

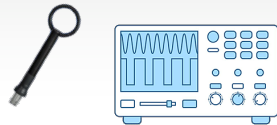
Hot Pixels: Frequency, Power, and Temperature Attacks on GPUs and Arm SoCs

Hritvik Taneja, Jason Kim, Jie Jeff Xu
Stephan van Schaik, Daniel Genkin, Yuval Yarom

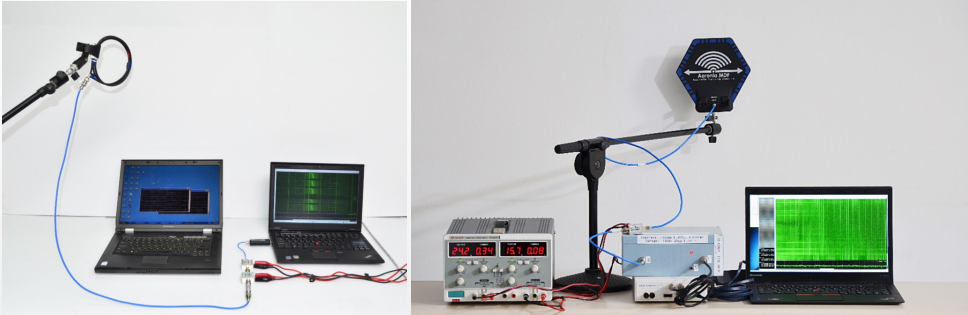


Evolution of Side-Channel Attacks

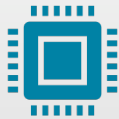
Physical Side-Channels



- **Equipment:** Oscilloscope, EM Probes etc.
- **Advantages:** Difficult to mitigate, usually works across different microarchitectures



Microarchitectural Side-Channels



- **Equipment:** Software
- **Advantages:** No physical proximity, can attack remotely



Hybrid Side-Channels

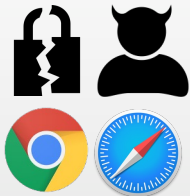


- **Equipment:** Software
- **Advantages:** Can attack remotely, difficult to mitigate, usually works across different microarchitectures

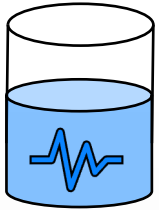
In this work, we show



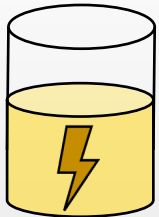
- Hybrid side-channels are everywhere
- Exploit them from unprivileged native user
- And, even from browser!



