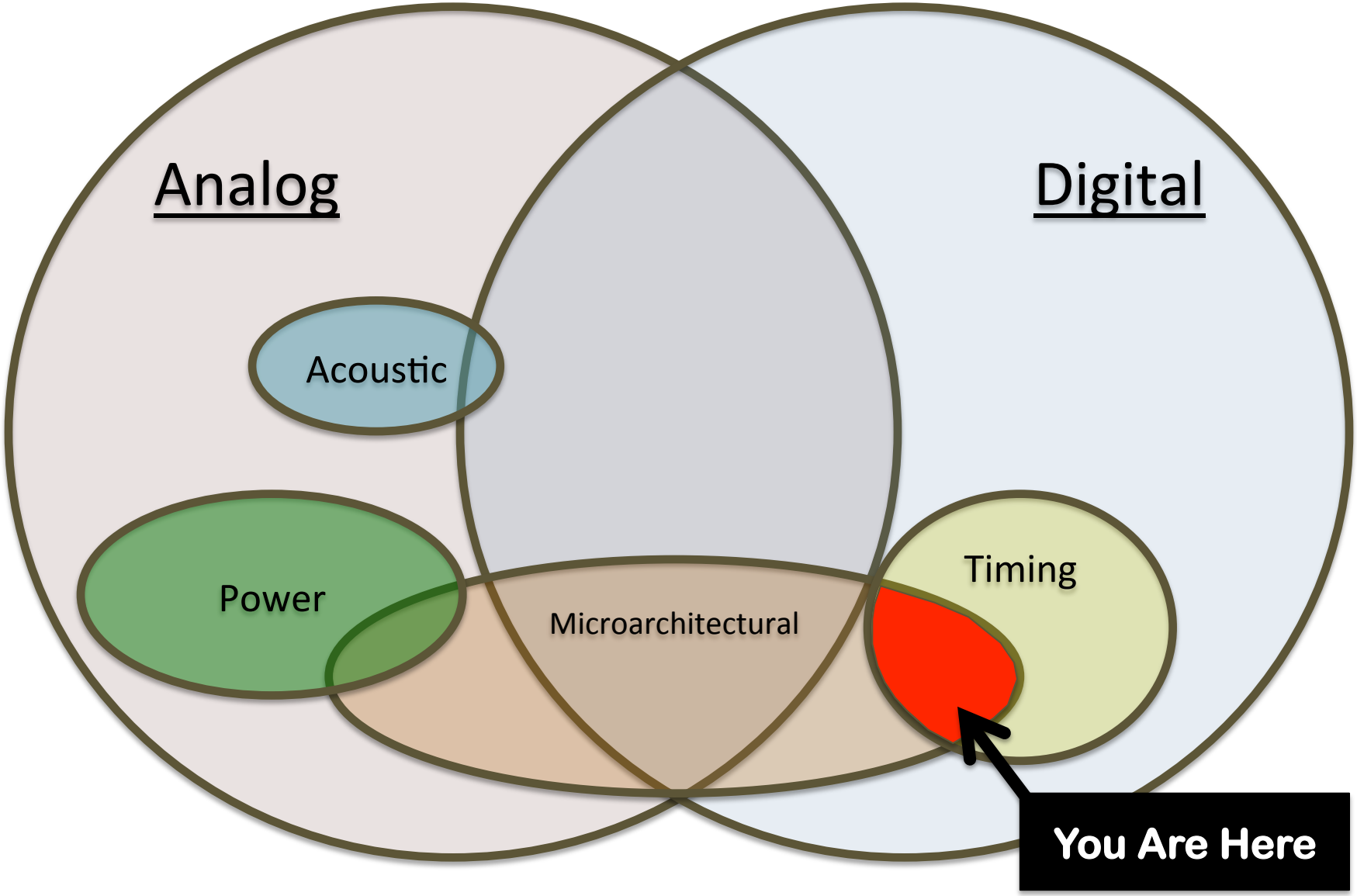
TimeWarp

Mitigating Microarchitectural Side-channels

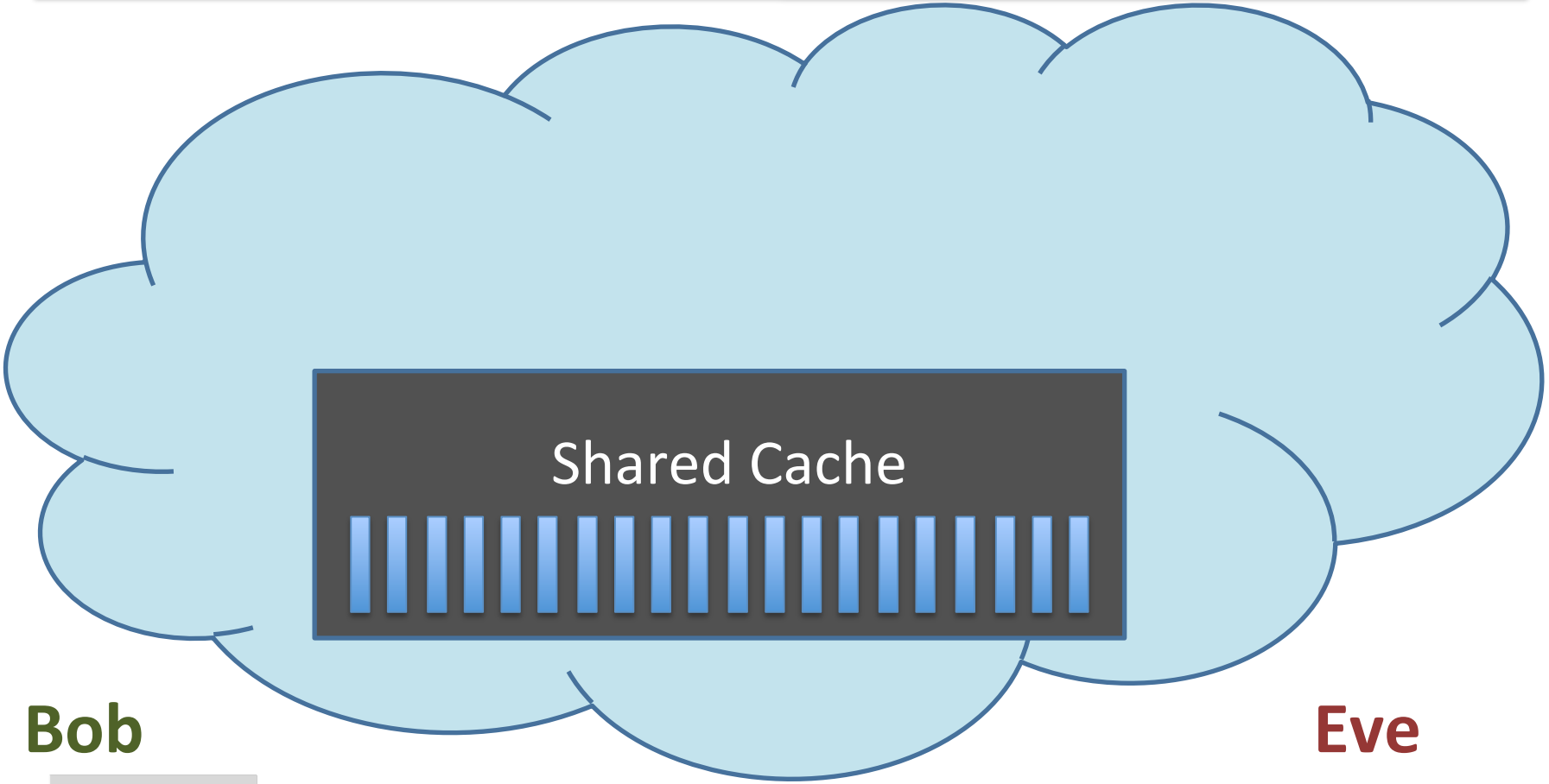
Robert Martin
John Demme
Simha Sethumadhavan

