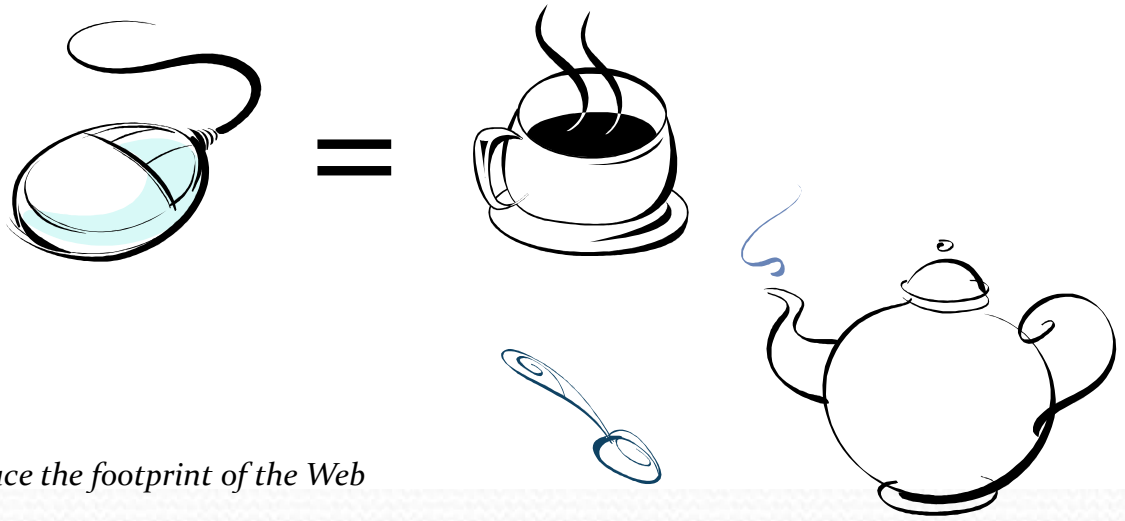
- Power usage break down:



The College of
WILLIAM & MARY

Potential Vulnerabilities

- Energy efficient computing assumes a cooperative working environment
  - Power saving is passive, dependent on workload
  - Not all workload consumes identical amount of energy