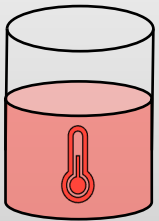
A sneak peek of our work



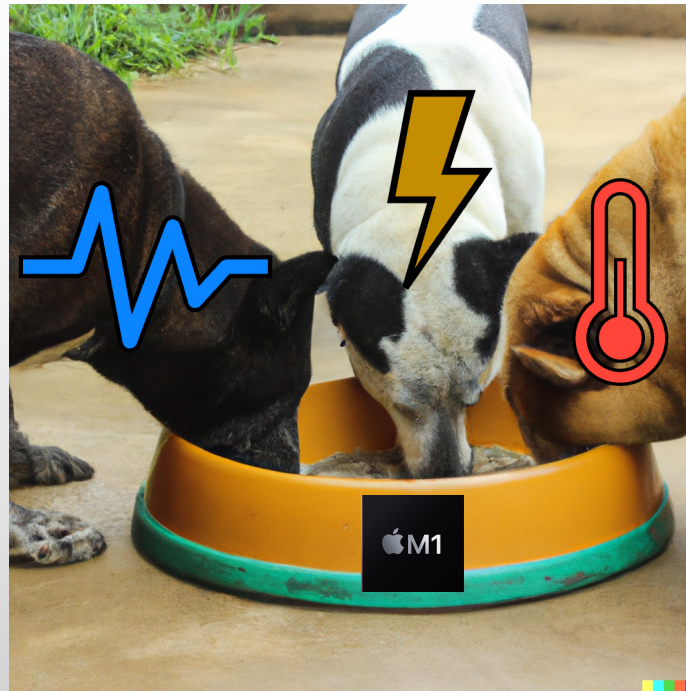
Frequency



Power



Temperature

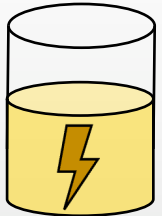


A sneak peek of our work

When one property becomes an operational constraint, other two leak



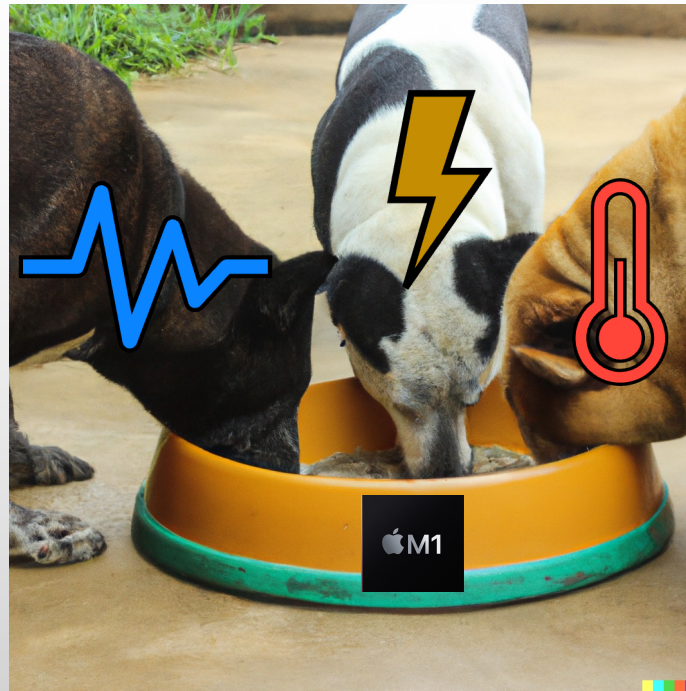
Frequency



Power



Temperature



- Pixel Stealing
- History Sniffing
- Website Fingerprinting



Original

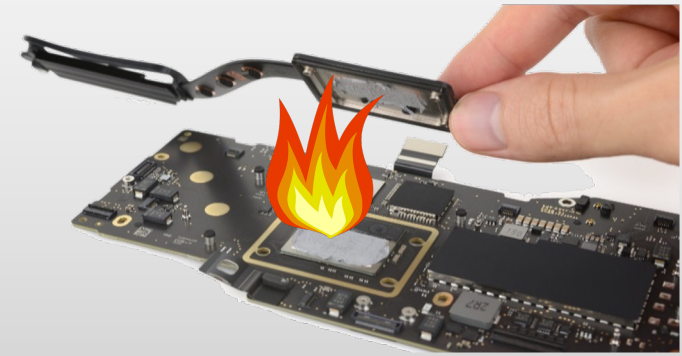
Leaked

Machine Cooling

- The more transistors we put on a chip the hotter it runs
- We need big heat sinks to keep the CPU cool and running
- Which makes laptops big and bulky, we don't like that!
- Dynamic Voltage and Frequency Scaling (DVFS)
 - Power or Thermal Limits
 - Run fast till it gets too hot
 - Slow and cool down. Repeat
- Does this idea have any security consequences?



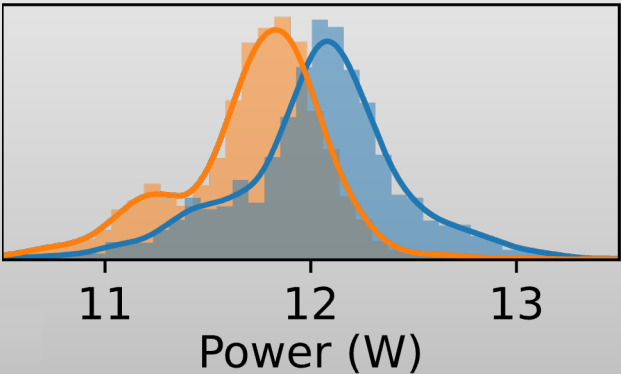
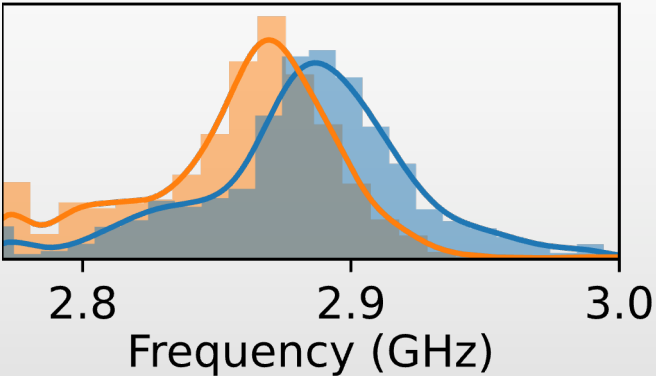
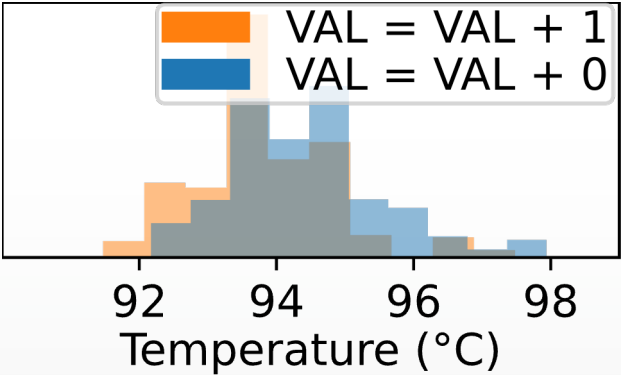
Typical cooling configuration