Computer Architecture and Security Technologies Lab
Columbia University

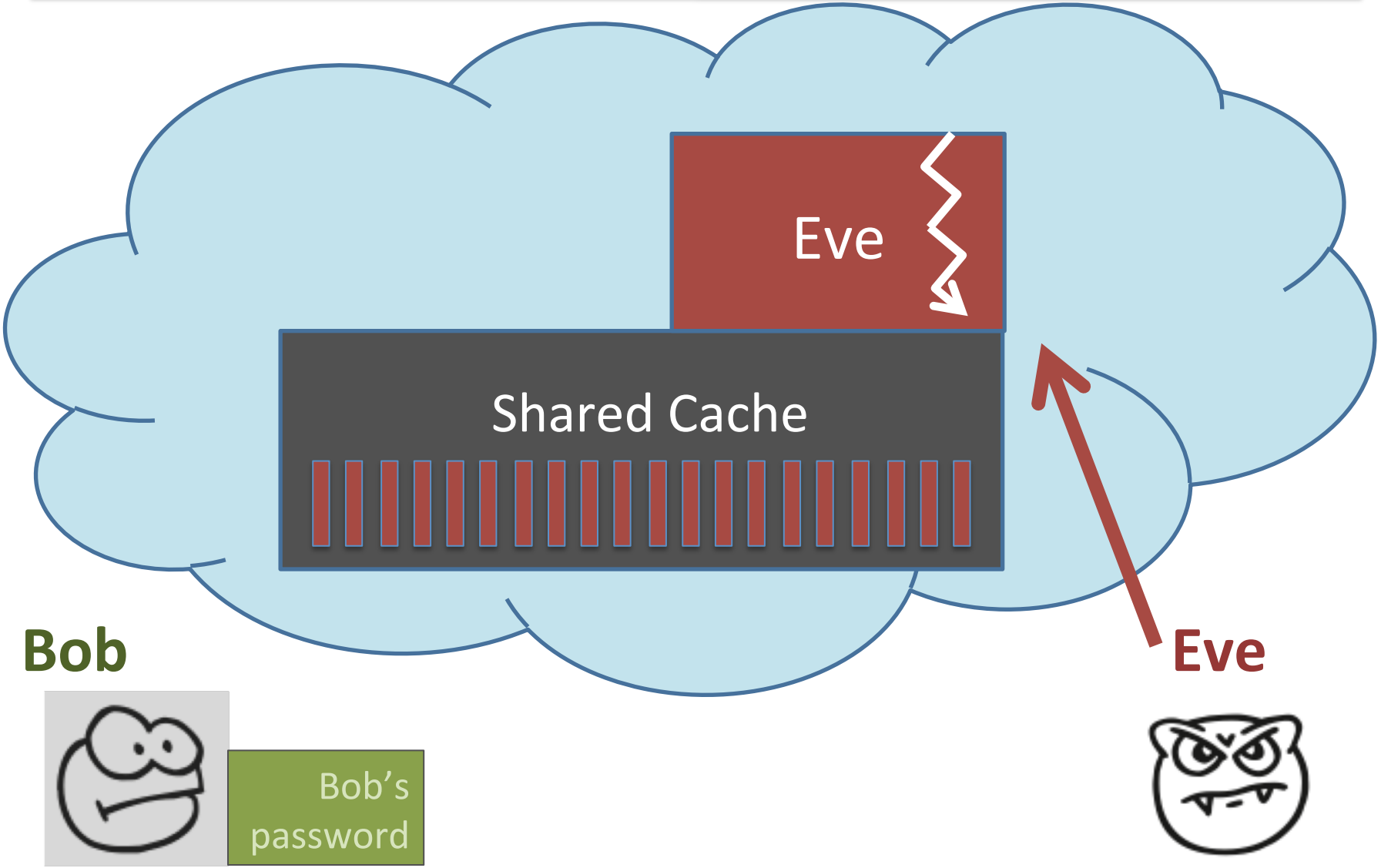
Side Channel Attacks



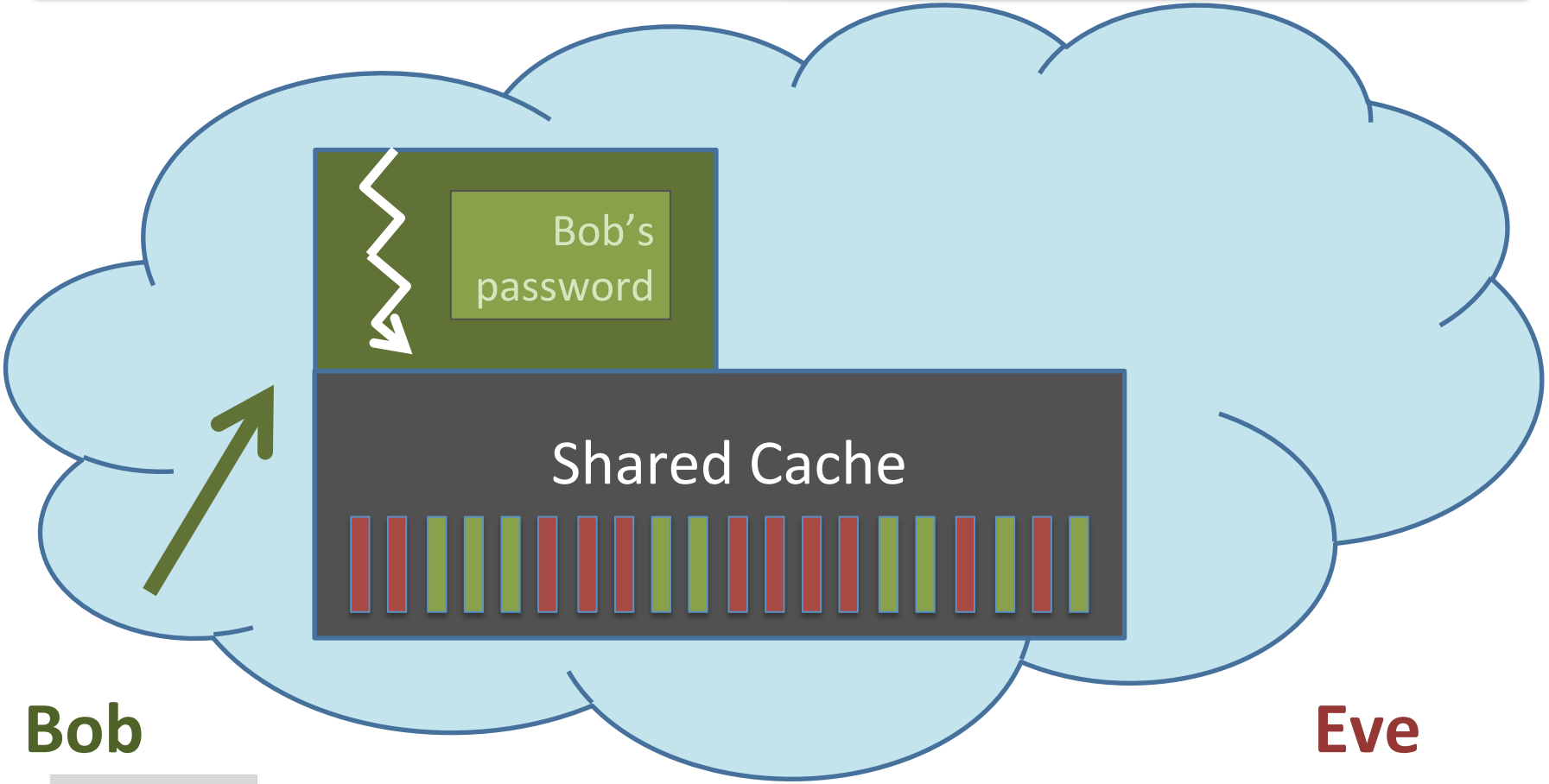
A Sample Microarchitectural Attack



A Sample Microarchitectural Attack



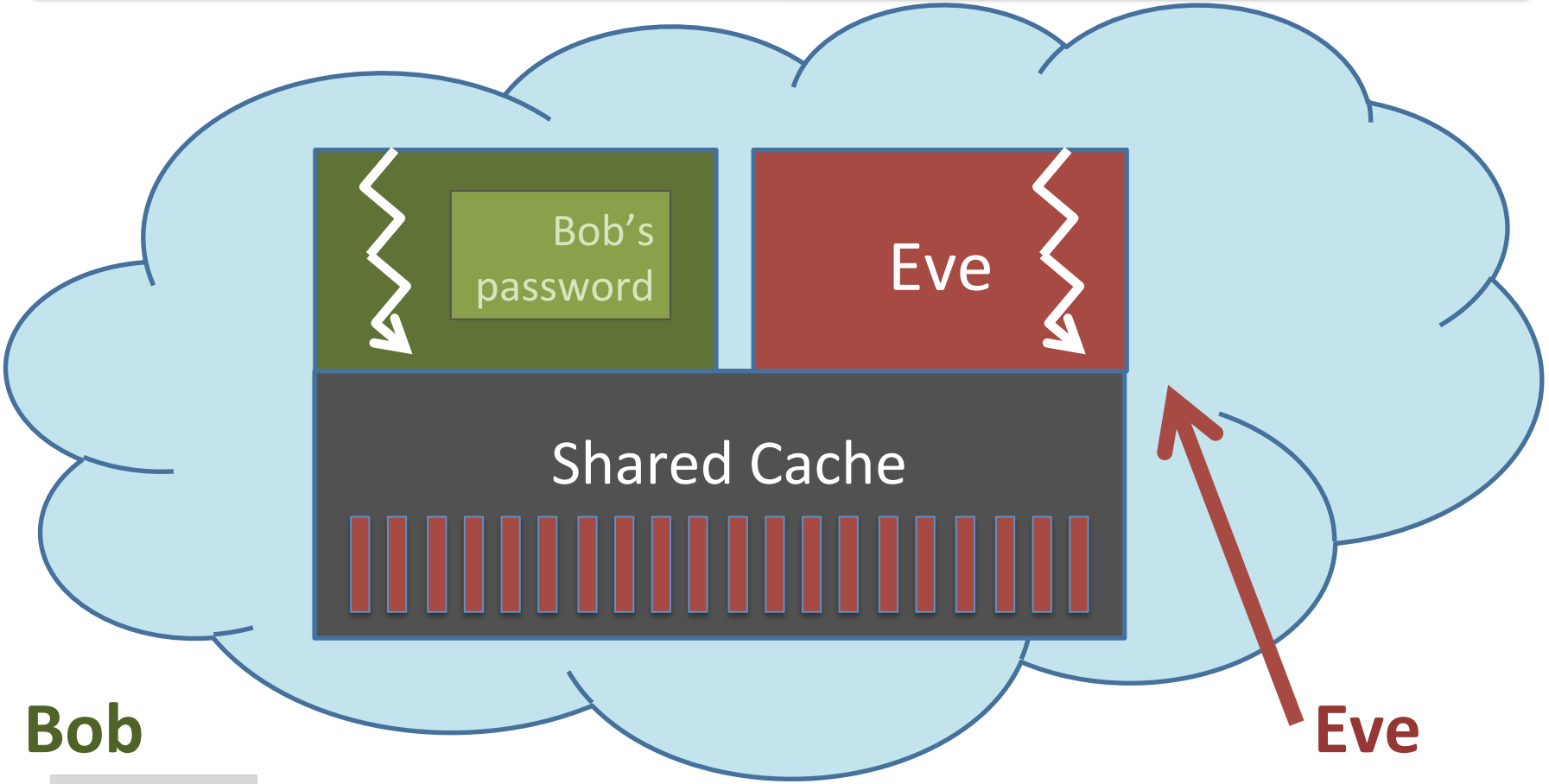
A Sample Microarchitectural Attack



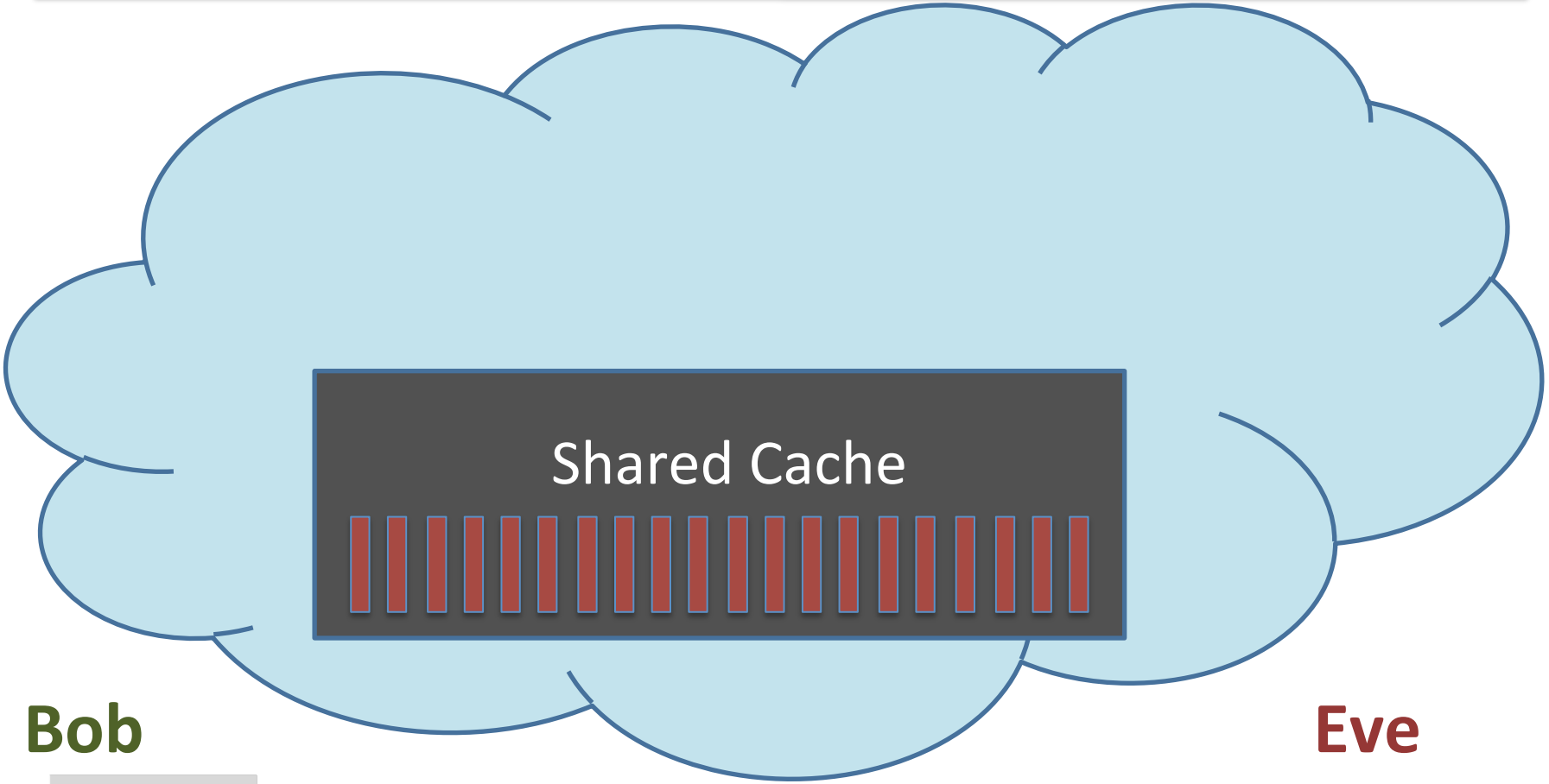
Bob

Eve

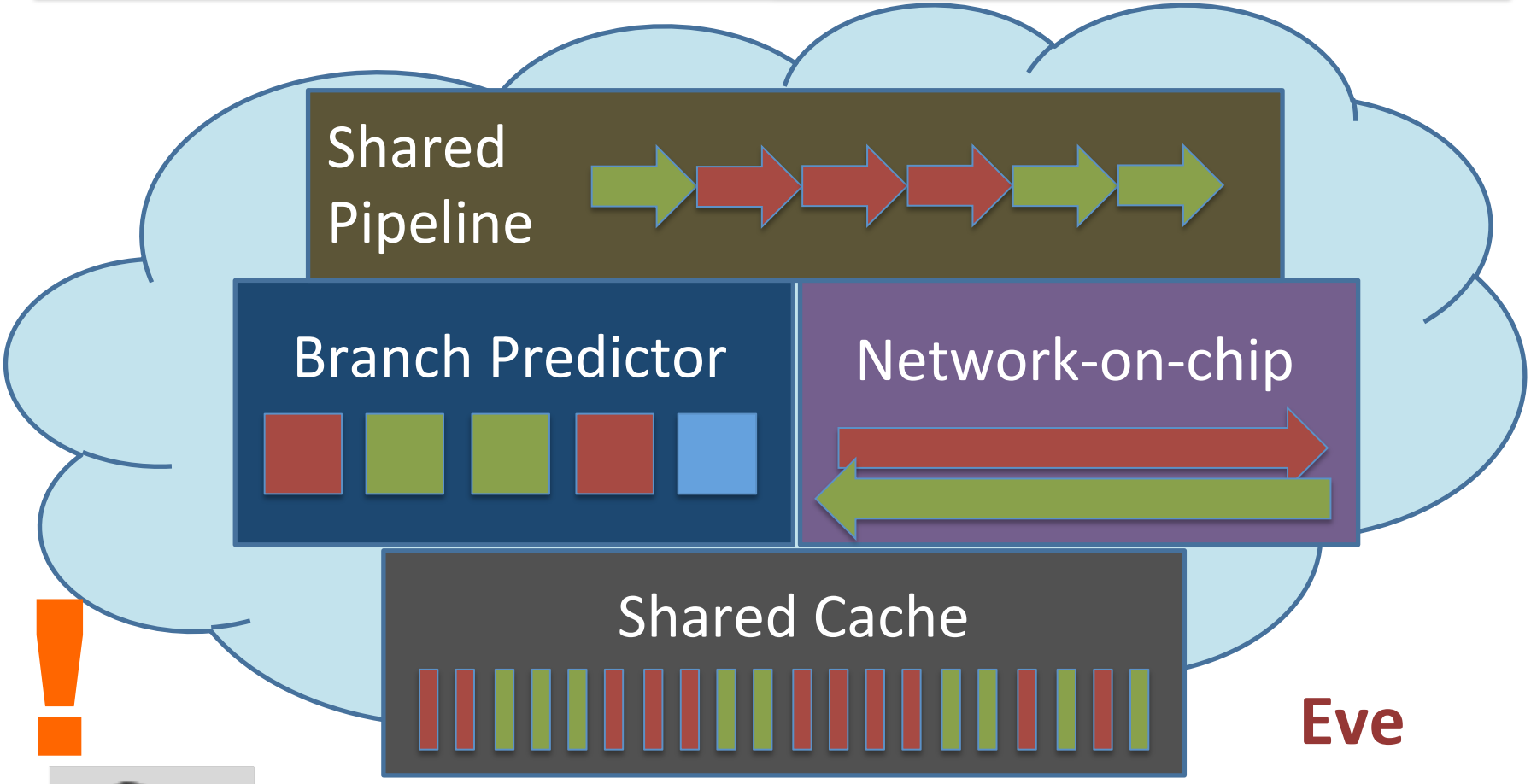
A Sample Microarchitectural Attack



A Sample Microarchitectural Attack



A Sample Microarchitectural Attack



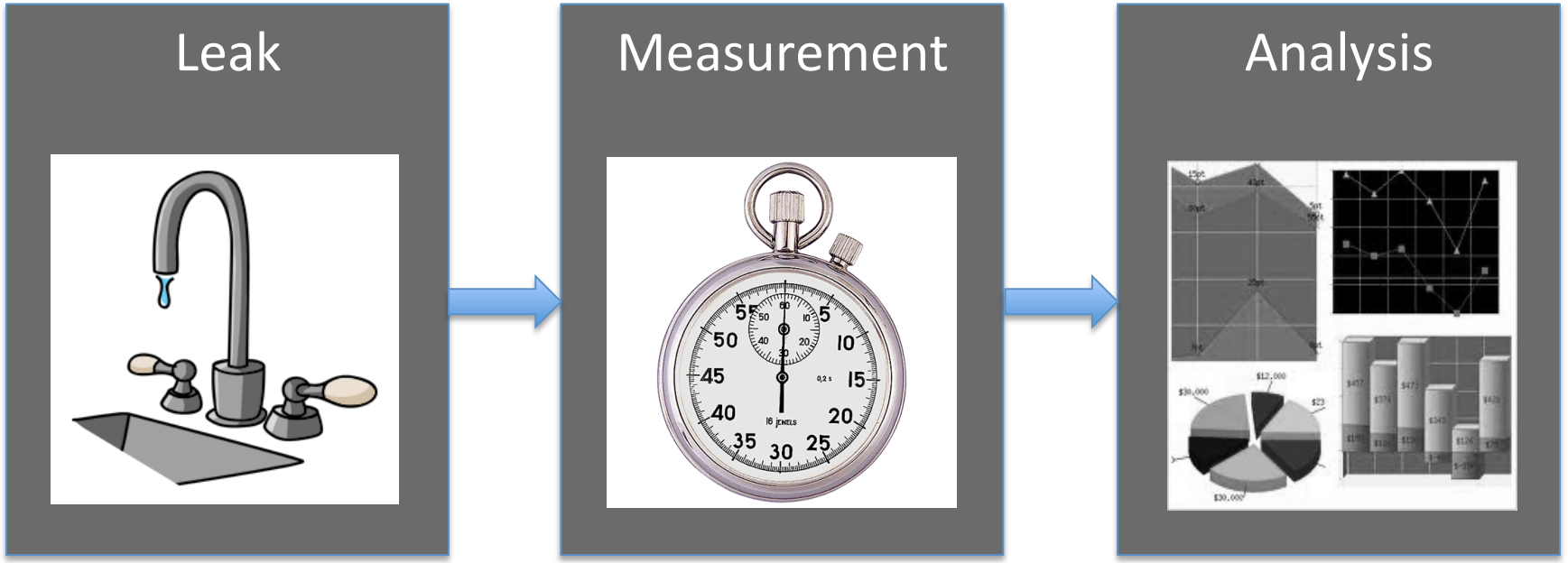
Prior Solutions

- Reactive
- Ad-hoc
- Fix individual leaks

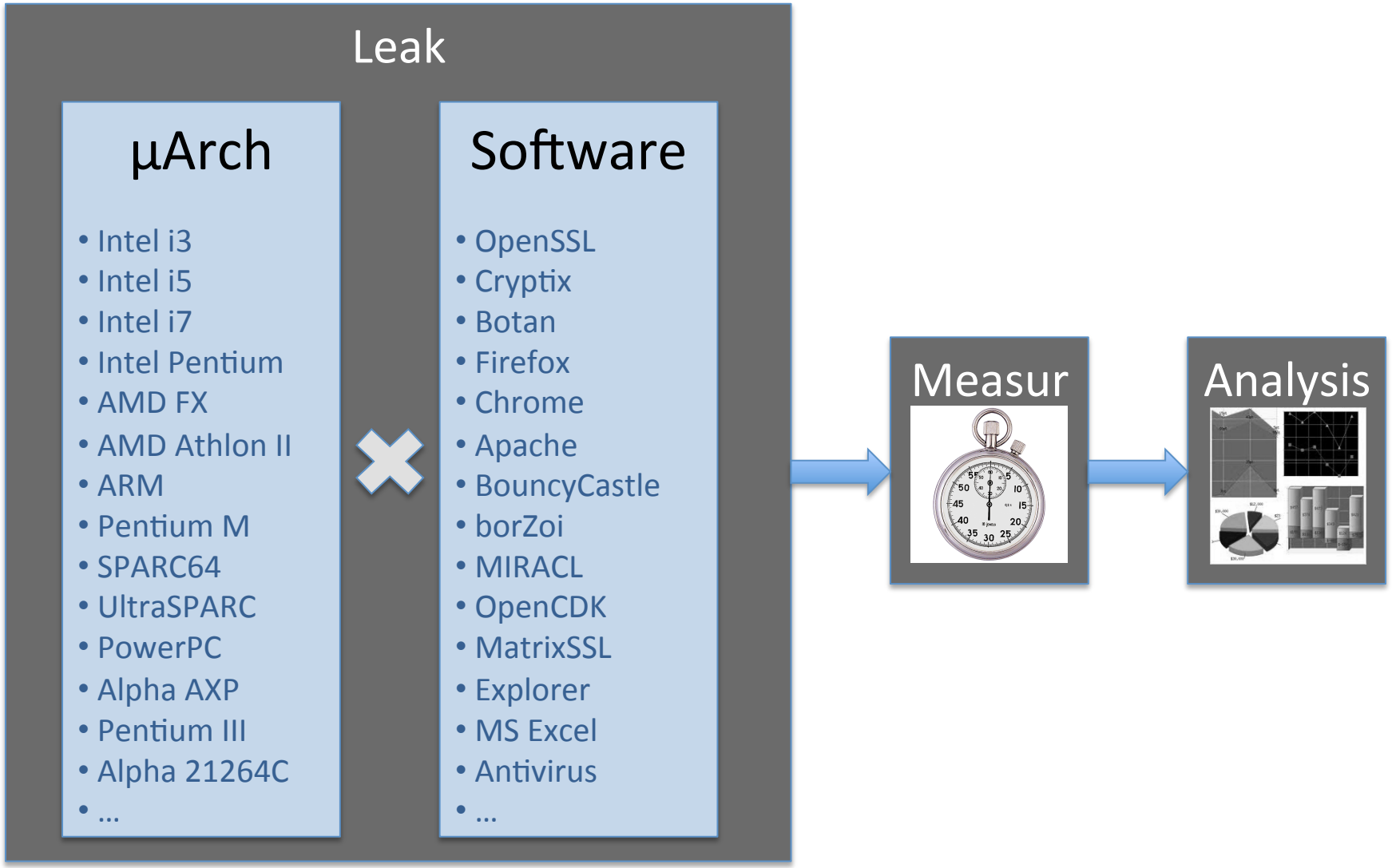
TimeWarp's New Approach

Instead of stopping leaks, stop measurements.