Alex Wissner-Gross,
How you can help reduce the footprint of the Web

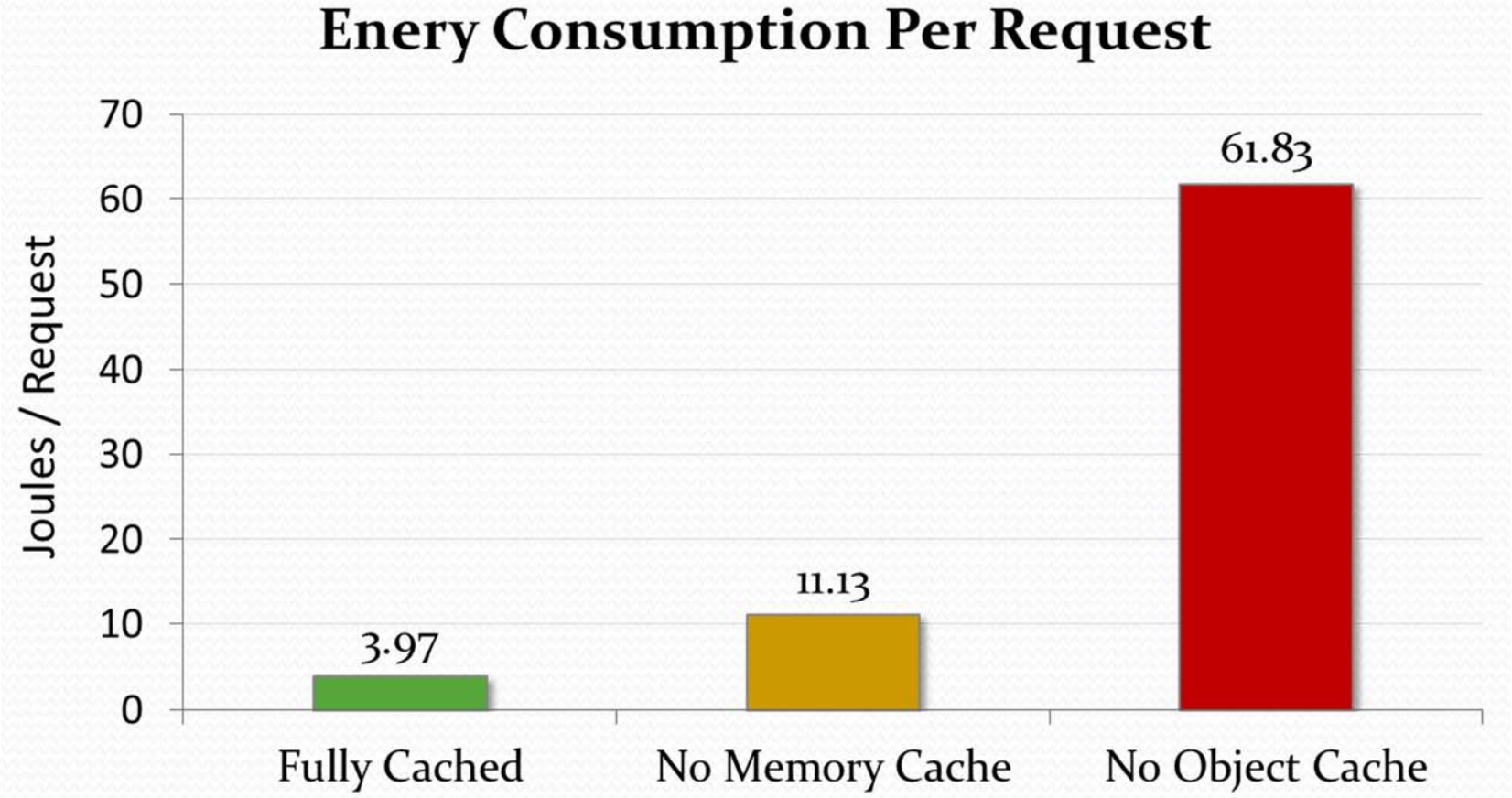
Formulating Attack Vectors

- Attack Vector:
  - Isolate high energy cost requests
    - Analyze the triggering conditions
  - Reproduce in high concentration
    - High percentages, but no necessarily large amounts
- Vulnerable systems: open services, such as search engine, knowledge base, public forum, etc.
  - Have little or no control over the incoming request
  - Energy consumption is largely dependent on the type and amount of service requests

Designing an Energy Attack

- We use a Wikipedia mirror server as the victim
  - Publicly available large scale database
  - Representative of standard open Internet services
- We discover the attack vector by profiling the server
  - Powered by MediaWiki, a large scale content management system.
  - Two levels of caching for efficient operation
    - Object Cache – for dynamically generated pages
    - Memory Cache – for recently executed database queries

Designing an Energy Attack



The College of
WILLIAM & MARY

Designing an Energy Attack

- Keys to launching the energy attack:
  - Generate Cache Misses
    - Much higher energy/request than normal workload
  - Avoid Generating Anomalies
    - Be low profile, non-obtrusive
    - Must not generate traffic anomaly
    - Must not cause obvious performance degradation