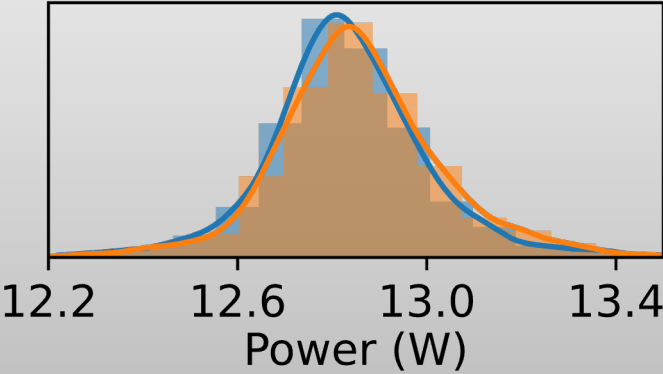
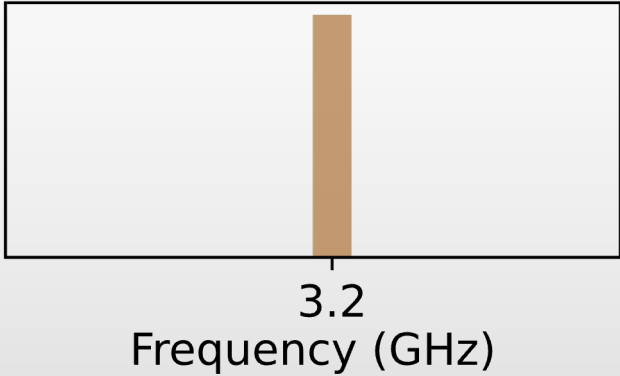
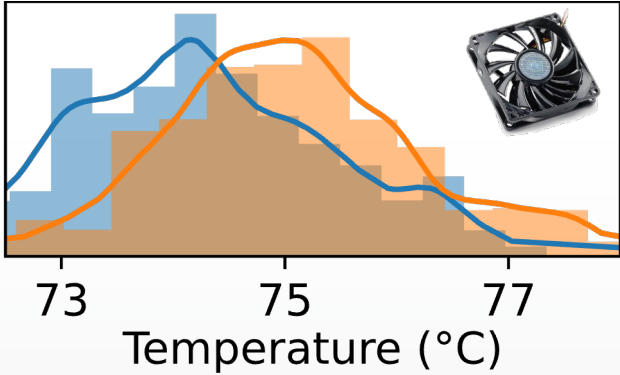
Mac motherboard

Is DVFS data-dependent?

M1 MacBook Air



M1 MacBook Pro



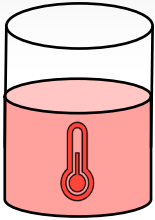
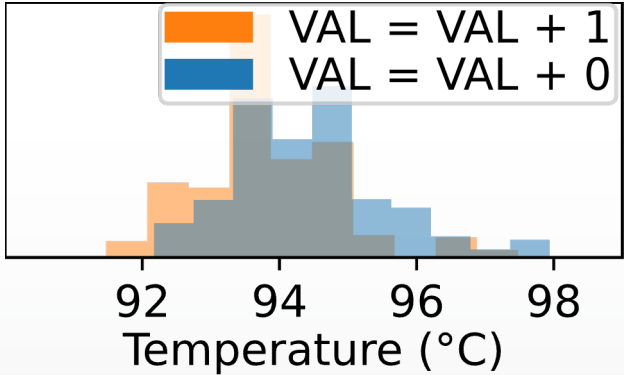
```
uint64_t val = 0;  
  
while (1)  
{  
    val = val + 1;  
}
```



```
uint64_t val = 0;  
  
while (1)  
{  
    val = val + 0;  
}
```