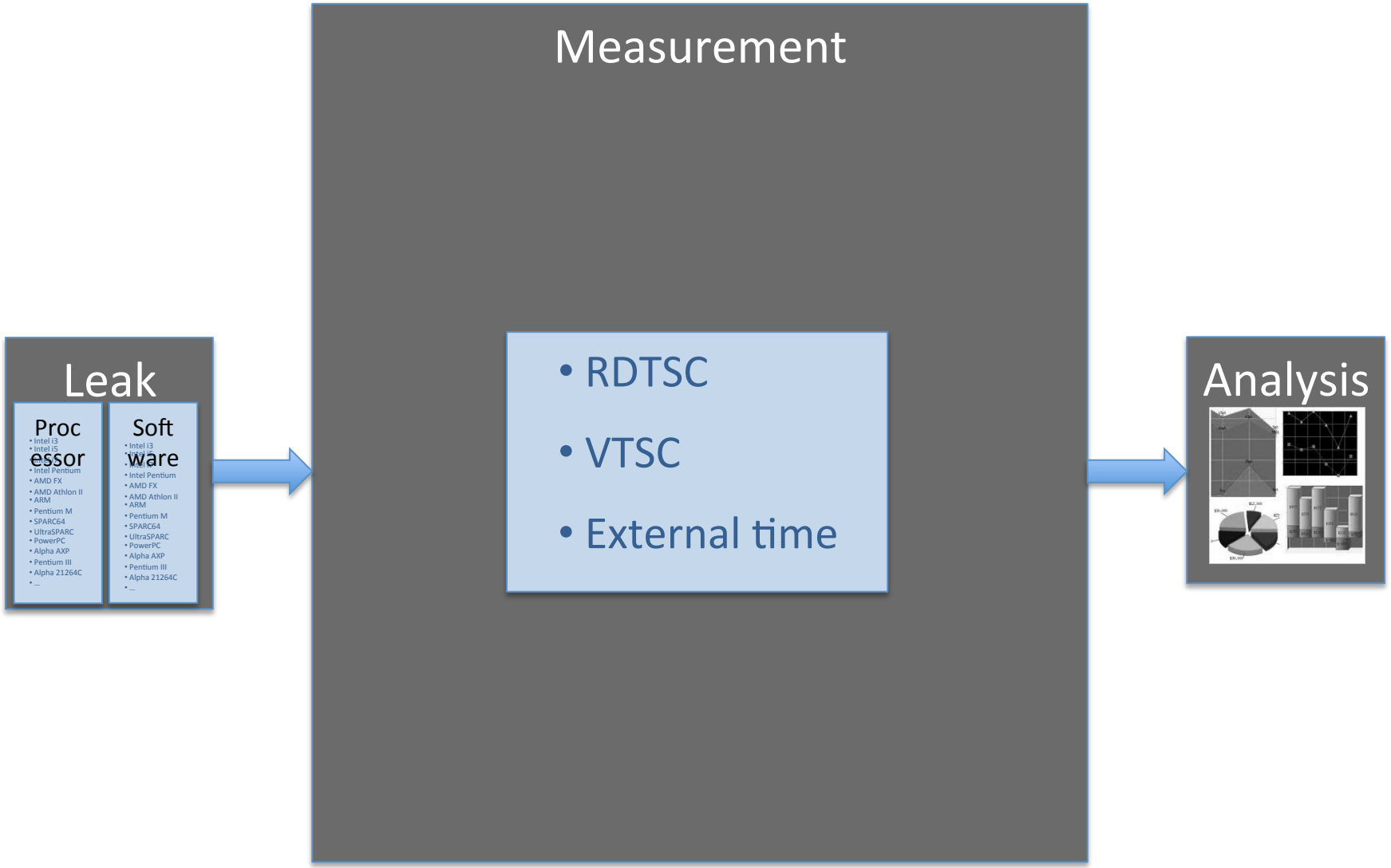
Anatomy of a Microarchitectural Attack



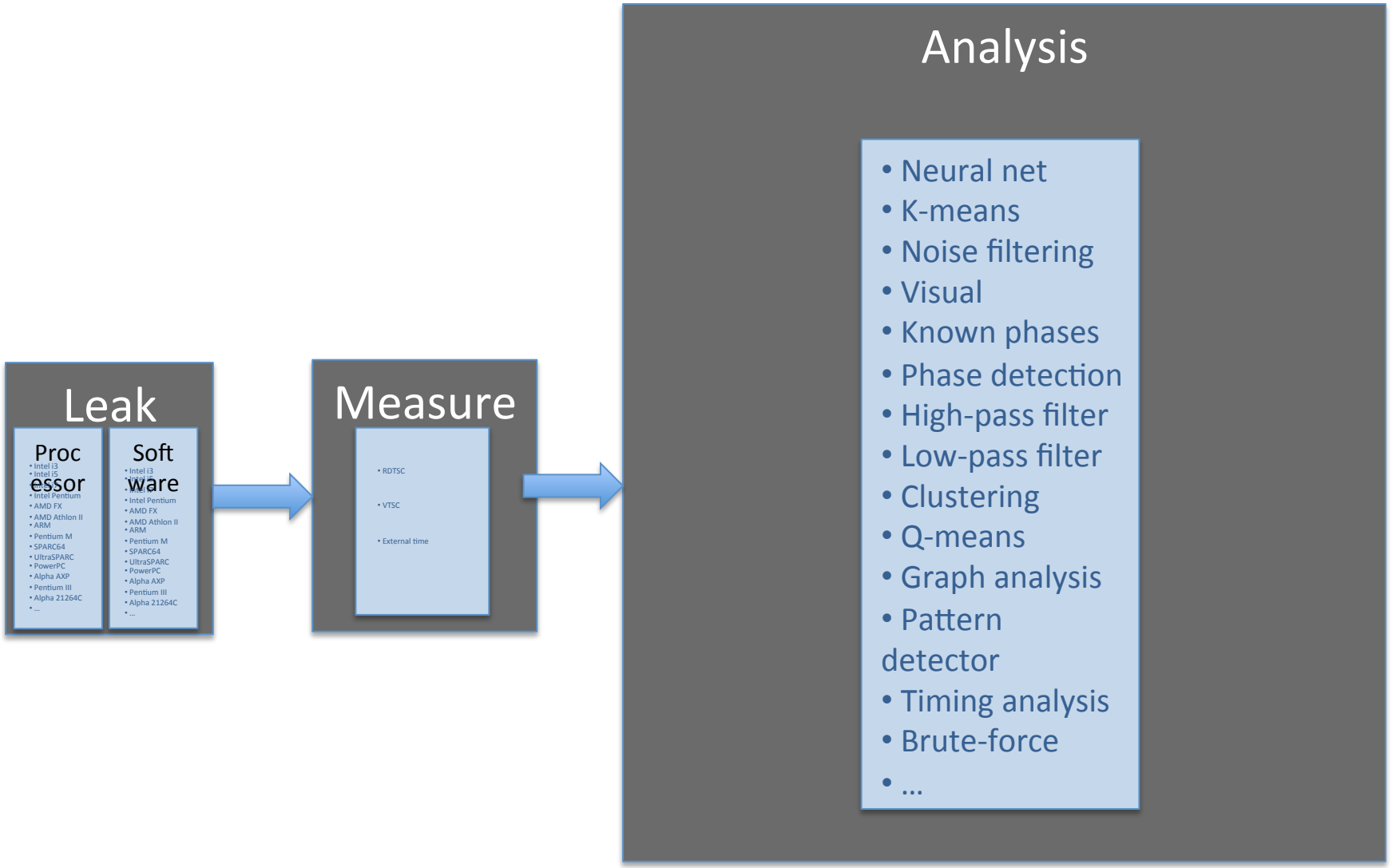
Anatomy of a Microarchitectural Attack



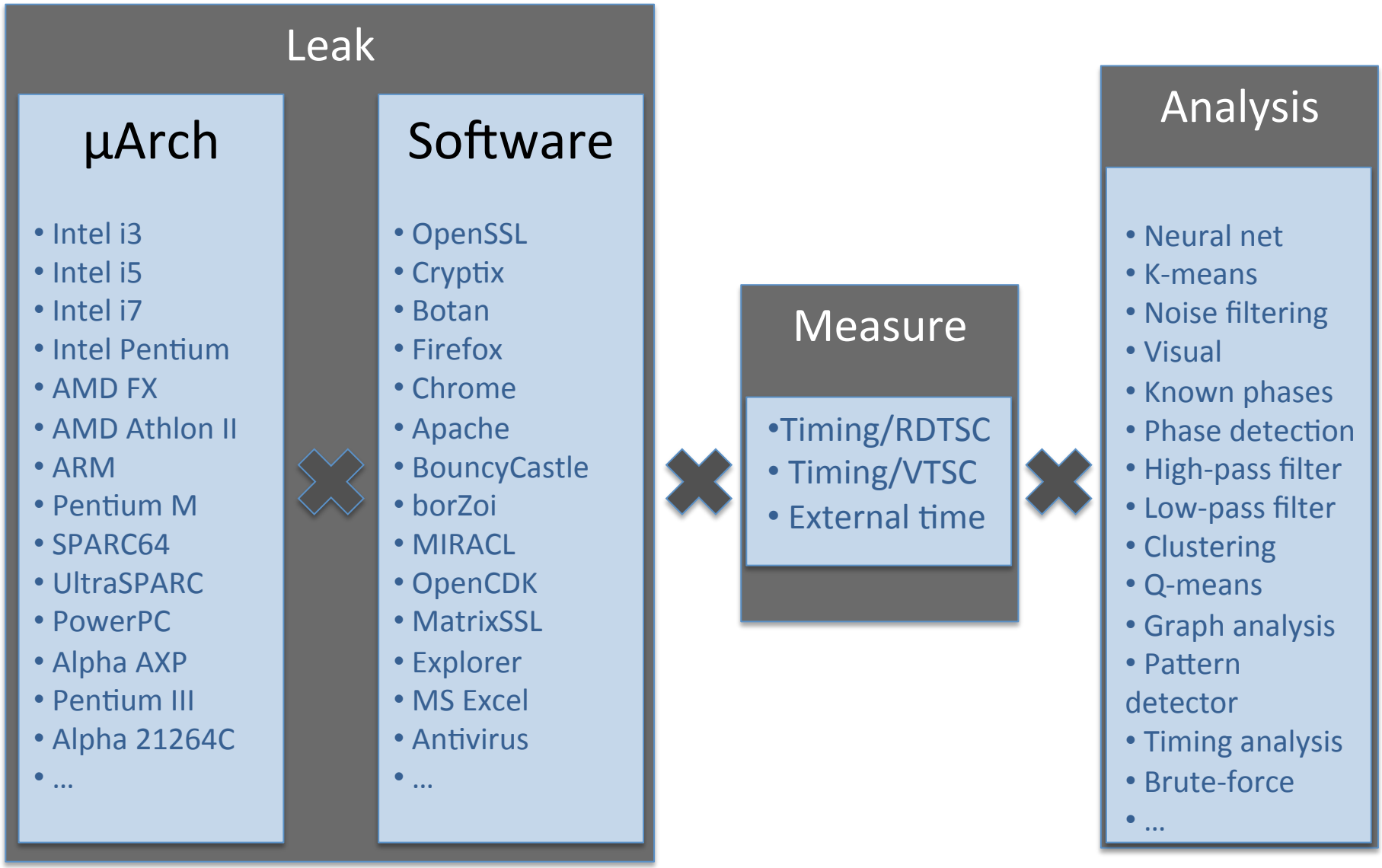
Anatomy of a Microarchitectural Attack



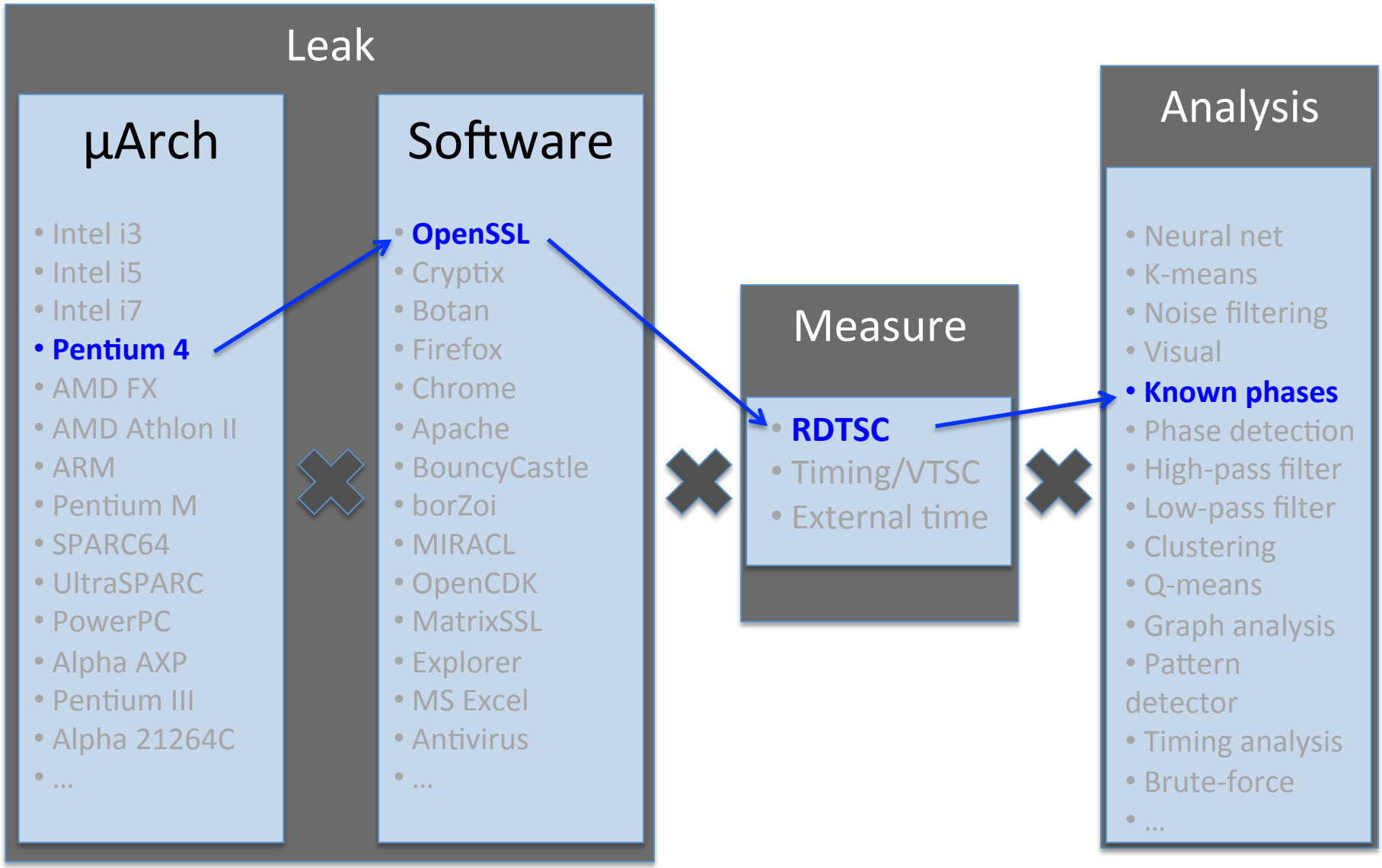
Anatomy of a Microarchitectural Attack



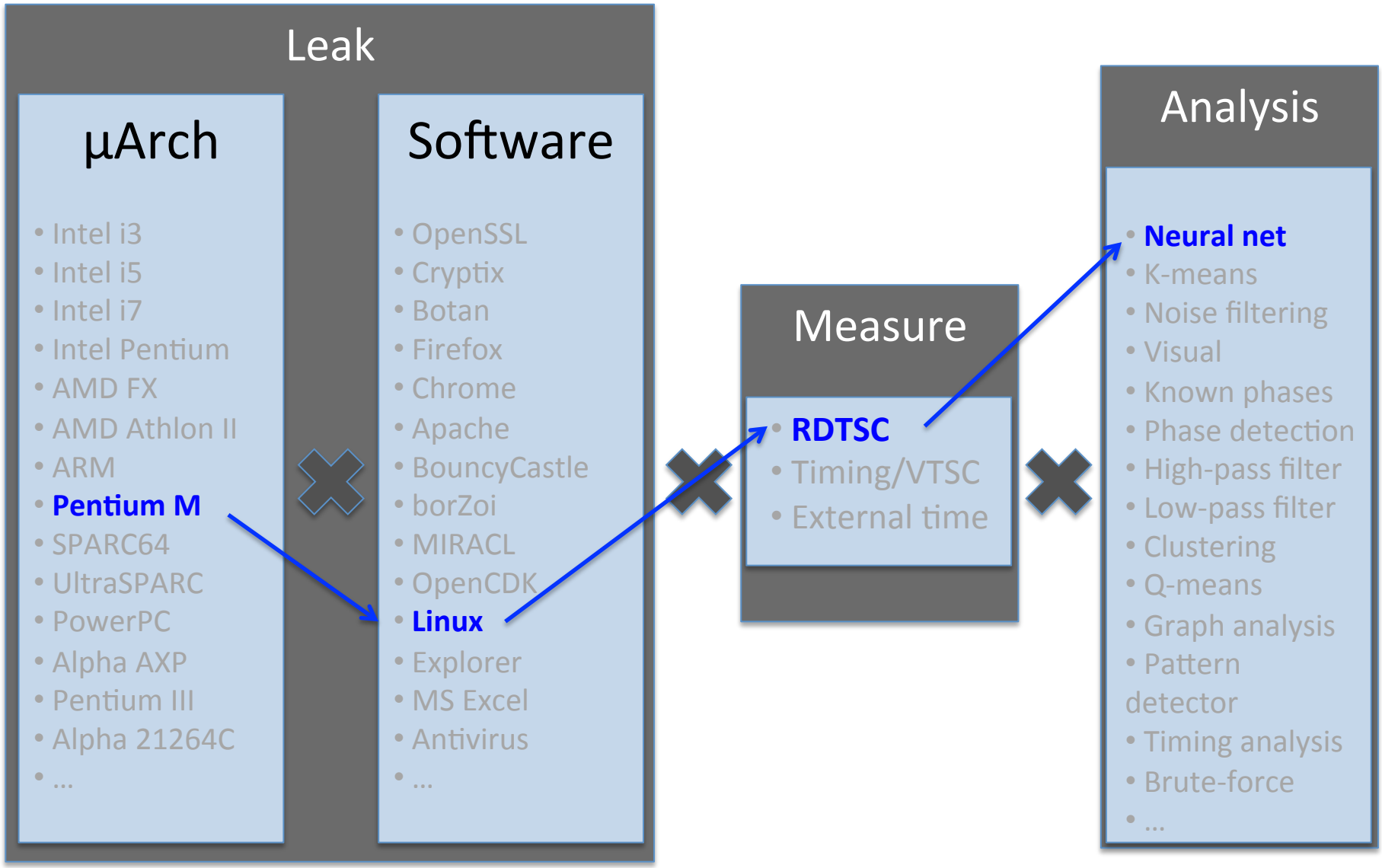
Attack Surface for MA Attacks



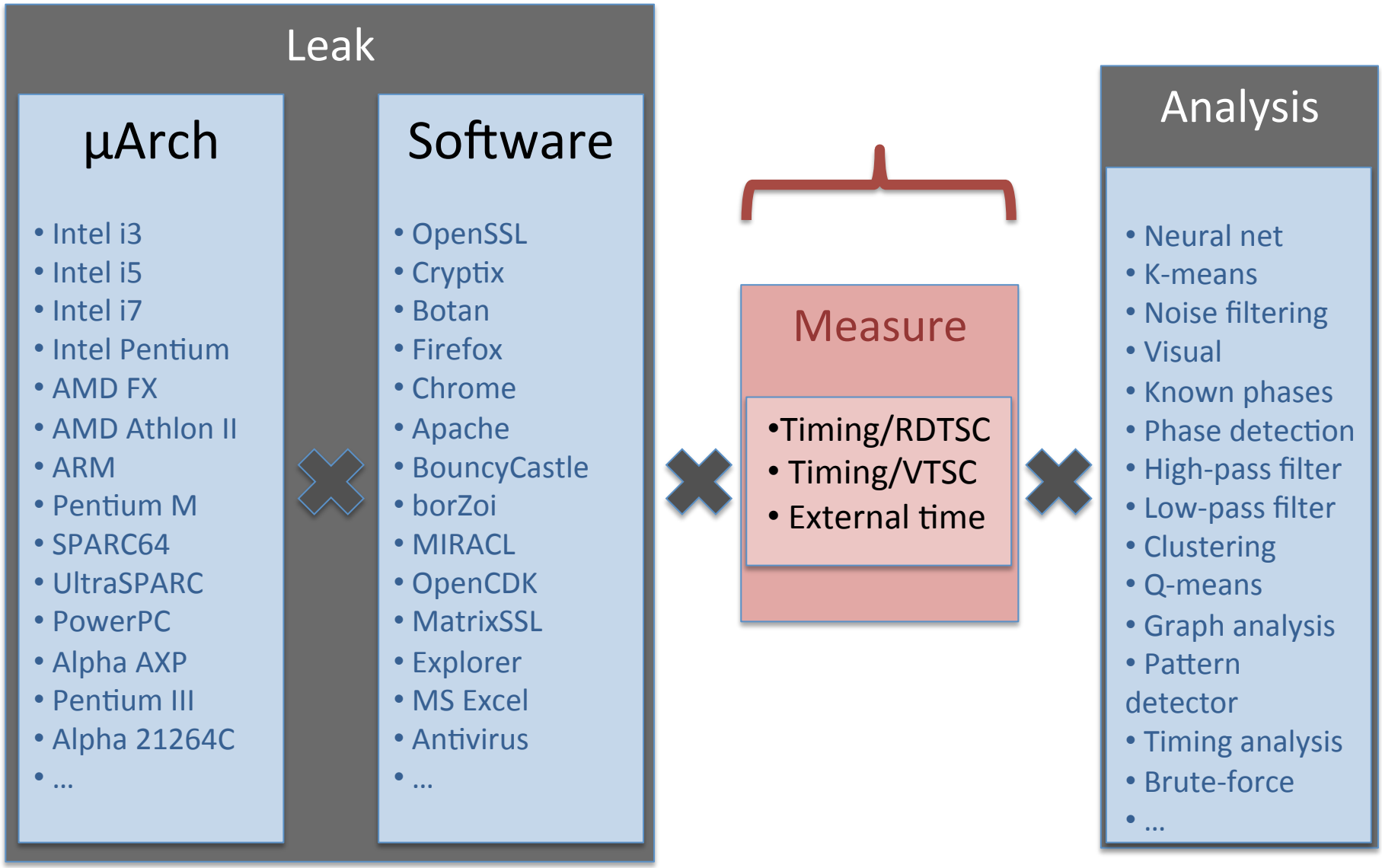
Percival's Attack (2005)



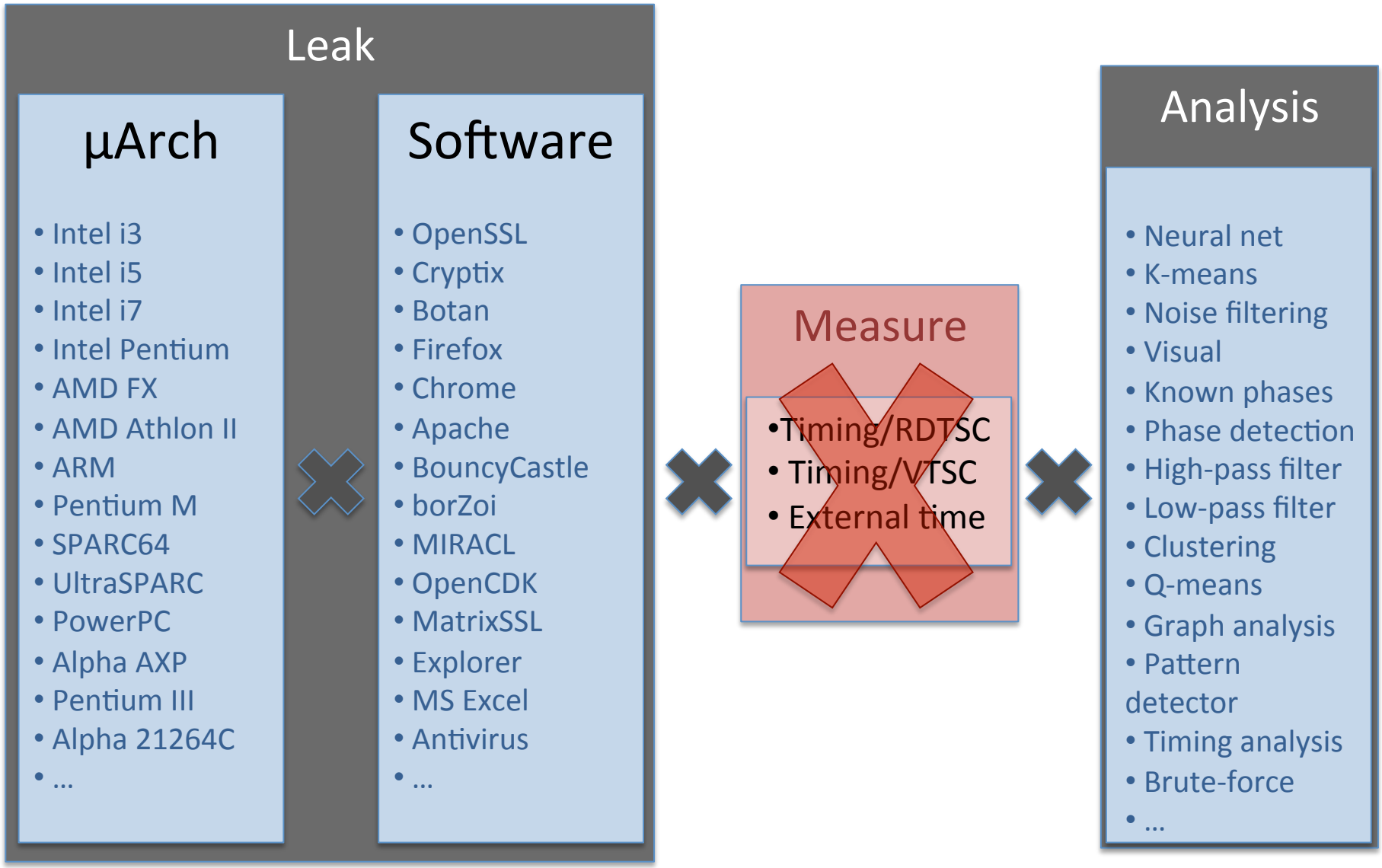
Cache Games (2011)



Attack Surface for MA Attacks



Attack Surface for MA Attacks



Measurement Instruments in MA Attacks

Measurement

1. Timing Instructions (RDTSC)
2. Inter-thread communication (VTSC)
3. External Timing Information

Measurement Instruments in MA Attacks

1. **Timing Instructions (RDTSC)**
2. Inter-thread communication (VTSC)