Designing an Energy Attack

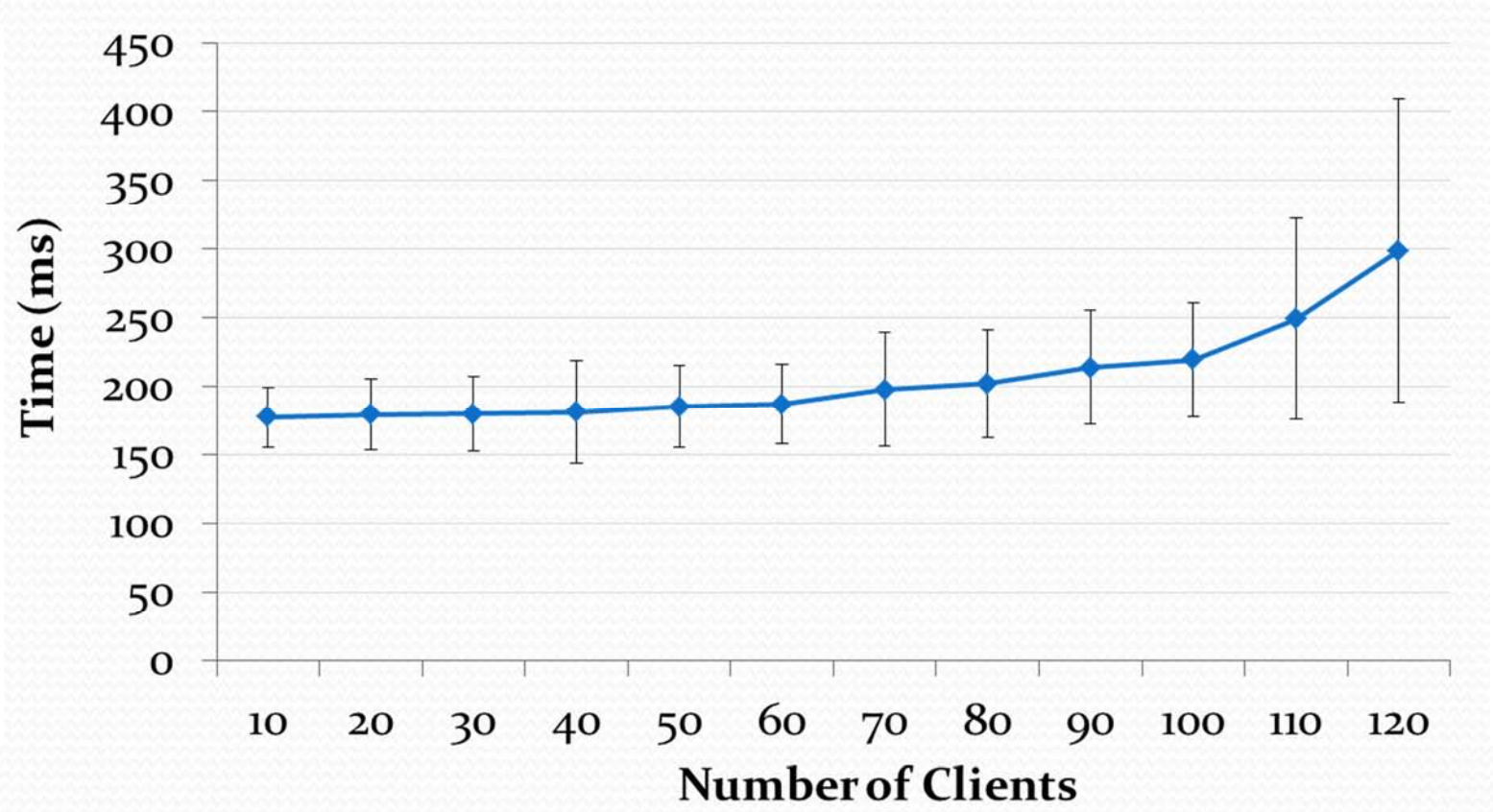
- Website access profiling
  - Cache Misses:
    - The frequency of a web page being accessed is inversely proportional to its rank (Zipf's law)
    - A small number of web pages are accessed very frequently
      A large number of web pages are accessed very infrequently
    - Different access patterns = **Cold** pages = Cache misses
  - Stealthiness:
    - The request inter-arrival time of human users follow Pareto distribution
    - The attackers can *mimic* normal users by sending requests at average rates, and following Pareto distribution

Measurements and Evaluations

- Server Configurations:
  - Dual Intel Xeon 5540 quad-core processor
  - 6GB DDR3 SDRAM
  - 2TB SATA HDD in RAID 5
  - Power usage monitored by Watts Up PRO power meter
- Experiment Methodology
  - The server is able to stably support accesses from up to 100 benign clients.
  - At different benign workloads (5~100 clients), we launch attack with varying intensity
    - Measure the increase in power consumption √ and latency ✗