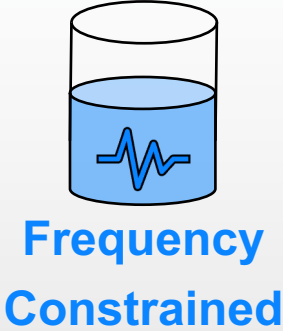
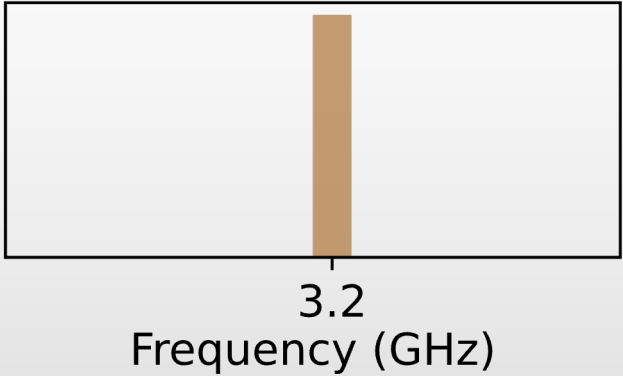
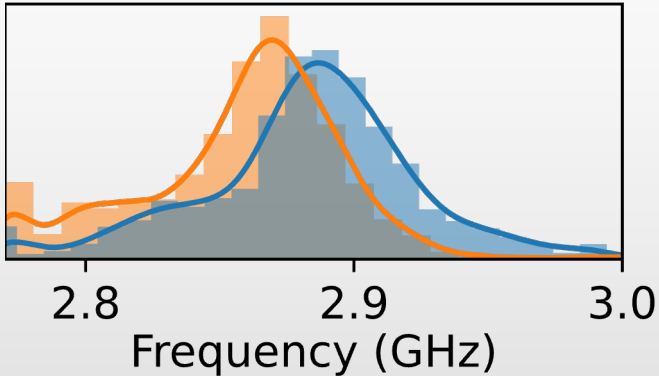
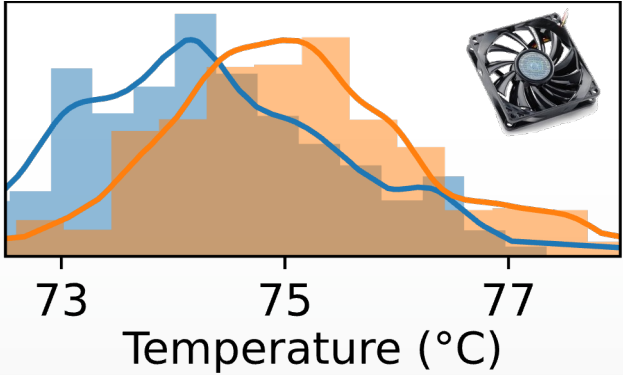
Is DVFS data-dependent?

M1 MacBook Air

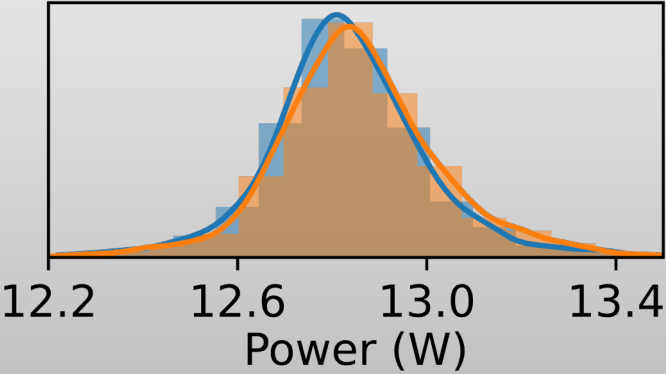
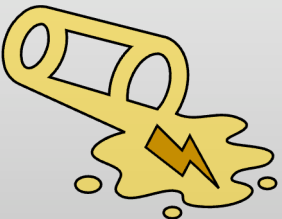
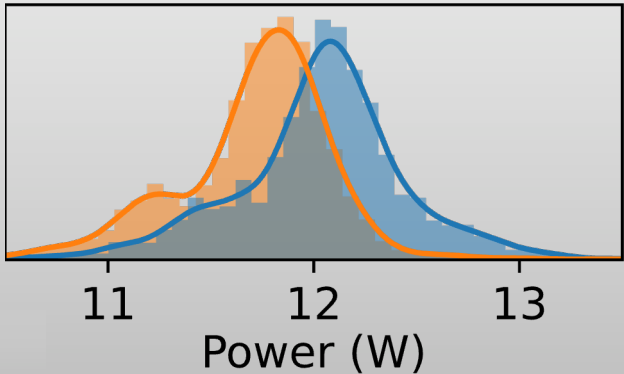


Thermally
Constrained

M1 MacBook Pro



Frequency
Constrained

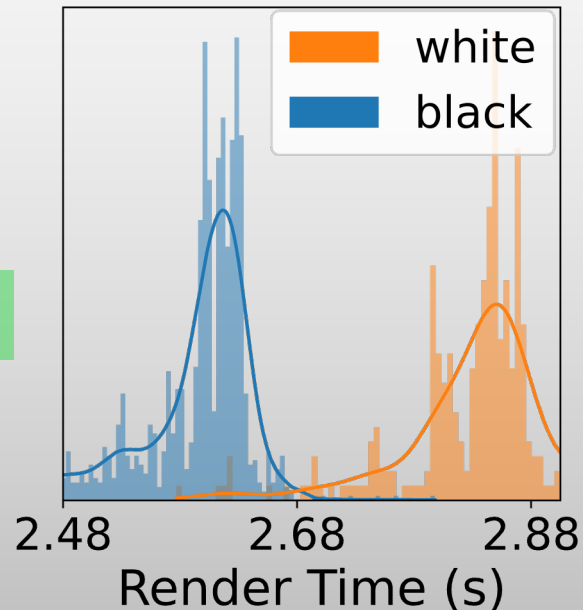


What about GPUs?