3. External Timing Information

```
A = RDTSC()
LOAD addr 0xABCD
B = RDTSC()
```

Measurement Instruments in MA Attacks

Previously Published Attacks

Year	Target	Authors	Hardware	Software	Measure method
2005	d-cache	Percival	Pentium 4	OpenSSL RSA	**RDTSC**
2005	d-cache	Bernstein	Pentium III	OpenSSL AES	**RDTSC**
2006	BPU	Aciicmez…	Pentium 4	OpenSSL RSA	**RDTSC**
2007	i-cache	Aciicmez…		OpenSSL RSA	**RDTSC**
2010	d-cache	Jayasinghe…		OpenSSL AES	**RDTSC**
2011	d-cache	Bangerter…	Pentium 4	OpenSSL AES	**RDTSC**

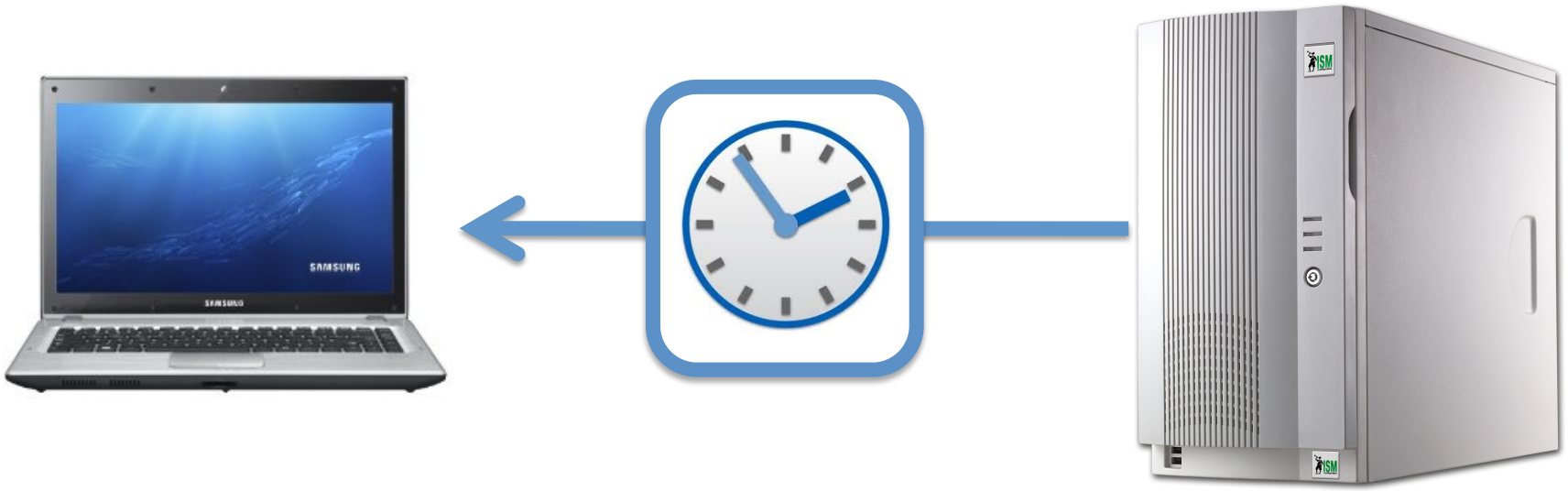
Measurement Instruments in MA Attacks

1. Timing Instructions (RDTSC)
2. **Inter-thread communication (VTSC)**
3. External Timing Information

```
A = addr 0x1000          loop:
LOAD addr 0xABCD         INC R1
B = addr 0x1000          STORE R1 → 0x1000
                         JUMP loop
```