Measurements and Evaluations

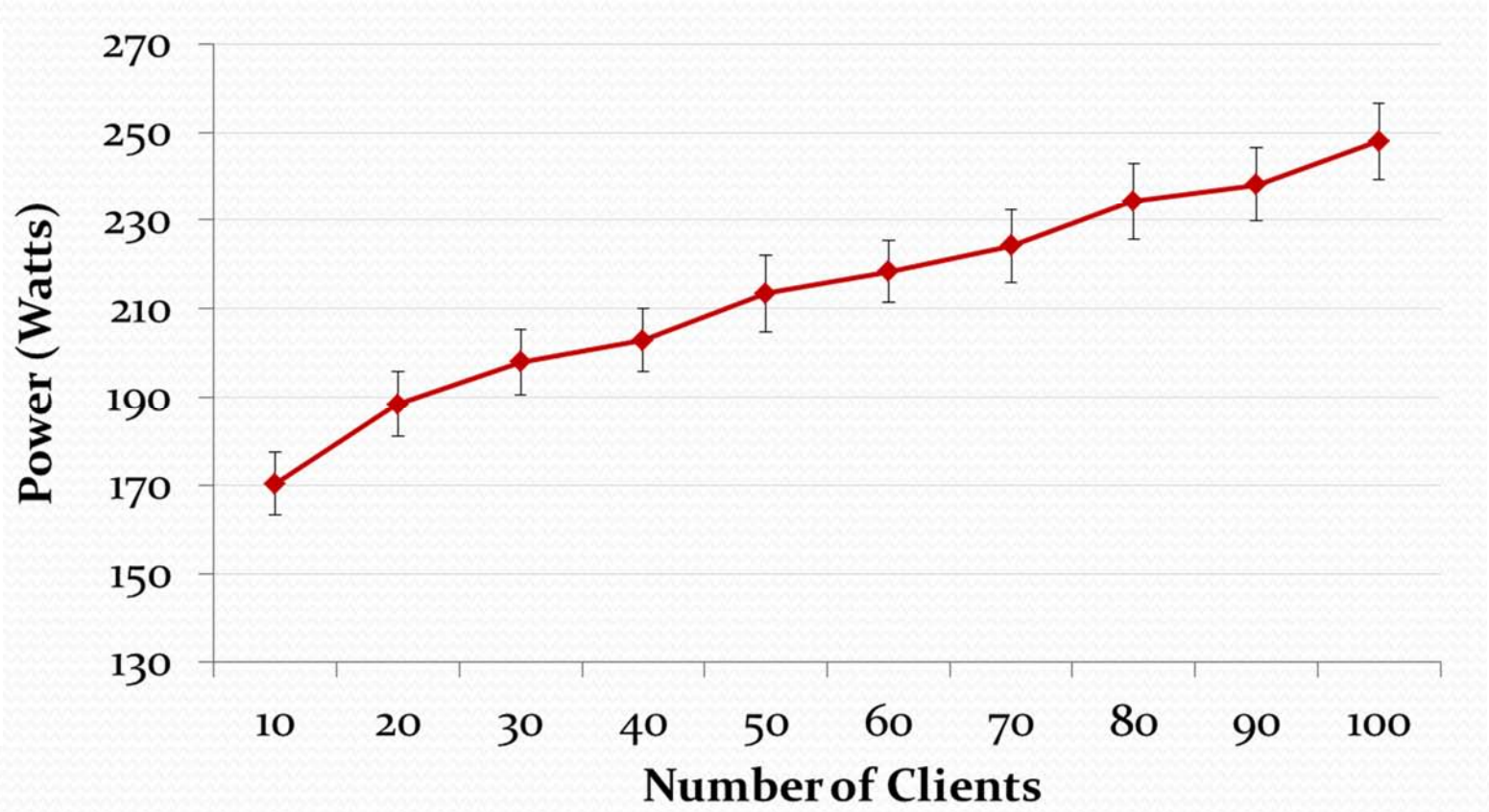
- Workload – Response Time Profile (Normal)



The College of
WILLIAM & MARY

Measurements and Evaluations

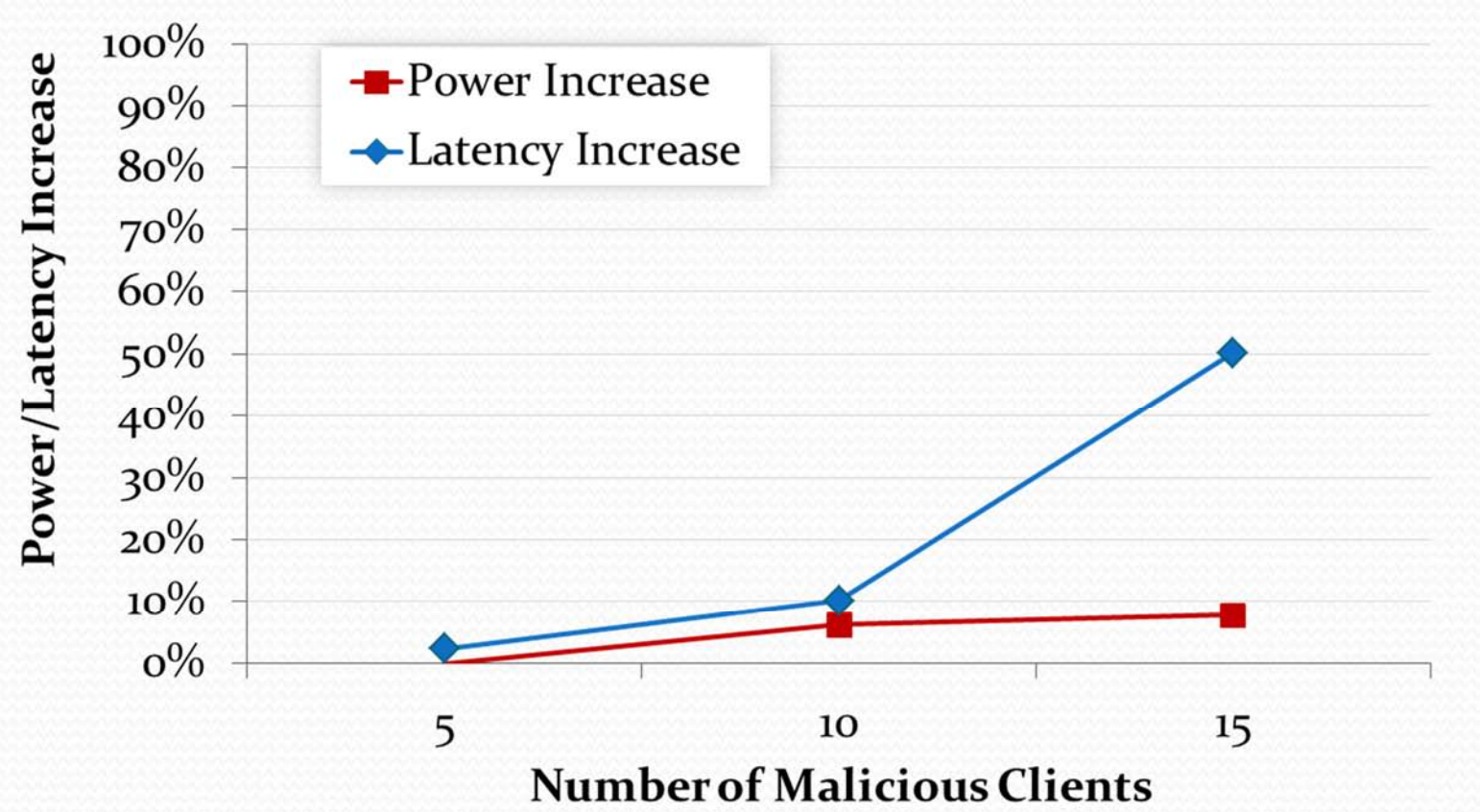
- Workload – Power Consumption Profile (Normal)



