Lecture 24: Cache, Memory, Security

• Today’s topics:
  ▪ Caching policies
  ▪ Main memory system
  ▪ Hardware security intro
Cache Misses

• On a write miss, you may either choose to bring the block into the cache (write-allocate) or not (write-no-allocate)

• On a read miss, you always bring the block in (spatial and temporal locality) – but which block do you replace?
  ➢ no choice for a direct-mapped cache
  ➢ randomly pick one of the ways to replace
  ➢ replace the way that was least-recently used (LRU)
  ➢ FIFO replacement (round-robin)
Writes

- When you write into a block, do you also update the copy in L2?
  - write-through: every write to L1 → write to L2
  - write-back: mark the block as dirty, when the block gets replaced from L1, write it to L2

- Writeback coalesces multiple writes to an L1 block into one L2 write

- Write-through simplifies coherency protocols in a multiprocessor system as the L2 always has a current copy of data
Types of Cache Misses

• Compulsory misses: happens the first time a memory word is accessed – the misses for an infinite cache

• Capacity misses: happens because the program touched many other words before re-touching the same word – the misses for a fully-associative cache

• Conflict misses: happens because two words map to the same location in the cache – the misses generated while moving from a fully-associative to a direct-mapped cache
Off-Chip DRAM Main Memory

• Main memory is stored in DRAM cells that have much higher storage density

• DRAM cells lose their state over time – must be refreshed periodically, hence the name *Dynamic*

• A number of DRAM chips are aggregated on a DIMM to provide high capacity – a DIMM is a module that plugs into a bus on the motherboard

• DRAM access suffers from long access time and high energy overhead
Memory Architecture

- DIMM: a PCB with DRAM chips on the back and front
- The memory system is itself organized into ranks and banks; each bank can process a transaction in parallel
- Each bank has a row buffer that retains the last row touched in a bank (it’s like a cache in the memory system that exploits spatial locality) (row buffer hits have a lower latency than a row buffer miss)
Hardware Security

• Software security: key management, buffer overflow, etc.

• Hardware security: hardware-enforced permission checks, authentication/encryption, etc.

• Security vs. Privacy

• Information leakage, side channels, timing channels

• Meltdown, Spectre, SGX
Meltdown
Spectre: Variant 1

Victim Code

\[
\text{if } (x < \text{array1}_\text{size}) \\
y = \text{array2}[\text{array1}[x]];
\]

- \( x \) is controlled by attacker
- Access pattern of \( \text{array2}[\ ] \) betrays the secret
- Thanks to bpred, \( x \) can be anything
- \( \text{array1}[\ ] \) is the secret
Spectre: Variant 2

Attacker code

Label0: if (1)

Label1: ...

Victim code

R1 ← (from attacker)
R2 ← some secret
Label0: if (…)

… ...

Victim code

Label1:

lw [R1]
or
lw [R2]