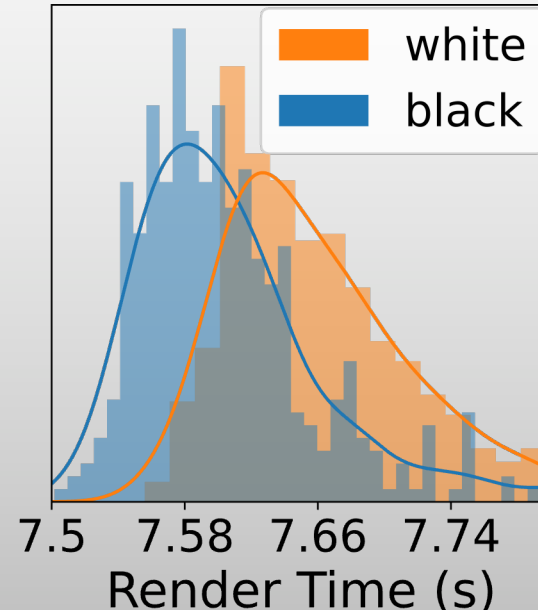
- GPUs have 1000s of cores
- They run hot
- They have massive cooling fans
- But is it enough to cool it?
 - No ☹️
- GPUs also implement DVFS 😊



M1 MacBook Air



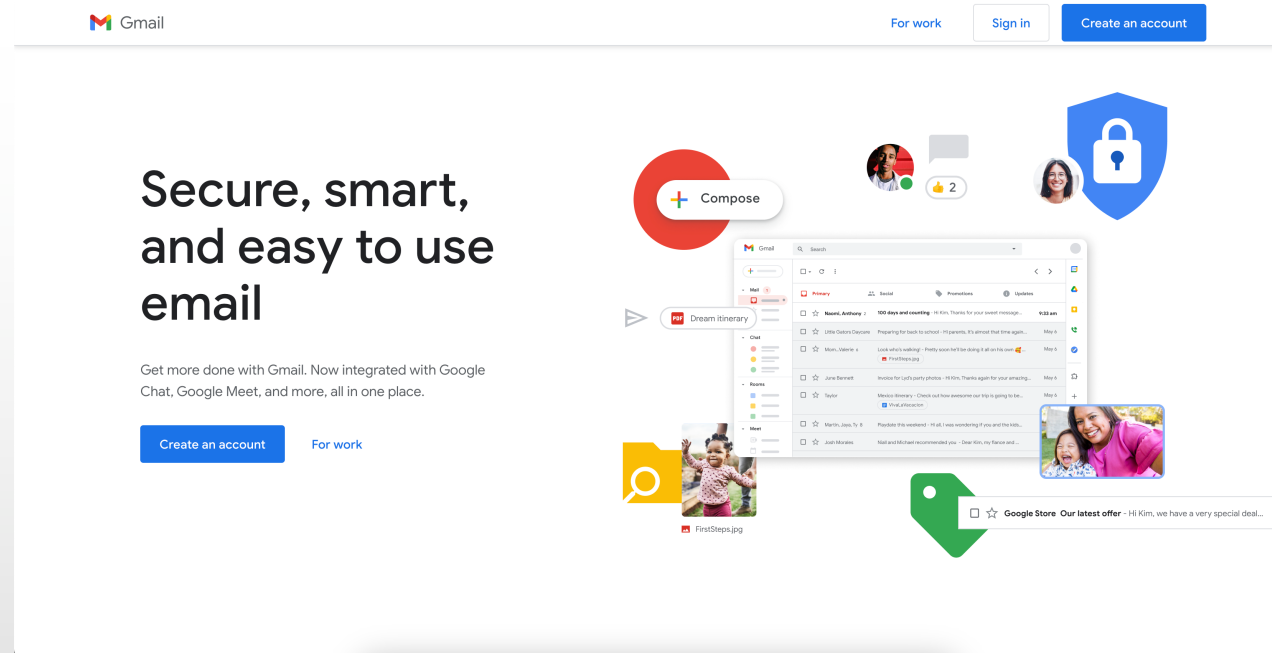
AMD Radeon RX 6600





Pixel stealing attack in Chrome – Primer

- SVG Filters can turn this ...