Measurement Instruments in MA Attacks

1. Timing Instructions (RDTSC)
2. Inter-thread communication (VTSC)
3. **External Timing Information**



Measurement Instruments in MA Attacks

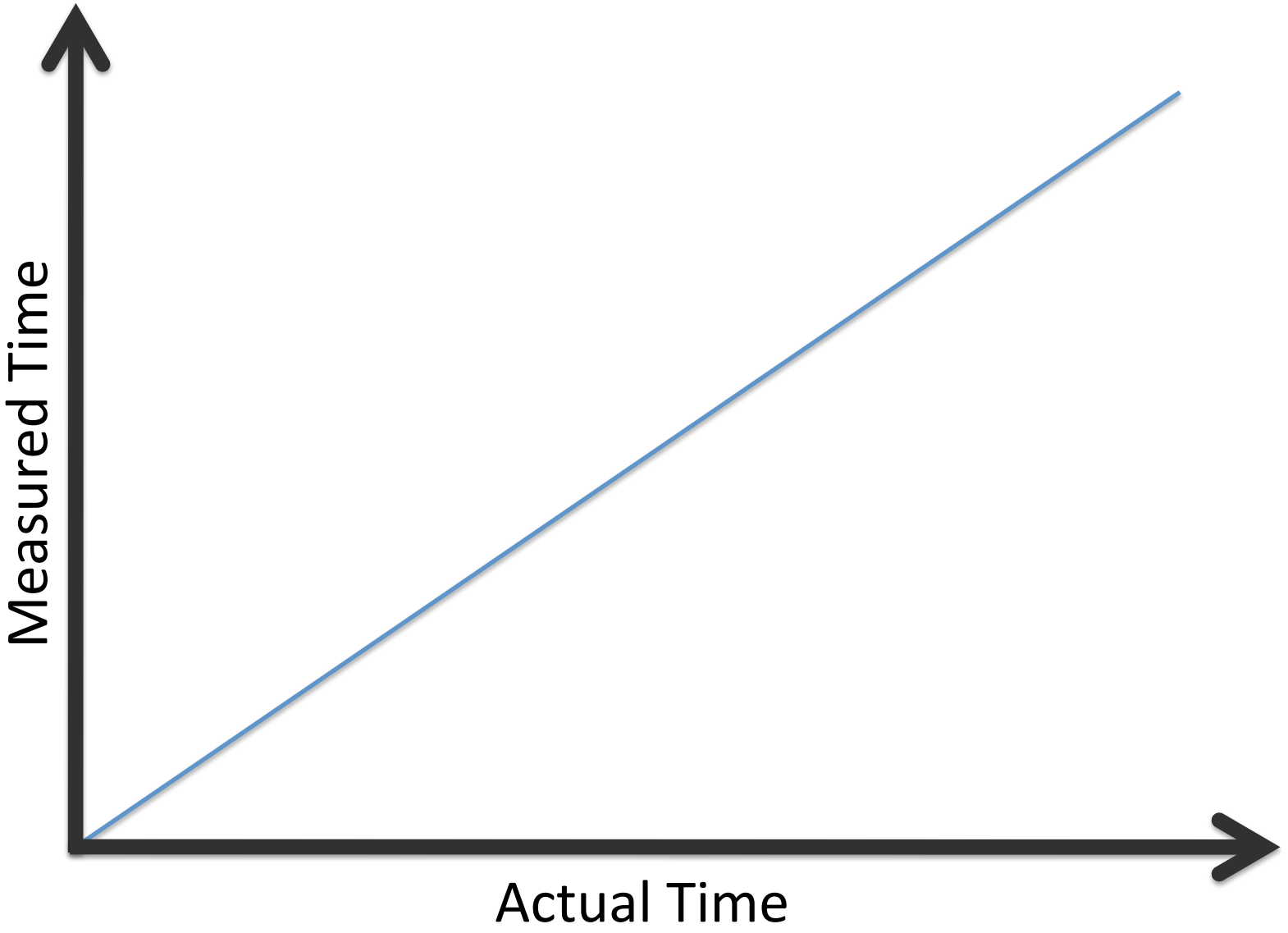
Measurement

1. Timing Instructions (RDTSC)
2. Inter-thread communication (VTSC)
3. External Timing Information

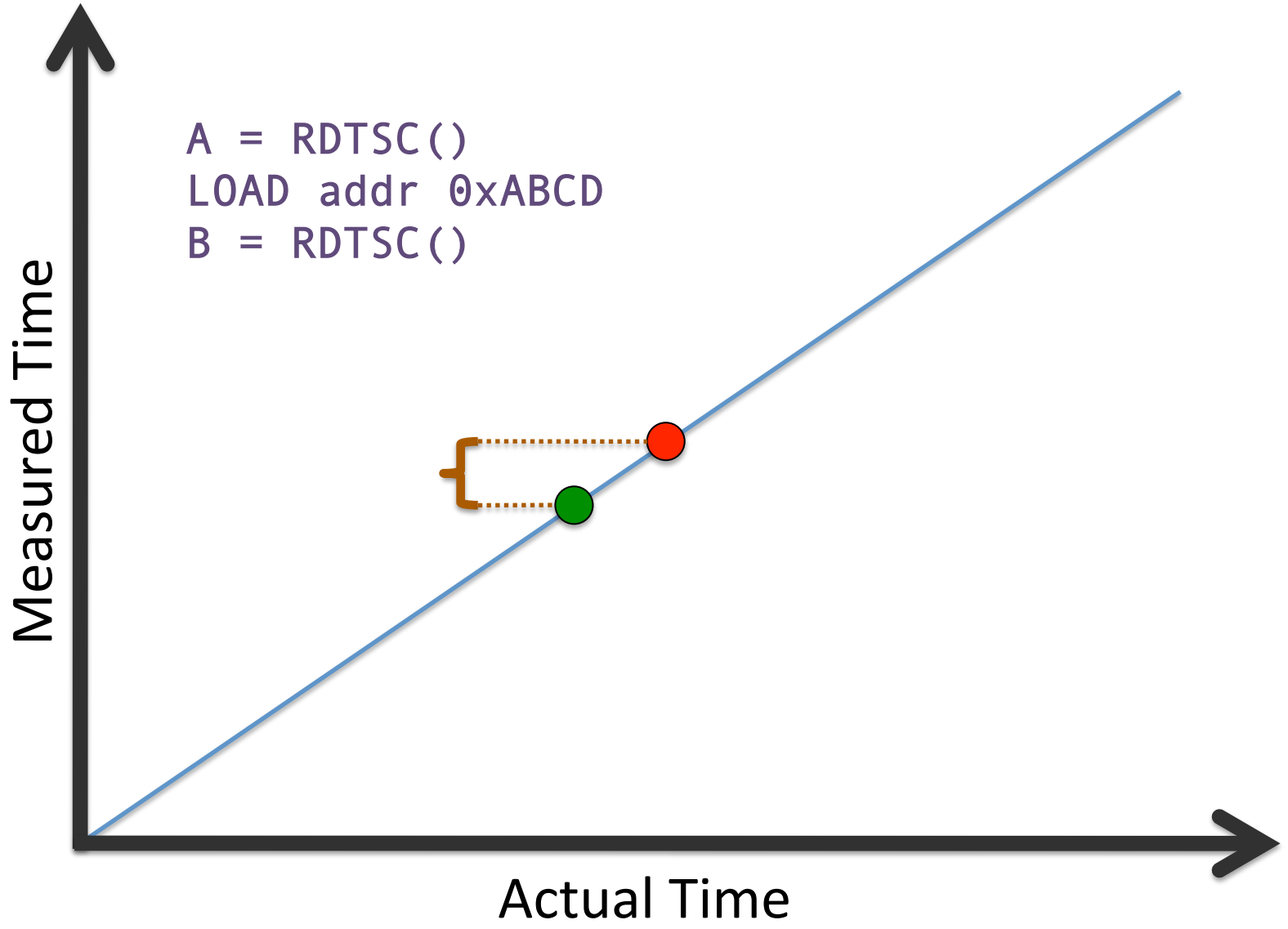
Countermeasure 1

RDTSC Fuzzing

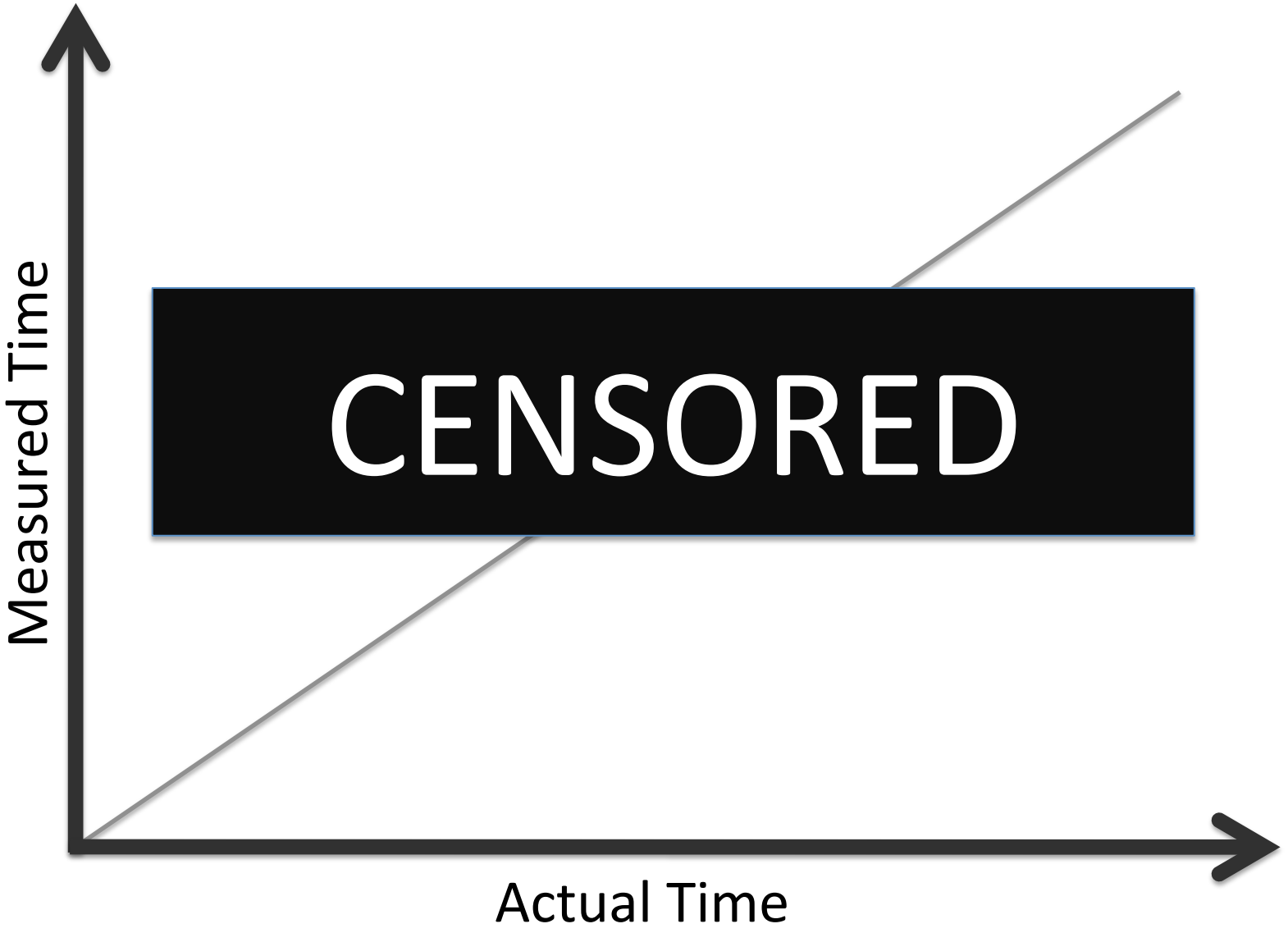
Countermeasure 1: RDTSC Fuzzing



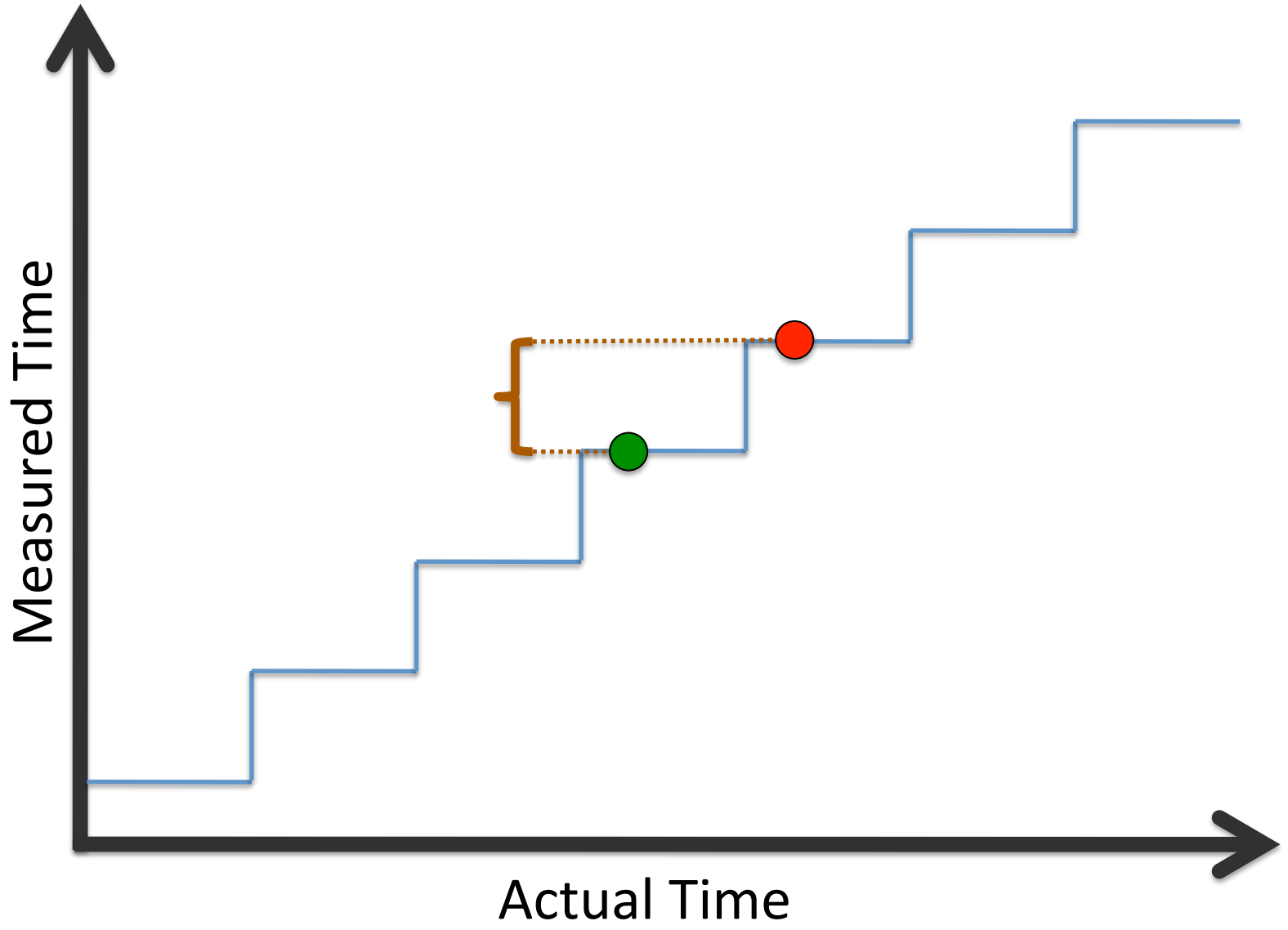
Countermeasure 1: RDTSC Fuzzing



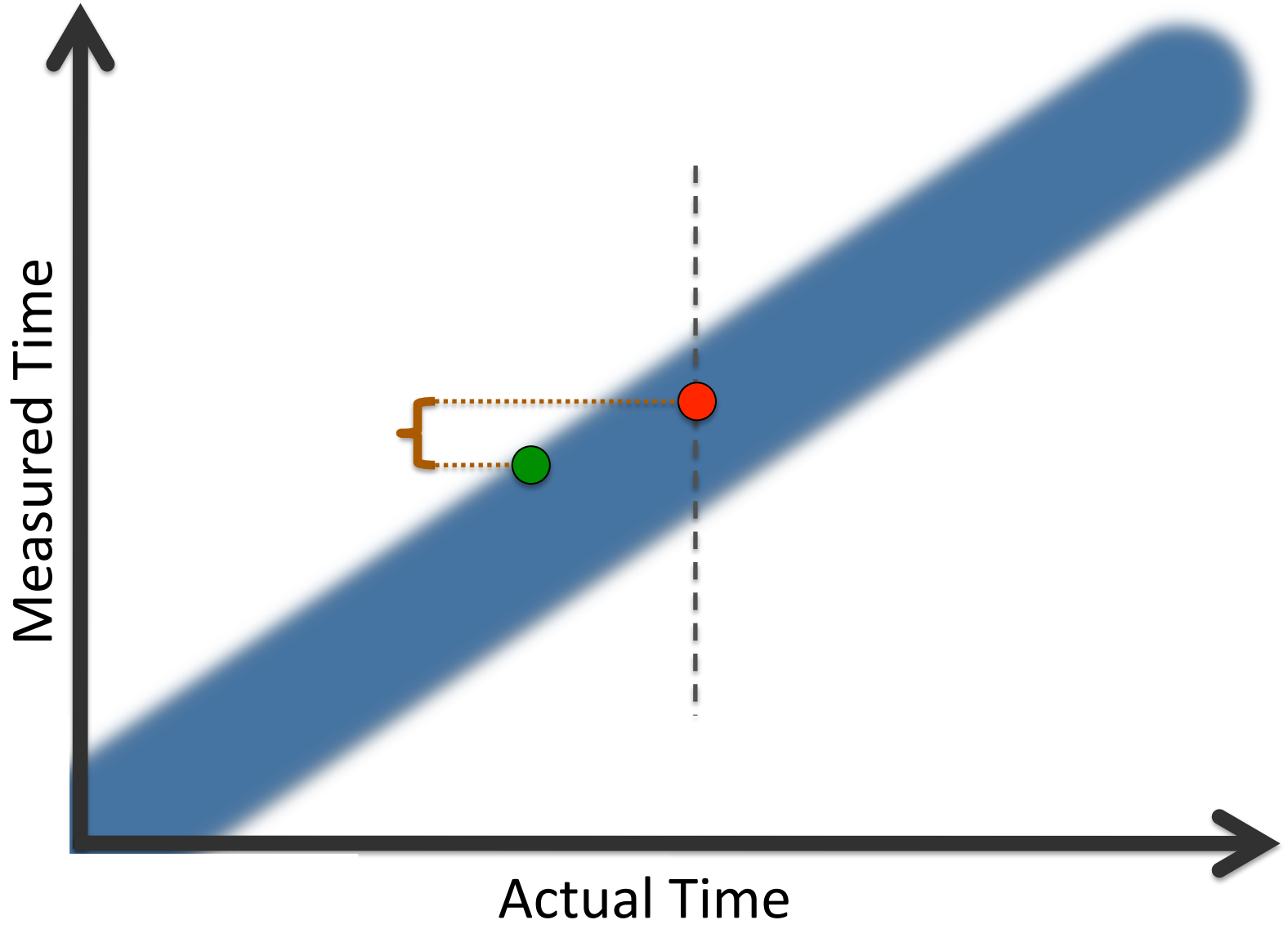
Disable RDTSC