The College of
WILLIAM & MARY

Measurements and Evaluations

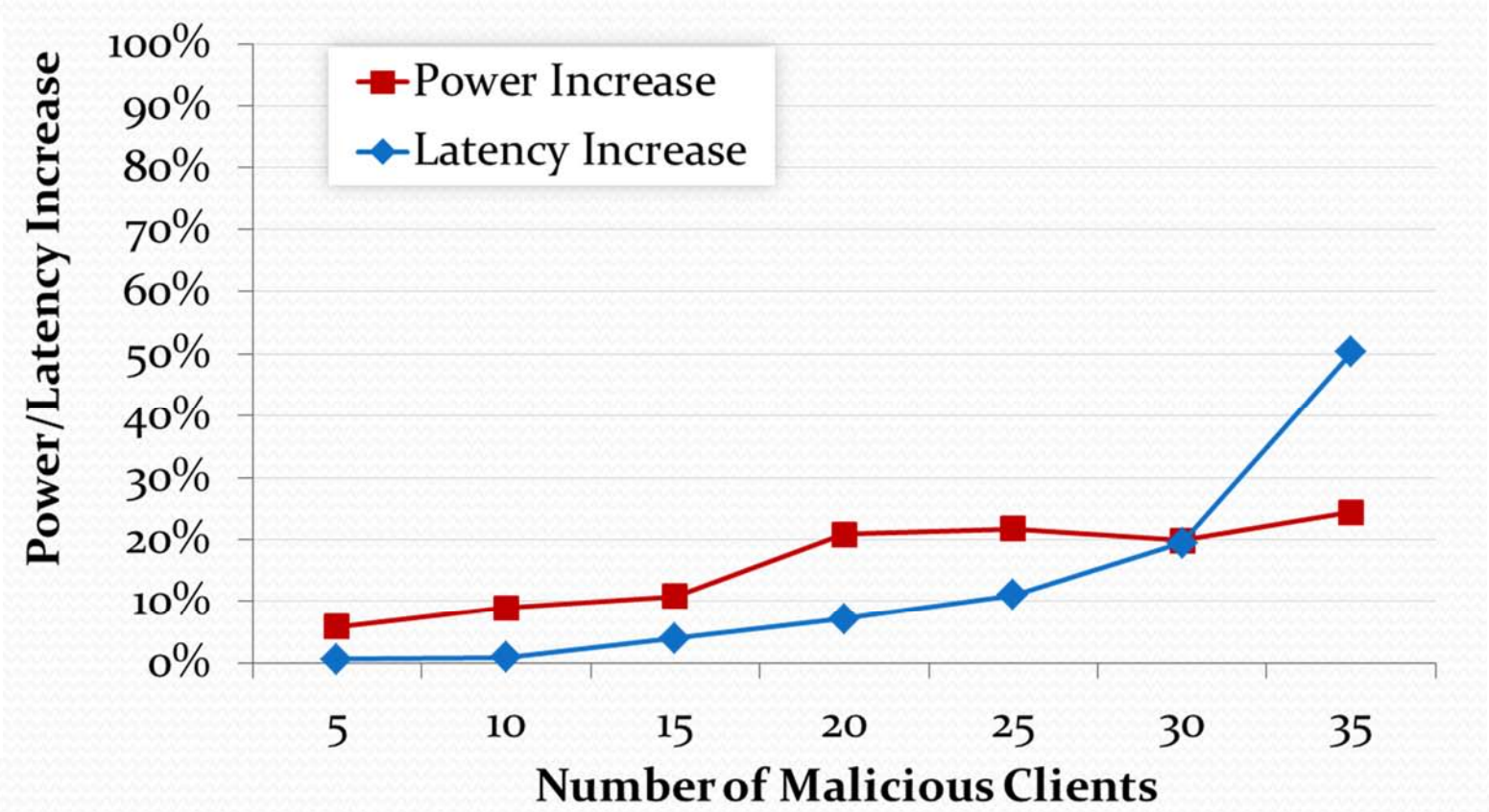
- Power vs. Latency Increase at high workloads (100 clients)