Pixel stealing attack in Chrome – Primer

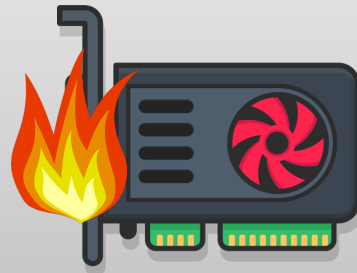
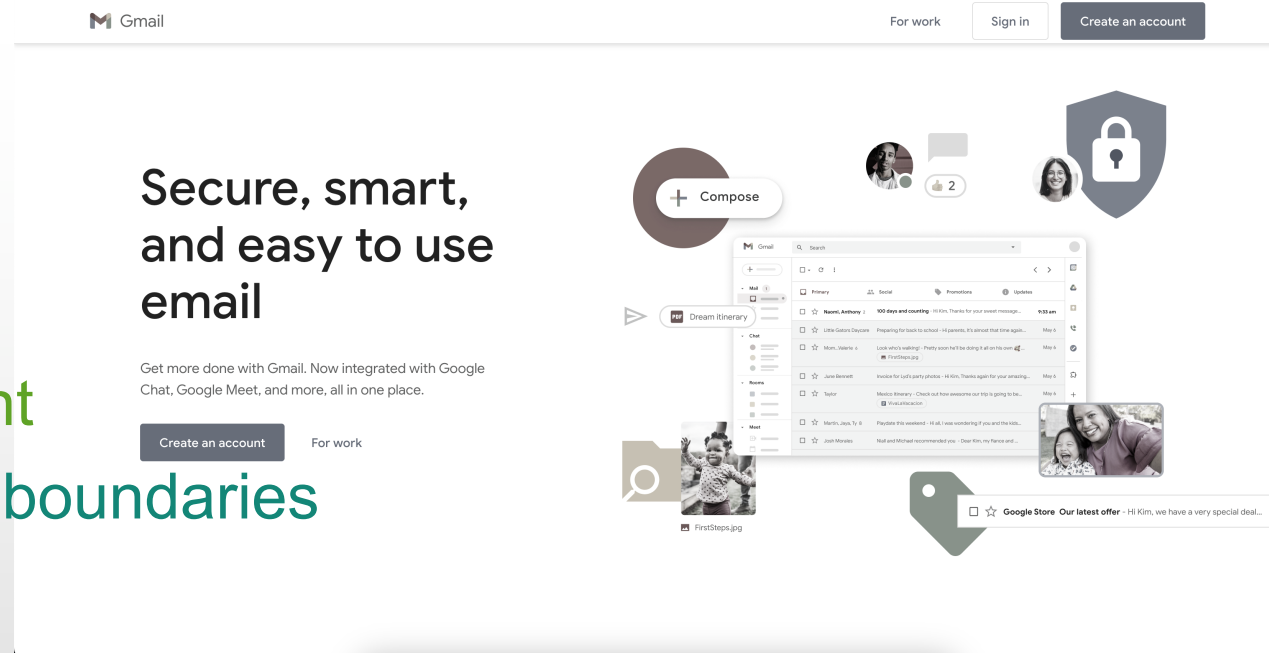
- SVG Filters can turn this ...
- Into this, using `<feGaussianBlur stdDeviation="3"/>`
- Various effects:
 - `<feGaussianBlur>` → blur
 - `<feColorMatrix>` → color saturation