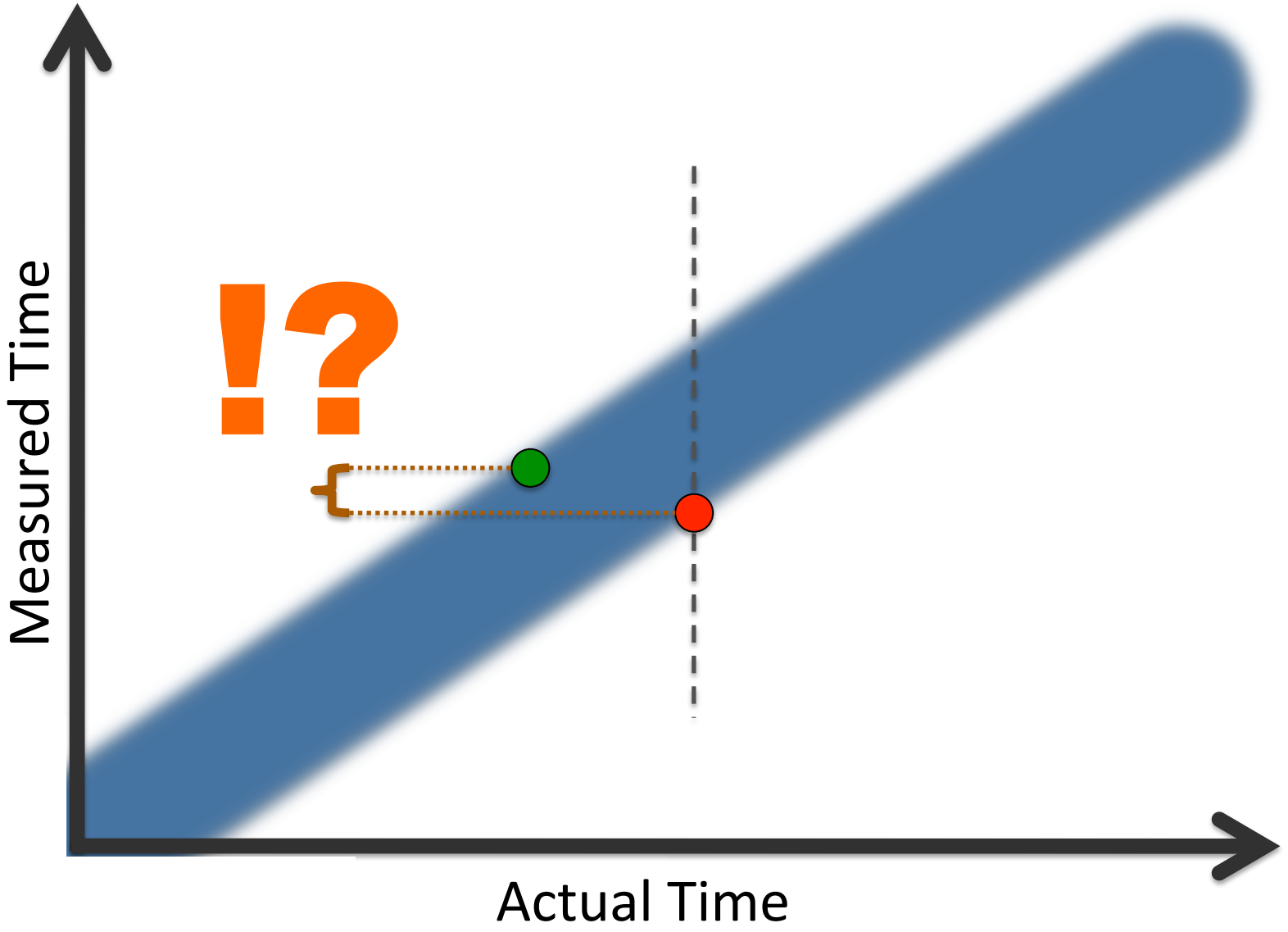
RDTSC Step Function



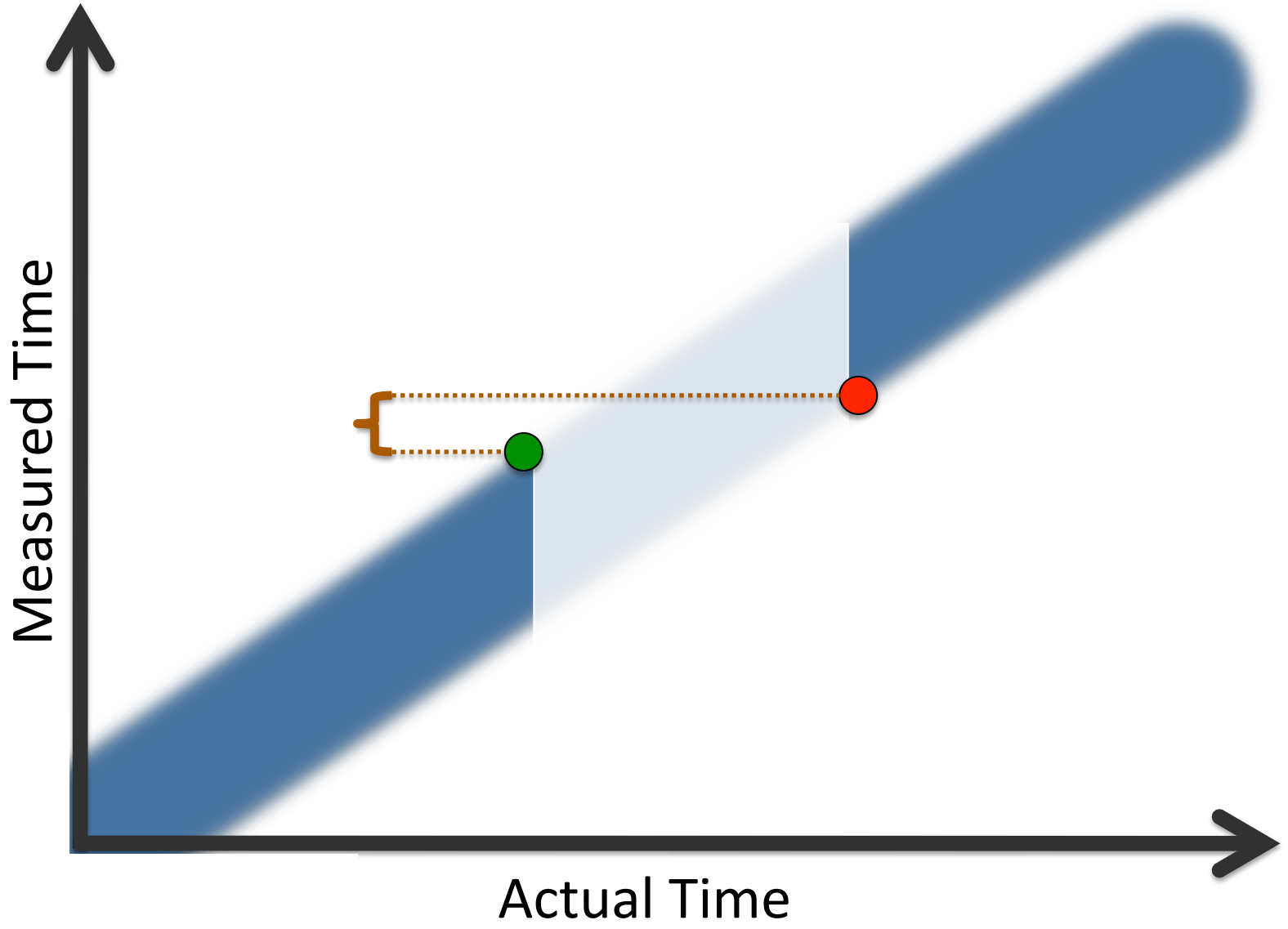
Random Offset to RDTSC



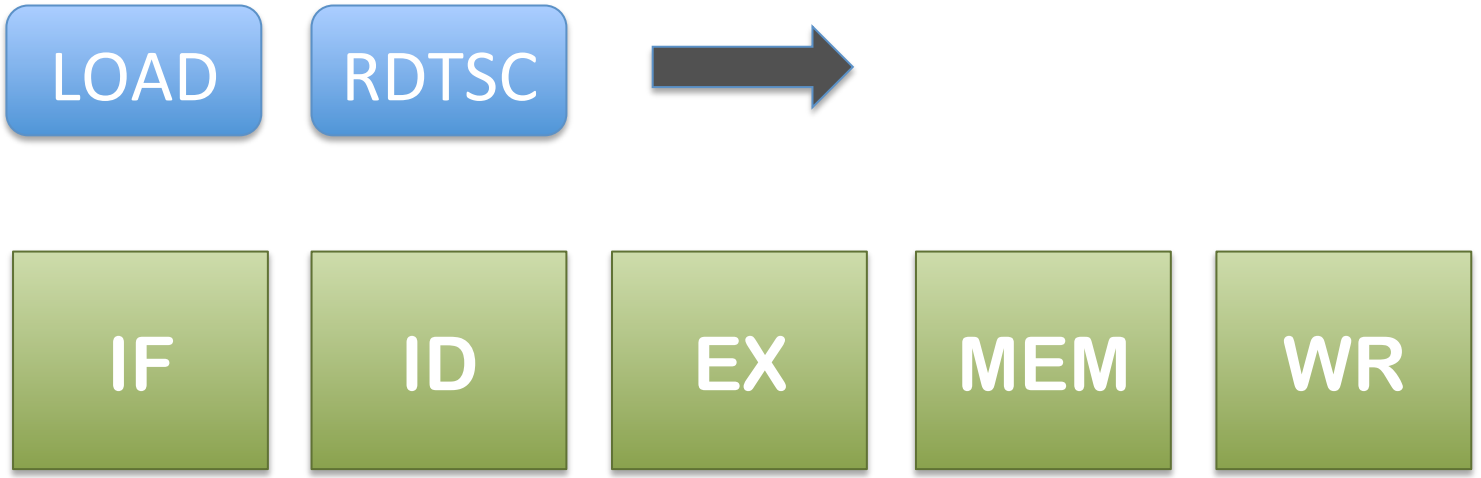
Random Offset to RDTSC



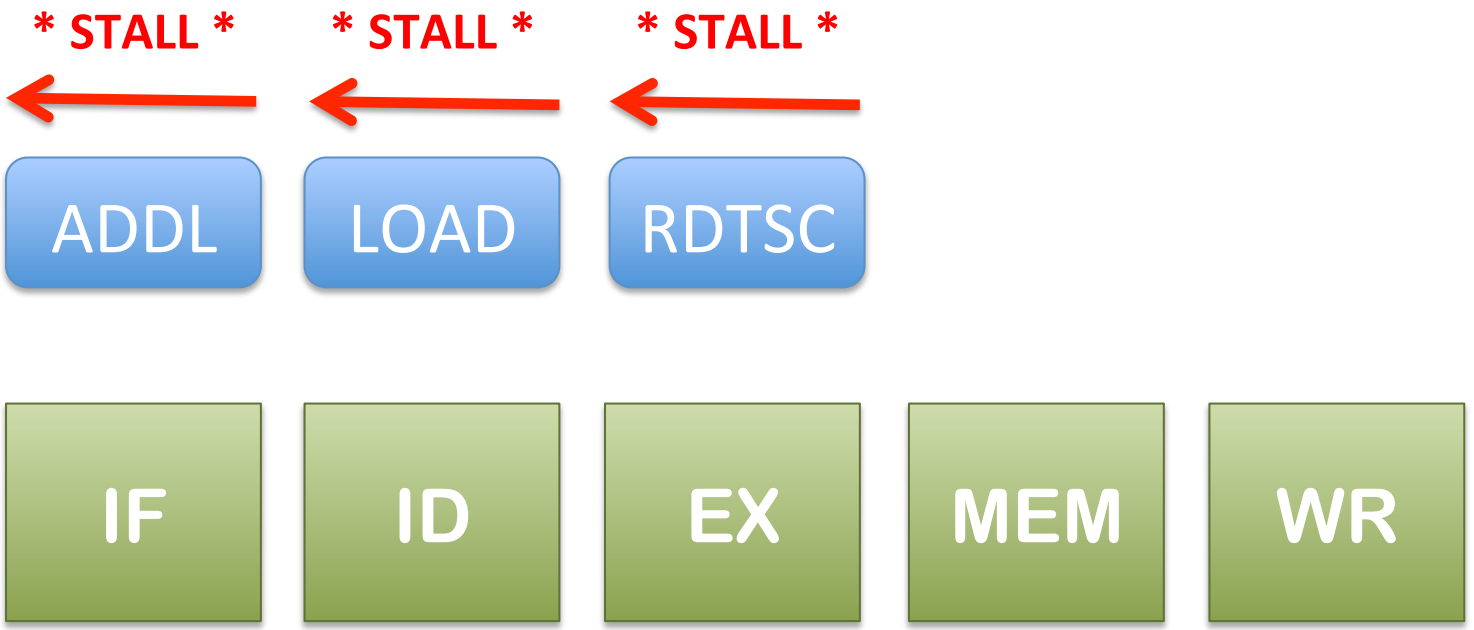
TimeWarp Fuzzing Scheme



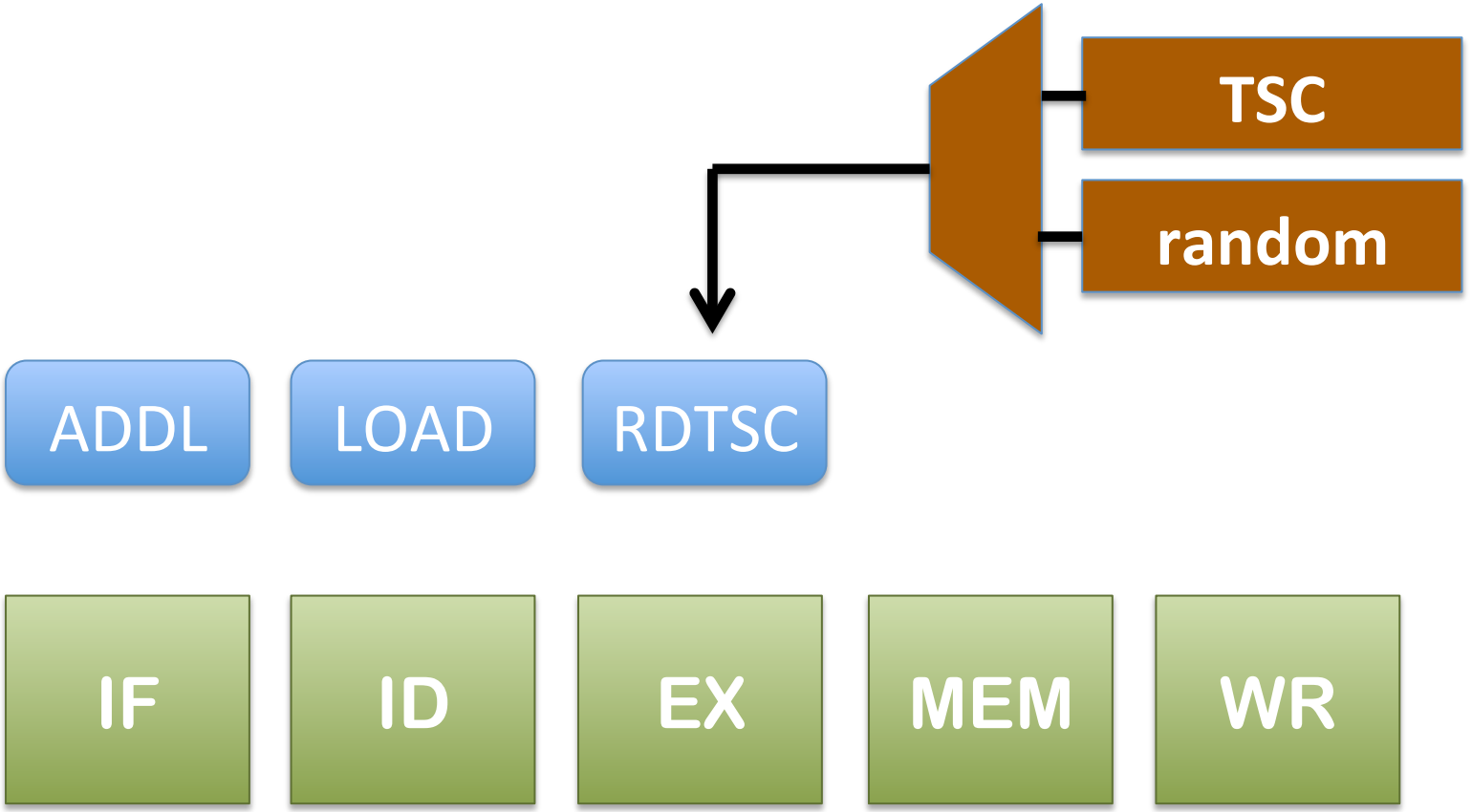
Implementation Details



Implementation Details



Implementation Details



Countermeasure 1: RDTSC Fuzzing

- Simple implementation

- Fuzzing is configurable

- Doesn't affect OS

Measurement Instruments in MA Attacks

Measurement

1. ~~Timing Instructions (RDTSC)~~
2. **Inter-thread communication (VTSC)**
3. **External Timing Information**