The College of WILLIAM & MARY

Measurements and Evaluations

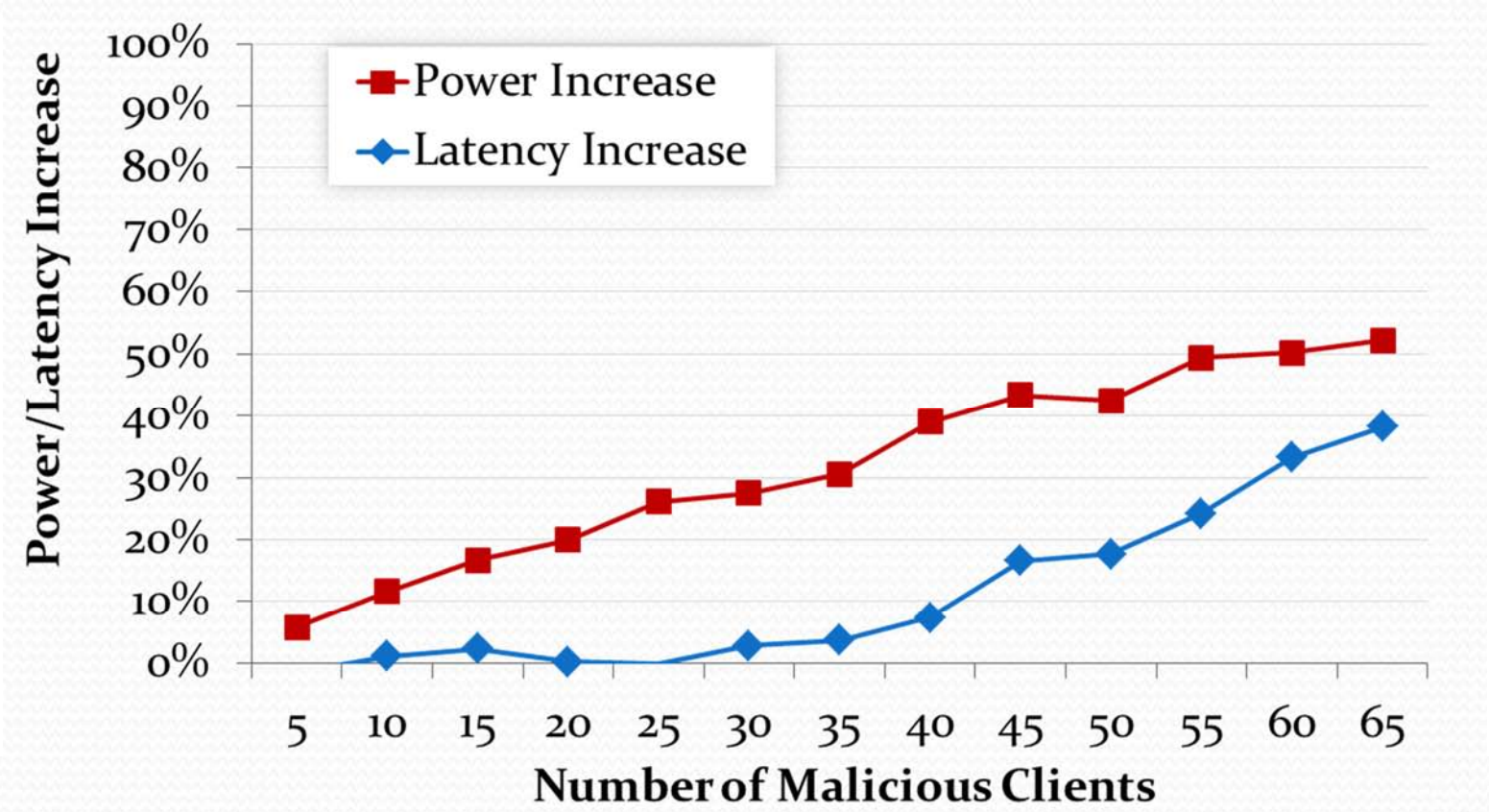
- Power vs. Latency Increase at medium workloads (50 clients)