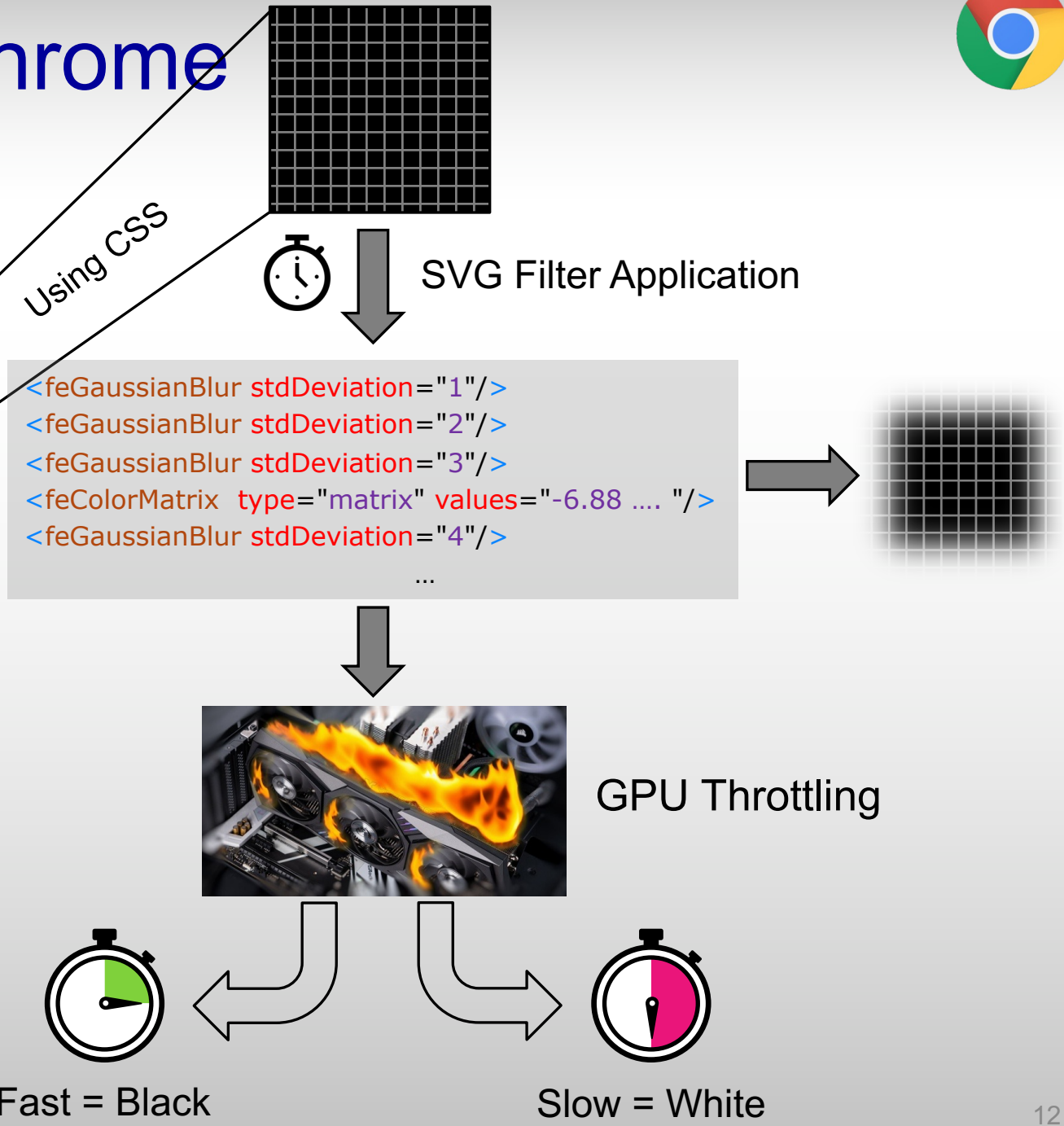
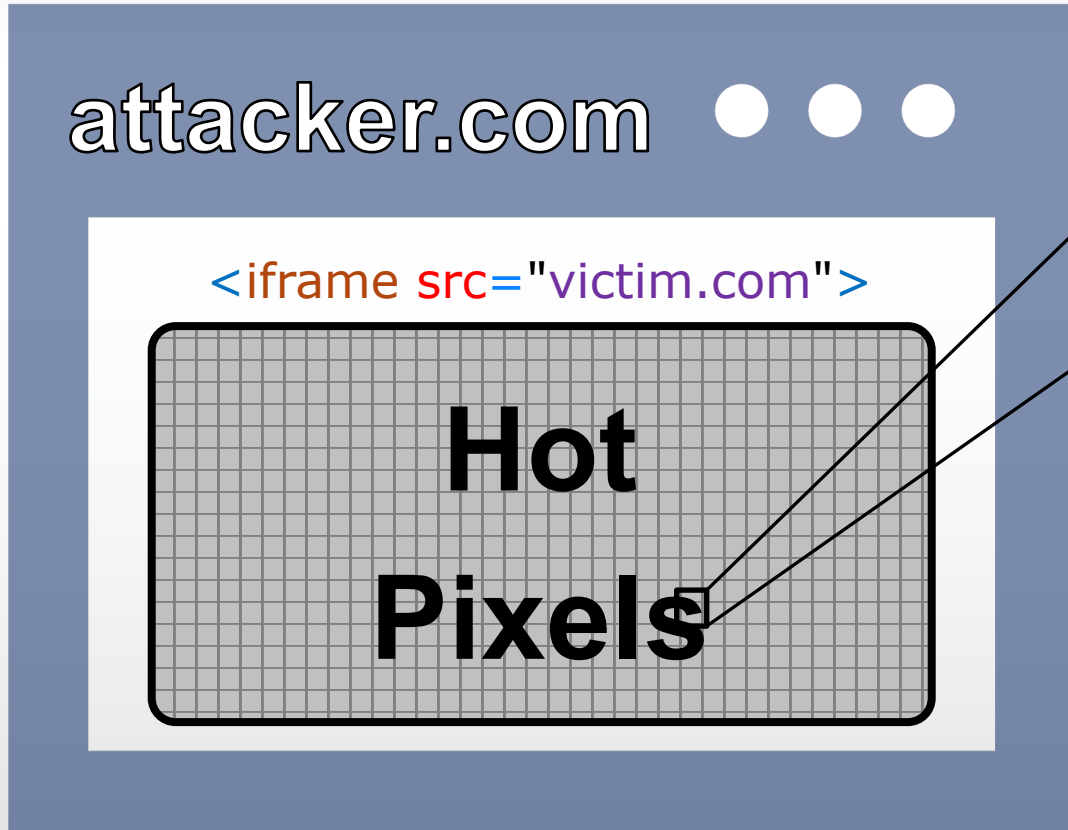
Pixel stealing attack in Chrome – Primer

- SVG Filters can turn this ...
- Into this, using `<feGaussianBlur stdDeviation="3"/>`
- Various effects:
 - `<feGaussianBlur>` → blur
 - `<feColorMatrix>` → color saturation
- Can be applied to:
 - `<div>`, `<iframe>` or any other element
- Can compute on pixels across security boundaries
- Stackable! 🍷
 - Stack until the GPU catches fire
 - Or in our case just throttles