Countermeasure 2

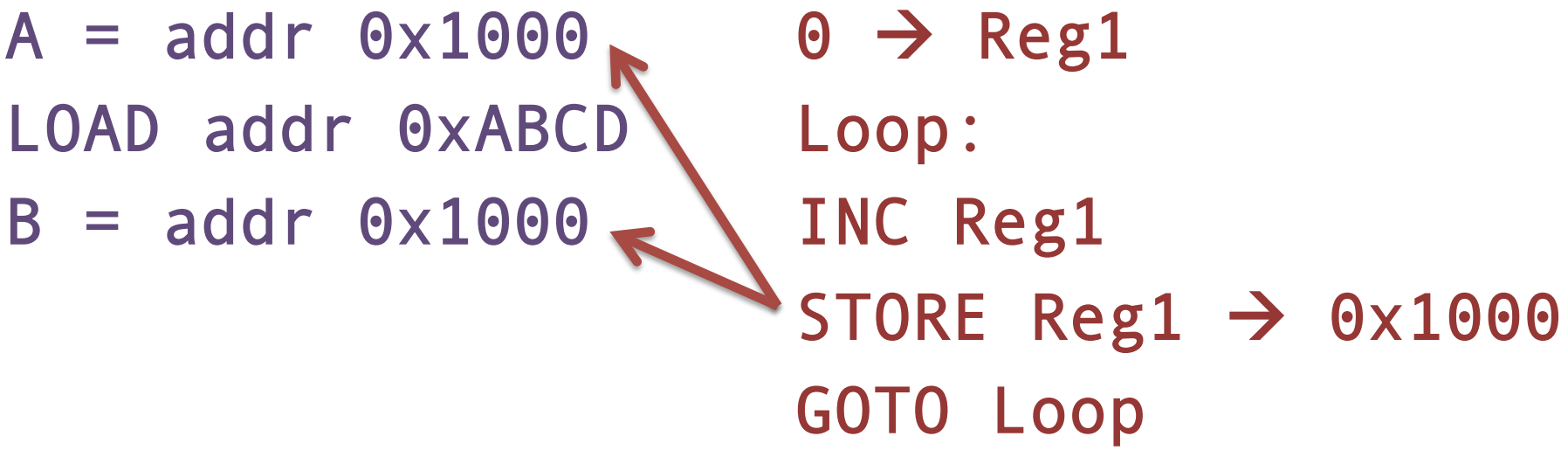
VTSC Fuzzing

Countermeasure 2: VTSC Fuzzing

- **Key insight: VTSC requires high-speed coherence traffic.**

Countermeasure 2: VTSC Fuzzing

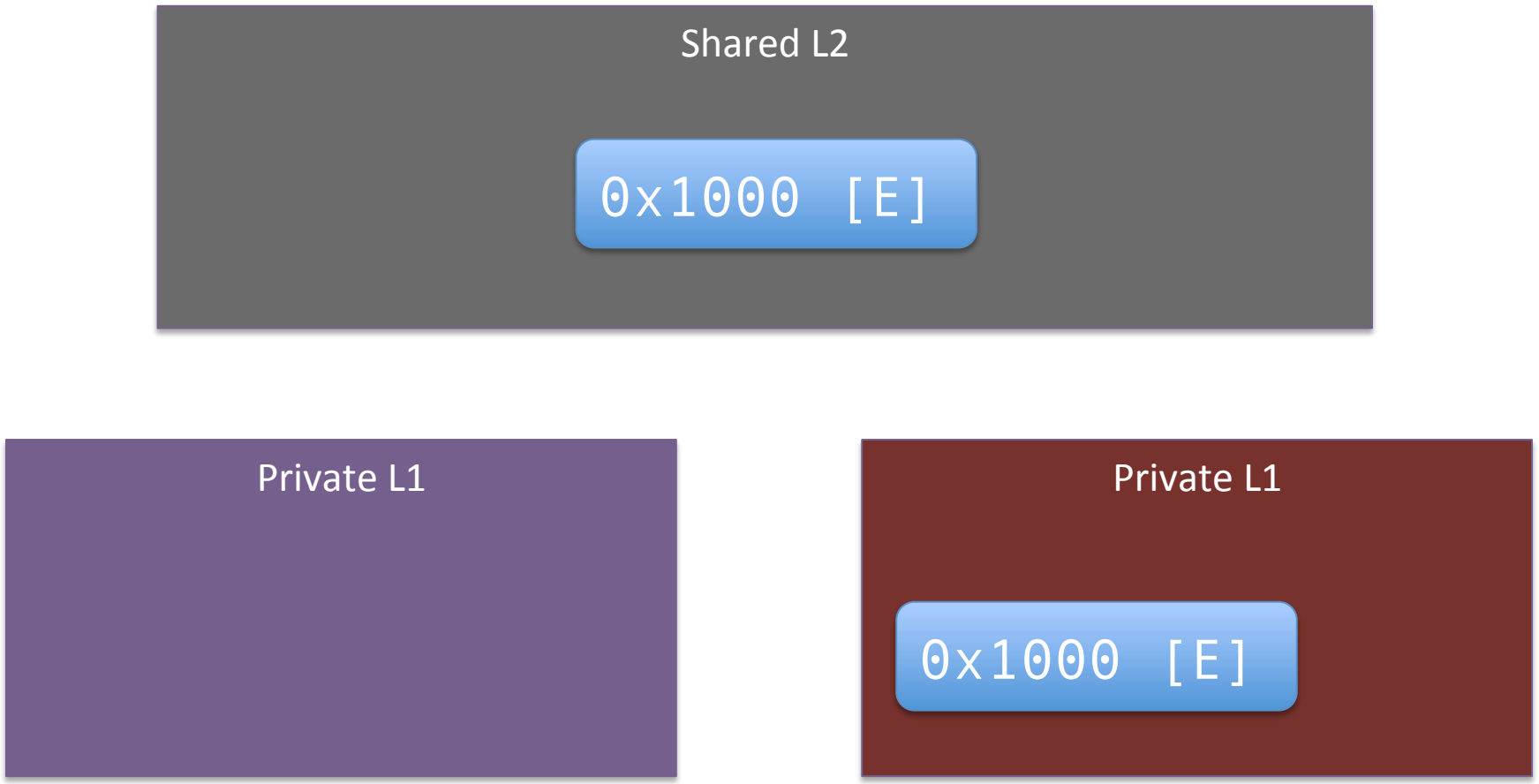
- **Key insight: VTSC requires high-speed coherence traffic.**



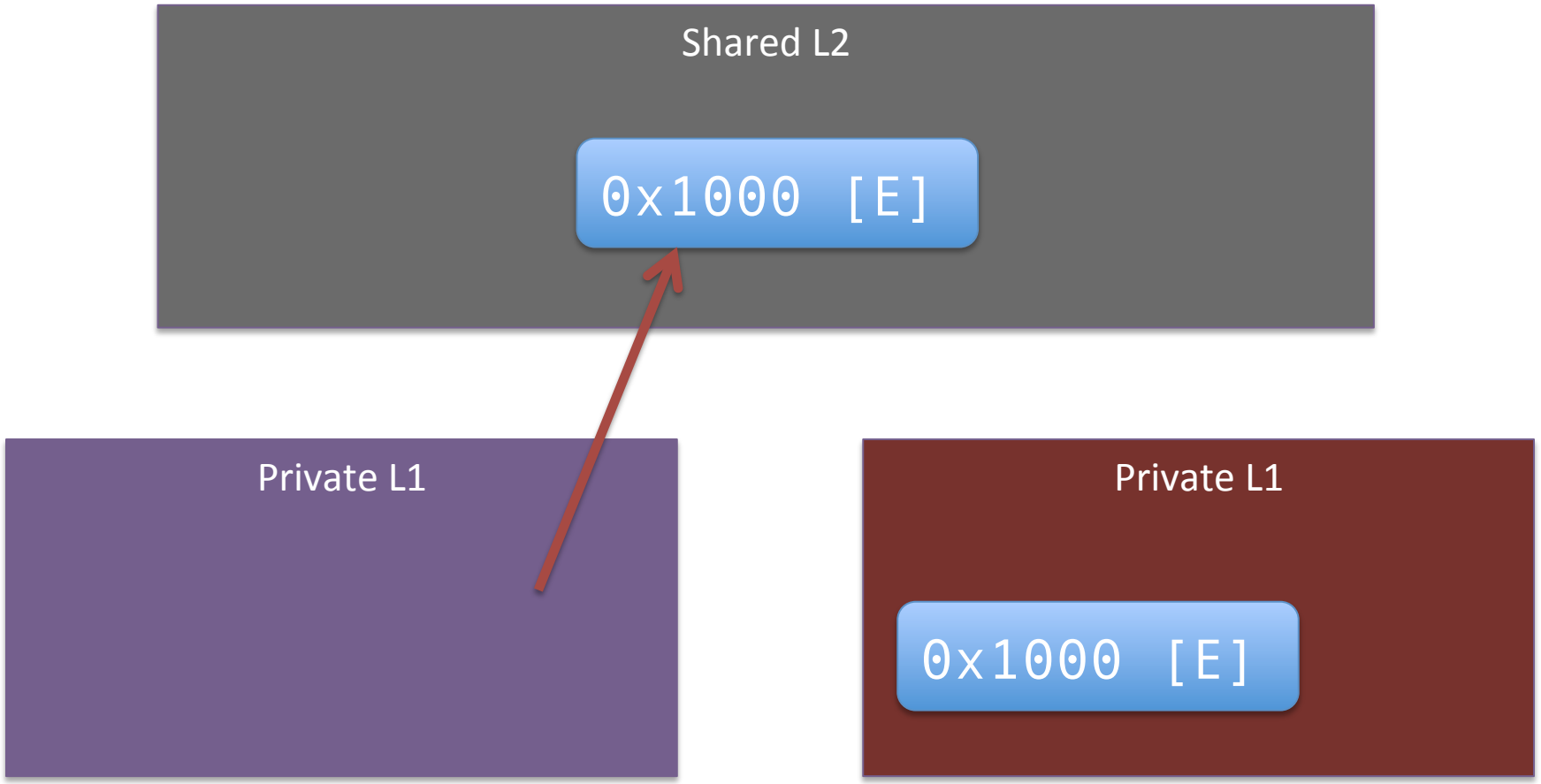
Countermeasure 2: VTSC Fuzzing

- **Key insight: VTSC requires high-speed coherence traffic.**

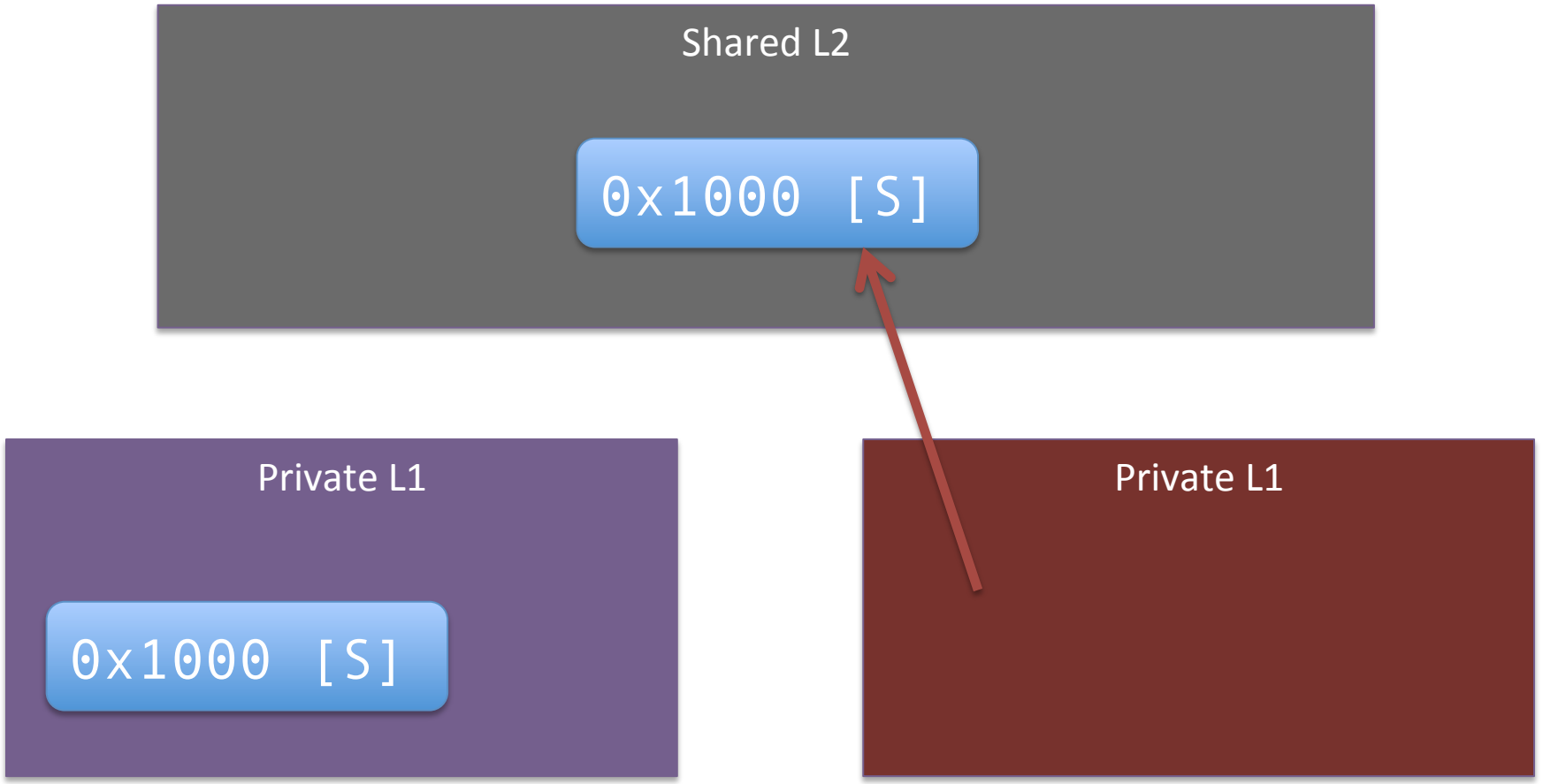
Countermeasure 2: VTSC Fuzzing



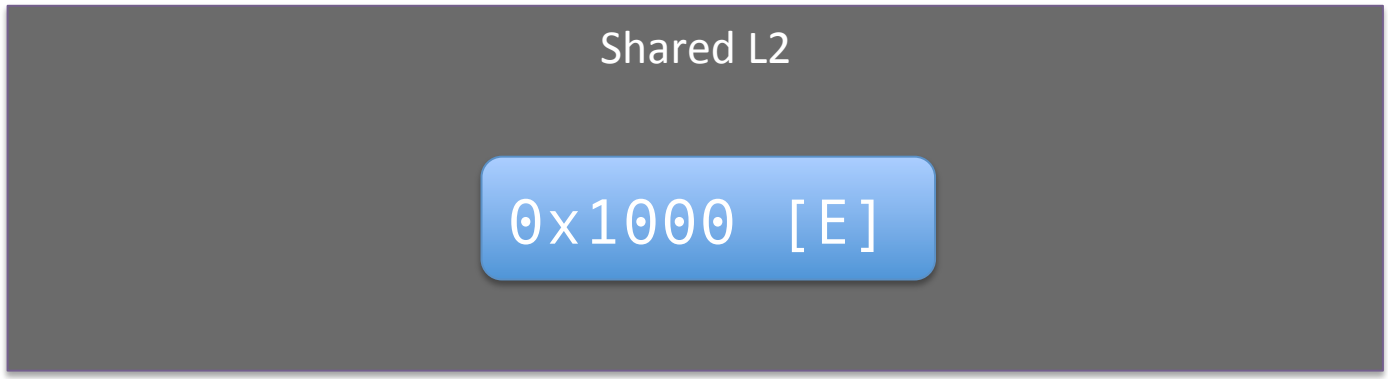
Countermeasure 2: VTSC Fuzzing



Countermeasure 2: VTSC Fuzzing



Countermeasure 2: VTSC Fuzzing

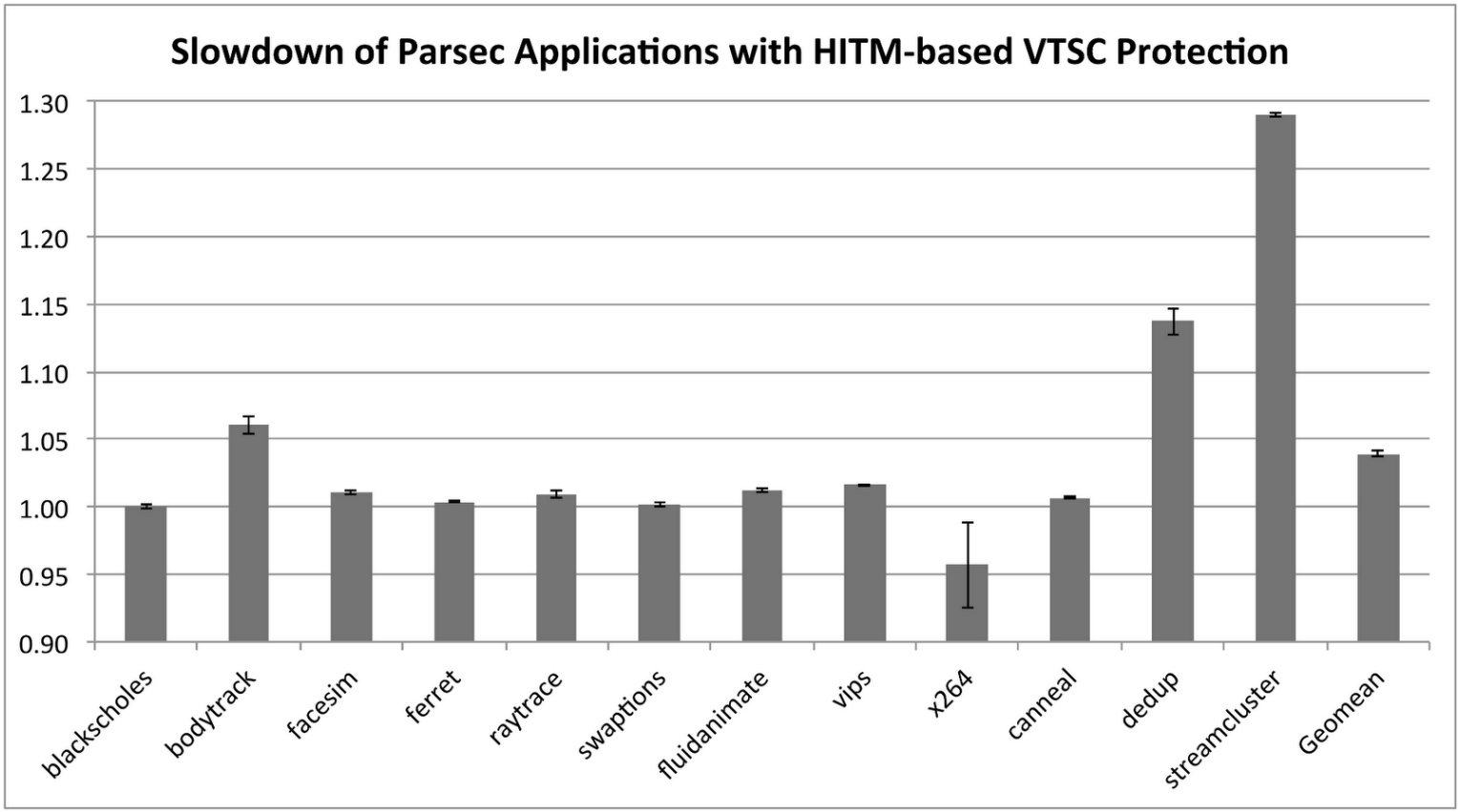



```
A = addr 0x1000
```

```
INC Reg1
STORE Reg1 → 0x1000
```

Countermeasure 2: VTSC Fuzzing

Disrupting these events does not significantly affect performance.