The College of
WILLIAM & MARY

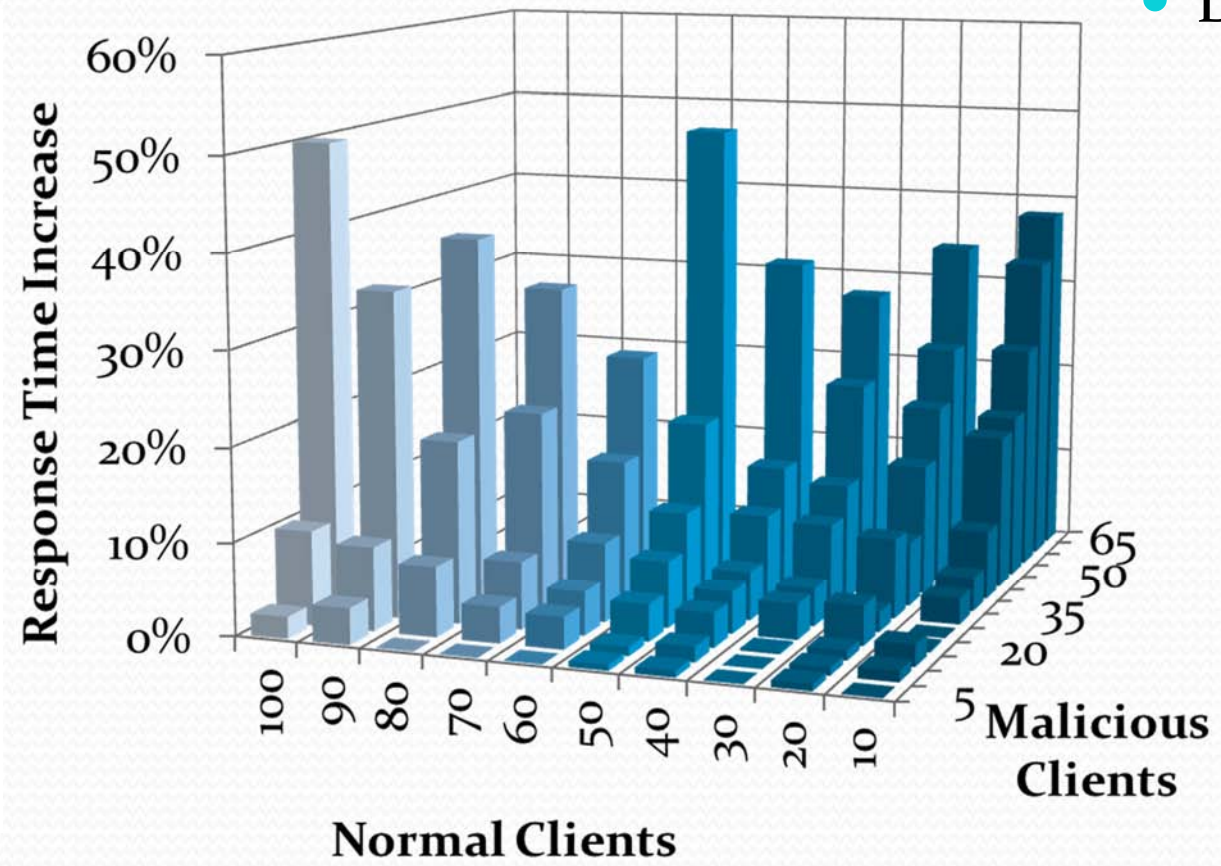
Measurements and Evaluations

- Power vs. Latency Increase at low workloads (10 clients)