Pixel stealing attack in Chrome





Pixel stealing attack in Chrome - Results



Original

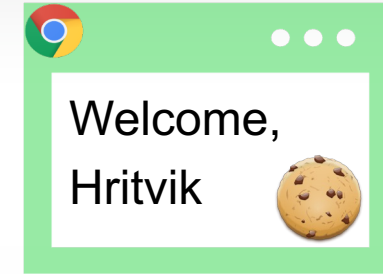
- **Works on:** Laptops, Phones with integrated GPUs and desktops with discrete GPUs.
- **Time to steal a pixel:** 10-20 sec
- **Accuracy:** 60-80%








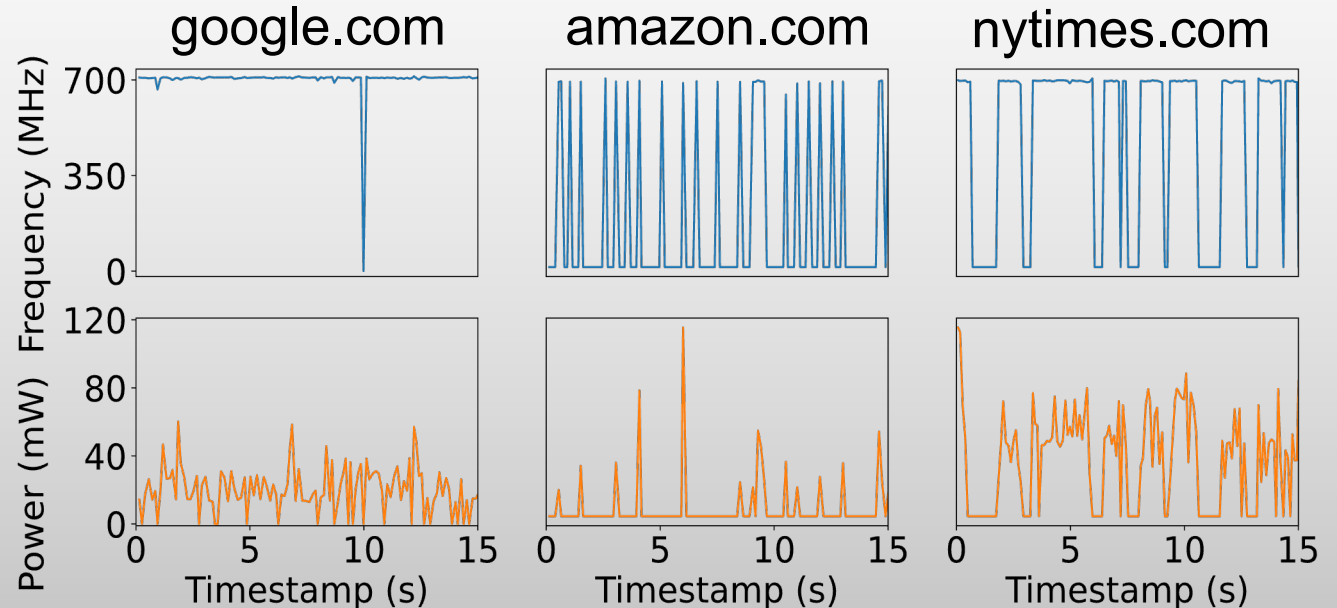
History sniffing attack in Safari

- Safari does not serve cookies across iframes
- Countermeasure to Pixel Stealing
 - Can't steal what is already public
- Can we steal something?
- Users' Browsing History
- attacker.com embeds links (<a>)
- Isolate a pixel and perform Pixel Stealing
- **Works on:** M1, M2 MacBook Air and iPhone 12, 13
- **Time to recover per link:** 200-250 seconds
- **Accuracy:** 90-100%



Website Fingerprinting

- macOS provides API measure GPU frequency and power
- Accessible by unprivileged user  
- This is a huge security problem 
 - Any random person can read your frequency and power
- Fingerprint Websites accessed by colocated user
- **Accuracy / 100 Websites:**
 - **Top 1: 27%**
 - **Top 2: 37%**
 - **Top 5: 49%**
 - **Baseline: 1%**



Summary

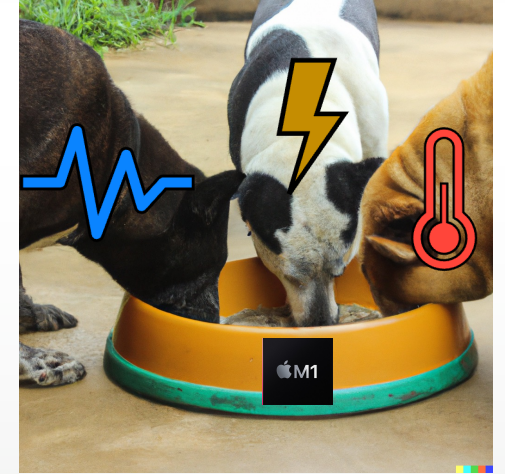
- 3-way tradeoff between Frequency, Power and Temperature
- Data Dependent CPU and GPU throttling behavior
- Attacks across multiple devices
- Pixel Stealing, History Sniffing, Website Fingerprinting



Original



Leaked



Thank you for listening!
Questions?

htaneja3@gatech.edu