Measurement Instruments in MA Attacks

Measurement

1. ~~Timing Instructions (RDTSC)~~
2. ~~Inter-thread communication (VTSC)~~
3. **External Timing Information**

Countermeasure 3

Handling External Clocks

Countermeasure 3: External Clocks

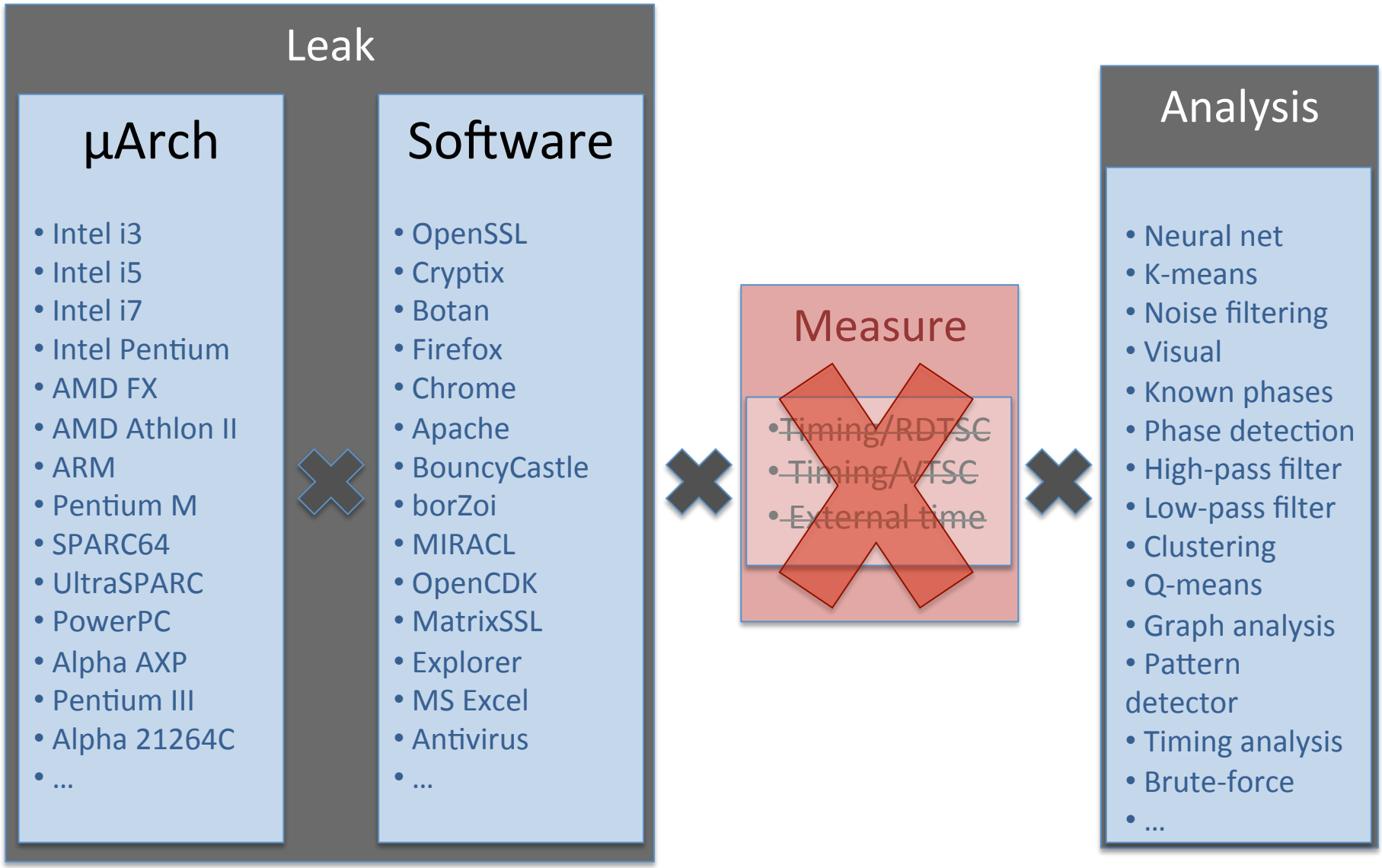
- Not a large threat at the moment.
- To be extra-safe, introduce randomness in OS

Measurement Instruments in MA Attacks

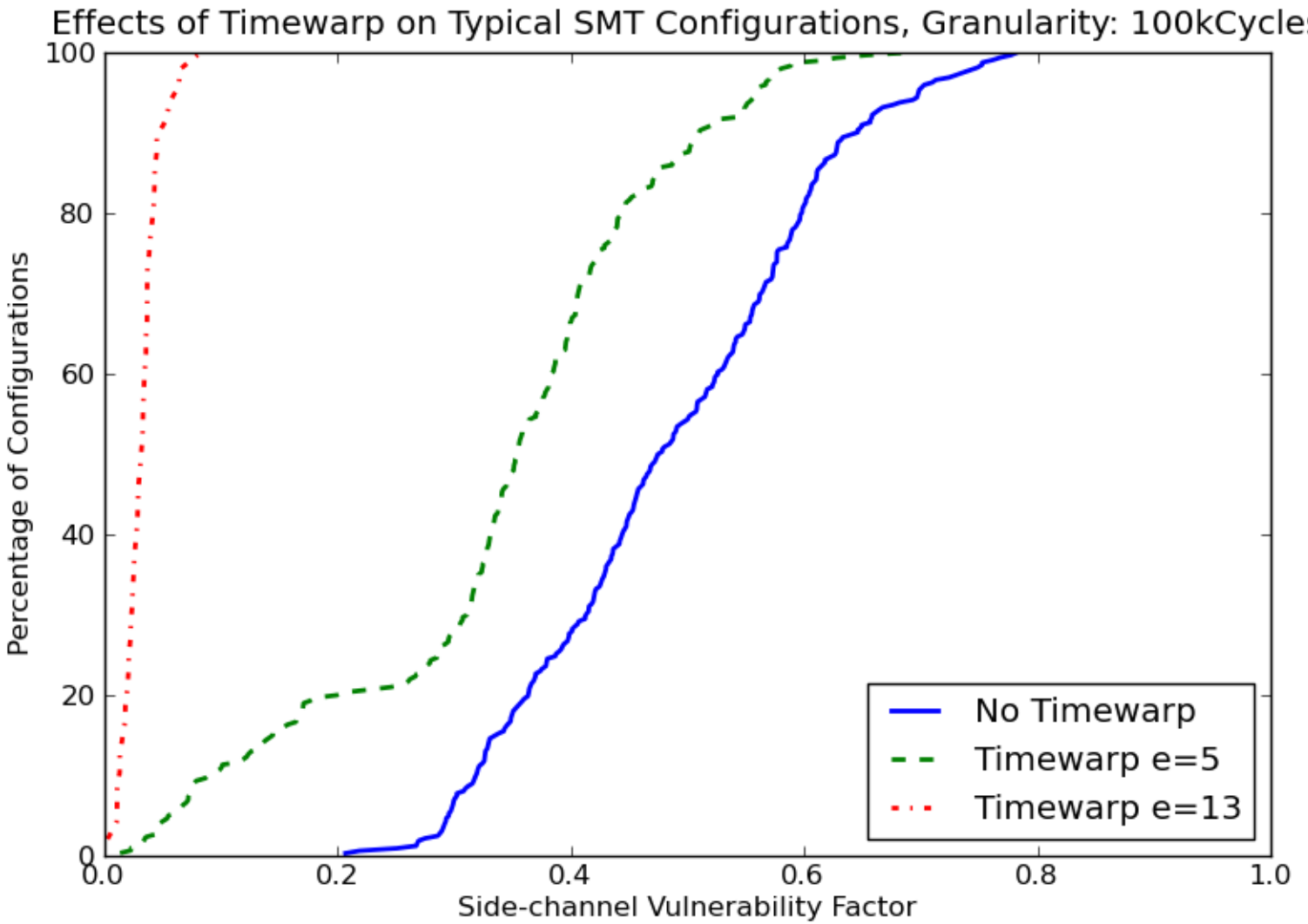
Measurement

1. ~~Timing Instructions (RDTSC)~~
2. ~~Inter-thread communication (VTSC)~~
3. ~~External Timing Information~~

Attack Surface for MA Attacks



Security Measurement of TimeWarp

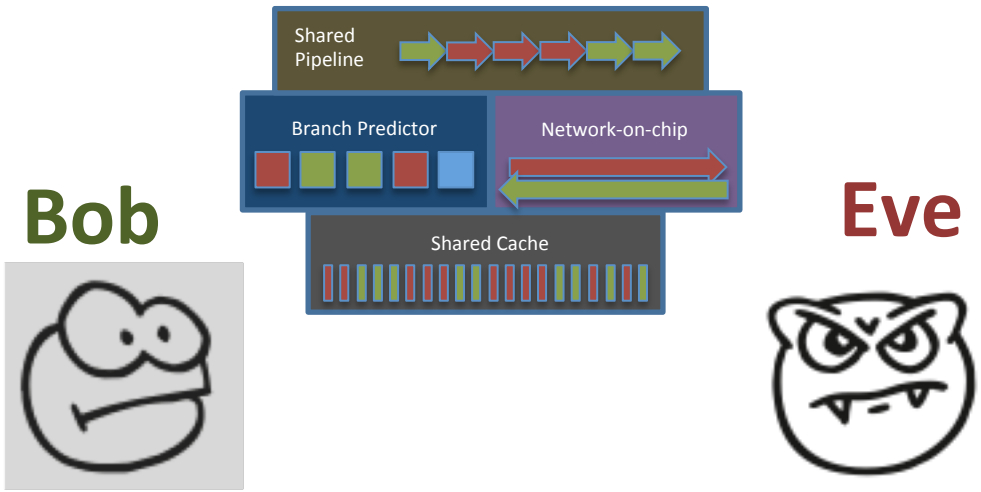


Limitations

- Only mitigates software-based attacks.
- Coarse timing attacks may still work.
  - But known attacks do not easily map to throughput attacks.
  - Throughput attacks should be easier for programmers to avoid.
- Averaging multiple runs may still work.
  - Much harder, because of our offsets and delays.
  - Many attacks require ‘lucky’ runs, which are rare, and hard to distinguish.

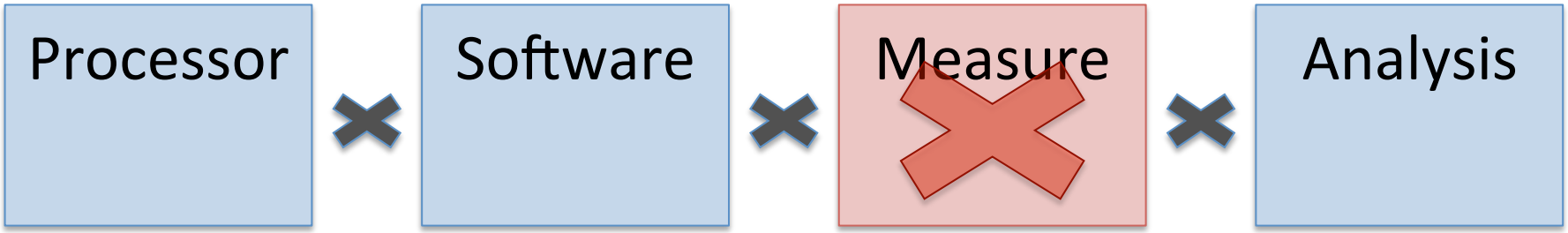
Conclusion

- **MA side channels are a dangerous problem.**



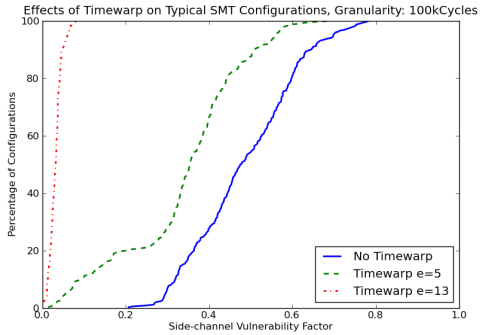
Conclusion

- **MA side channels are a dangerous problem.**
- **TimeWarp obscures measurement of MA events.**



Conclusion

- MA side channels are a dangerous problem.
- TimeWarp obscures measurement of MA events.
- SVF measurements indicate it is effective.



Conclusion

- MA side channels are a dangerous problem.
- TimeWarp obscures measurement of MA events.
- SVF measurements indicate it is effective.
- TimeWarp allows microarchitects to develop high-performance designs without worrying as much about side channels.

Backup Slides

Measurement Instruments in MA Attacks

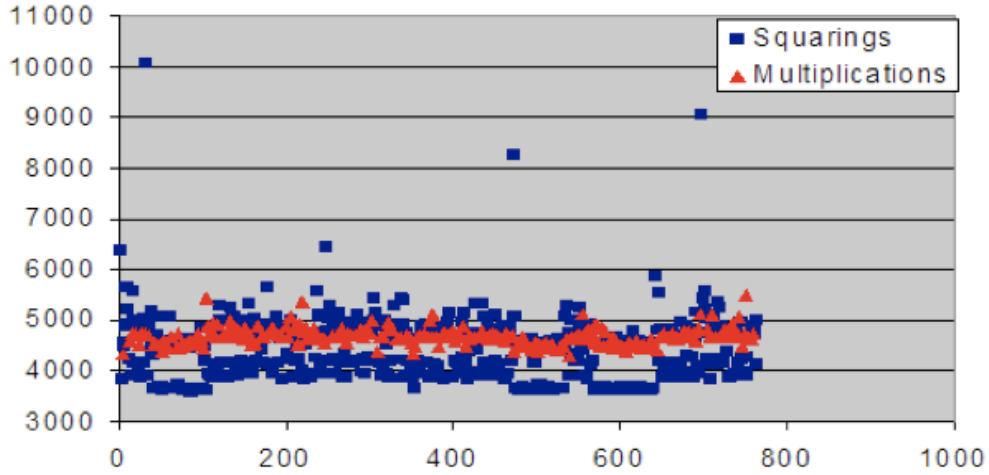
Previously Published Attacks

Year	Target	Authors	Hardware	Software	Measure method
2005	d-cache	Percival	Pentium 4	OpenSSL RSA	**RDTSC**
2005	d-cache	Bernstein	Pentium III	OpenSSL AES	**RDTSC**
2006	BPU	Aciicmez…	Pentium 4	OpenSSL RSA	**RDTSC**
2007	i-cache	Aciicmez…		OpenSSL RSA	**RDTSC**
2010	d-cache	Jayasinghe…		OpenSSL AES	**RDTSC**
2011	d-cache	Bangerter…	Pentium 4	OpenSSL AES	**RDTSC**

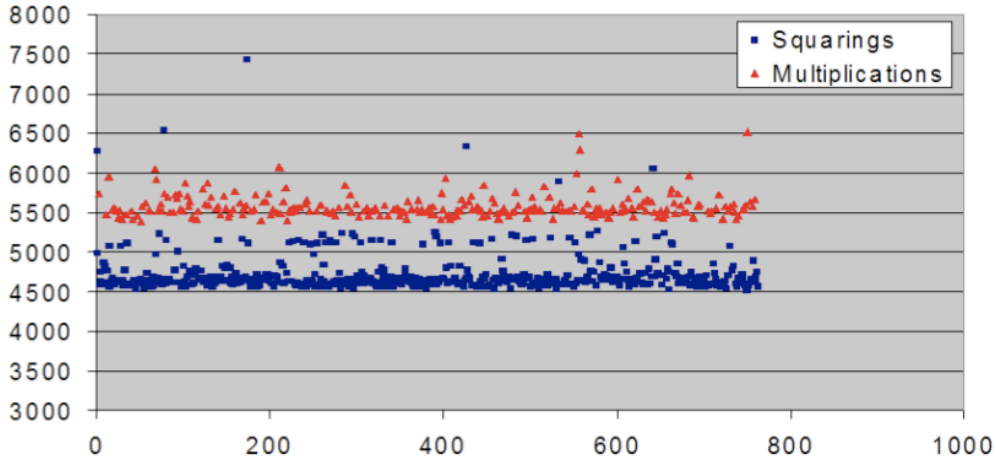
Coarse Timing Attacks

- Attacks requiring coarse timing still possible
  - 1 second vs. 5 seconds
- Reconfigure fuzzing to mask new attack?
- Rely on programmers to avoid these mistakes.

Averaging Multiple Runs



Normal



"Lucky"