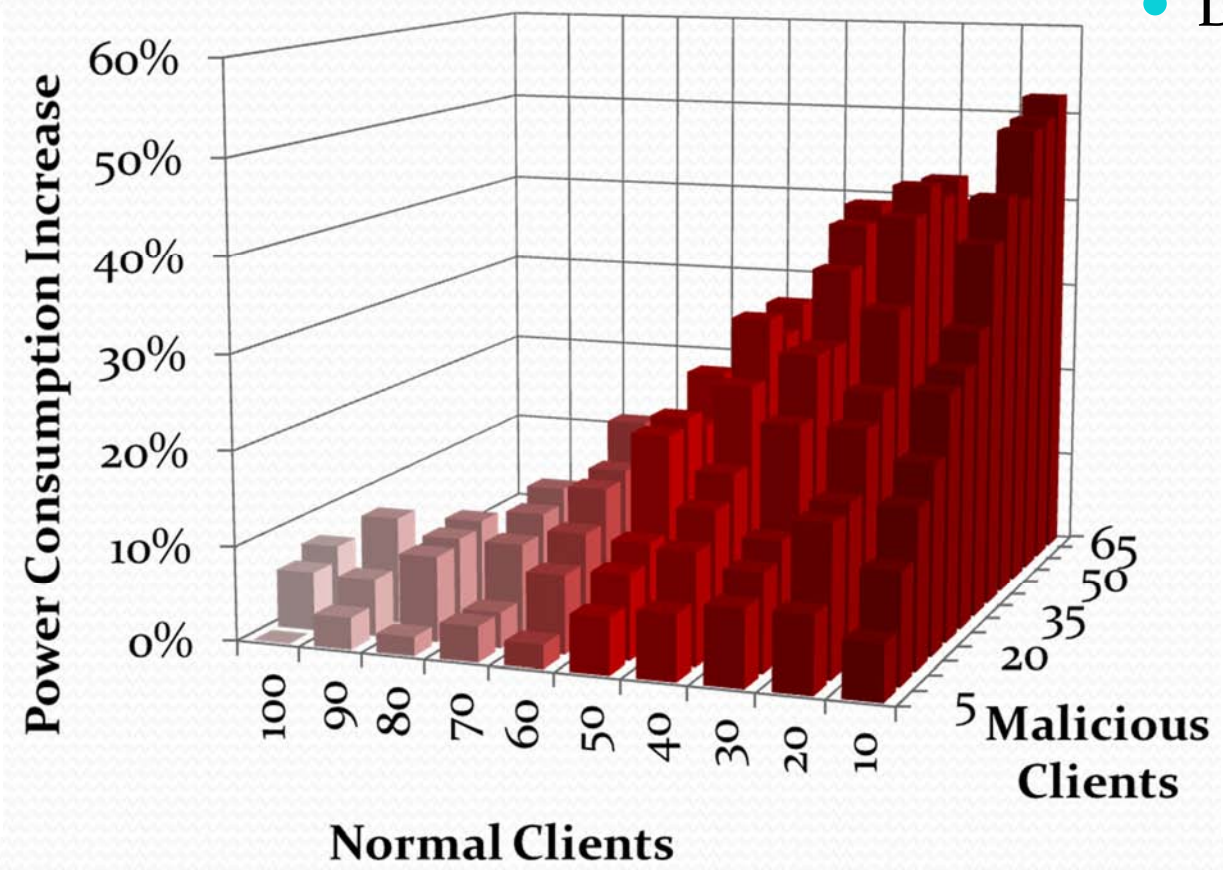
The College of WILLIAM & MARY

Measurements and Evaluations



- Damage achieved:
  - 6.2% ~ 42.3% additional power usage, depending on workload.
  - For typical server workloads: 21.7% ~ 42.3% power wastage

Measurements and Evaluations



- Damage achieved:
  - 6.2% ~ 42.3% additional power usage, depending on workload.
  - For typical server workloads: 21.7% ~ 42.3% power wastage

Alternative Energy Attack Vectors

- Algorithmic Complexity Attacks
  - Algorithms that have high worst-case run time
    - Plain quick sort, naïve hash-table, etc.
  - Originally proposed as DoS attacks, can be adapted to use as energy attacks
    - Processors are the most power consuming devices
    - Be stealthy: lower intensity, target non-computation intensive servers (such as file depositing services)

Alternative Energy Attack Vectors

- Example:
  - Linux directory cache vulnerability
    - Simple hash for quick file name lookup
    - Vulnerable to collision attack
  - FTP server
    - Setup: upload thousands of files with colliding names
    - Attack: download, rename, read/write metadata, etc.

Alternative Energy Attack Vectors

- Sleep Deprivation Attacks
  - Originally proposed as DoS attacks in sensor network, can be adapted to use as energy attacks
    - Target components that have large dynamic power range
    - Doesn't require high per-unit power consumption
- Example:
  - A hard drive consumes 12~16 watts of power in operational states, but only ~1 watt in spin-down
    - File servers usually have tens of hard drives!
    - Malicious access patterns can interfere with power management and prevent expected spin-down

Challenge of Defense

- The key is still missing:
  - What we want to do
    - Differentiate high energy cost workload
  - What we have at hand
    - Coarse grained power measurement instrument
      - "We are under attack! …
        … And we have to suck it."
    - Fine grained performance counters (approximation)
      - Good for single task systems (mobile phone / PDA / etc.)
      - Incompetent for highly parallel environment
  - What we really need:
    - Fine grained power measurement support

Future Work

- Extend beyond single server
  - Server-clusters, server farms
  - Data center, massively virtualized environment
  - Etc.
- Explore software-based countermeasures
  - Temporary workarounds to the lack of hardware support
  - Explore possibility of inferring workload natures from application behavior profiling