Security-Aware Processor Architecture Design
CS 6501 – Fall 2018
Ashish Venkat
Hi, this is Ashish!

- Assistant Professor in the Dept. of Computer Science at UVA
- Ph.D. from University of California, San Diego
- Research interests: Computer Architecture, Compilers, Computer Security
Why study CS 6501?

The microprocessor mediates nearly every aspect of human life.
Why study CS 6501?

The microprocessor industry is at a crossroads ...

Architect’s metric of importance: High Performance
Why study CS 6501?

The microprocessor industry is at a crossroads...

Architect’s metric of importance: High Performance, Power/Thermal Efficiency, Security, Programmability
The Tension between Performance and Security

- SPECTRE
- 35 LAWSUITS FILED AGAINST CPU COMPANY
- NEW SIDE-CHANNELS
- PROCESSOR STOCK MARKET DROPPED
- HEARTBLEED
- ROWHAMMER
- MELTDOWN
The goal of this course is to ... study, motivate, and propose the next generation of security-aware processor architectures.
Agenda

• Course Learning Goals
• Course Structure, Prerequisites, Grading, and Logistics
• Course Milestones
• Quick Processor Architecture Review
• Research Themes
Course Learning Goals

• **Related Work:** To become conversant with security issues and concerns that plague the modern microprocessor industry, and understand state-of-the-art defense mechanisms.

• **Motivation:** To identify new processor flaws and/or motivate new solutions to existing security threats.

• **Mechanisms:** To design novel security-aware high performance processor architectures.

• **Methodology:** To design comprehensive experiments that measure the effectiveness of the proposed architecture.

• **Evaluation:** To extensively evaluate the proposed architecture.

*This course is structured around the sections of a scientific paper.*
Course Learning By-products

• **Research Methodology:** Gain/hone research skills
• **Breadth:** Work on cross-disciplinary CS areas
• **Collaboration:** Work together to accomplish the set learning goals
• **Publishable Paper/Thesis:** Opportunity to boost your CV
• **Funding:** Do well in this class and get funded next semester!
Course Structure

• We will explore five research themes related to processor security.

• Students will form groups to pursue a research project in one of the five identified research themes.

• **End goal:** Project report that will eventually morph into a top-tier conference publication.

• **Milestones:**
  • Assignment 1: Related Work
  • Assignment 2: Motivation/Feasibility Study
  • Assignment 3: Mechanisms/Prototype Implementation
  • Assignment 4: Methodology/Pencil-sketch graphs
  • Assignment 5: Results
Class Structure

- **Today:** Overview, motivation, and some background
- **Thursday:** More project-specific background
- **Next week onwards:**
  - Representative members from each student group will present papers from their chosen research themes (more details on this in next class).
  - Rest of the class will participate in discussions – required and more important than the presentation itself (accounts for a higher grade percentage).
  - As we progress through the class, we will talk less about related work and more about your project – ideas, experiments, issues, and challenges.
Grading

• 85% of the grade will go towards the project
  • Spilt equally across all five assignments (related work, motivation, mechanisms, methodology, and evaluation)

• 15% will go towards class participation
  • Most of this will go towards how you interact and participate in brainstorming sessions during the class, in office hours, and on Piazza (our class forum).
  • You do not have to do a great job in terms of paper presentation, but if you come unprepared to lead a discussion, you will lose most or all of this 15%.
Prerequisites

• Open to all graduate students in CS, CpE, and ECE.

• Third and Fourth year undergraduate students should meet a minimum prerequisite requirement of the undergraduate computer architecture course CS 3300 or equivalent.

• Graduate students who focus on other complimentary CS disciplines are encouraged to enroll, but are expected to pick up relevant architecture background as we progress through the course.

• This course satisfies breadth requirements under the "Computer Systems" category.
Course Logistics

• Office hours:
  • Tu/Th 11am-12pm @ Rice 312 (right after class)
  • In addition to these, we will also schedule group-specific weekly project meetings.

• Class website:
  • http://www.cs.virginia.edu/venkat/classes/cs6501/fa18/

• Class forum:
  • https://piazza.com/virginia/fall2018/cs6501

• Email: venkat@virginia.edu
Email and Piazza

• When to use email?
  • Private questions (e.g., asking about your grade/standing in class)
  • Assignment submissions
  • Express Feedback: Grades available the next day of submission.

• When to use Piazza?
  • For all other discussions
  • Project abstracts
  • Slides
  • Useful links and resources
Project Meetings

• I will have weekly project meetings with each student group in addition to my office hours.

• These are optional and you can cancel at a day’s notice.

• I will provide significant research mentorship during these meetings.

• I will also use these meetings to discuss your grades and help you make forward progress through the course.
UVA Honor Code

• All students at UVA are required to abide by the honor code and pledge to not commit academic fraud.

• What you can do?
  • Collaboration and brainstorming of ideas is encouraged both within and outside your group.
  • You’re also free to lookup the internet for papers/presentation slides and use open-source software for your projects.

• What you’re not allowed to do?
  • Plagiarize text from someone else’s assignment or the internet.
  • Falsify data.

• Cheating will be taken seriously and consequences will be severe!
Agenda

• Course Learning Goals
• Course Structure, Prerequisites, Grading, and Logistics
• Course Milestones
• Quick Processor Architecture Review
• Research Themes
Course Milestones

• Project abstracts with 5 related papers have been posted on Piazza for each of the 5 research themes.

• I will provide more overview and background today and on Thursday.

• **Milestone 0 – Project Selection (due 8/30 11:59:59pm):**
  
  • Make a Piazza post with your group name, group members, and chosen research theme.
  
  • Note that you’re not required to read all 25 papers listed thoroughly before you make a choice – just read the paper abstracts and introduction to get a feel for the problem statement and current state-of-the-art.
  
  • You can also suggest your own project as long as you convince me of its novelty and relevance.
Course Milestones

• **Milestone 1 – Related Work (due 9/13 11:59:59pm):**
  • Go beyond the 5 listed research papers for your chosen research theme.
  • Make sure you come talk to me about the set of papers you’ve chosen to study during our designated project meeting.
  • Write an end-to-end one-page survey report for your chosen research theme.
  • You have two weeks for this.
Course Milestones

• **Milestone 2 – Motivation (due 9/27 11:59:59pm):**
  
  • Brainstorm amongst yourselves and with me for a solution (look for hints today and on Thursday).
  
  • Come up with one or at most two experiments that can evaluate the *feasibility* of your idea – more help regarding this during office hours and project meetings.
  
  • Present the experiment and results in a one-page written report.
  
  • You have two weeks for this.
Course Milestones

• **Milestone 3 – Mechanisms (due 10/23 11:59:59pm):**
  
  • Design a prototype for the suggested solution in milestone 2 in the language of your choice, and using the tools of your choice.
  
  • Document your design mechanisms in a 2-4 page report.
  
  • No experiments/results are required at this point of time. But thoroughly think through all possibilities – attack and legitimate execution scenarios.
  
  • You have almost one month for this.
Course Milestones

• **Milestone 4 – Methodology (due 11/6 11:59:59pm):**
  
  • Discuss what experiments you will run to evaluate your prototype solution.
  
  • Describe these experiments and provide pencil sketch graphs that show your expectation of the results – these graphs do not have to be exact or even in the ballpark range of the actual results.
  
  • You are just designing experiments, not running them yet!
  
  • Need both security and performance evaluation.
  
  • Need about 8-10 experiments that evaluate different aspects of your solution and compare against state-of-the-art.
  
  • Deliverable: 1-2 page report documenting all of this.
  
  • You have two weeks for this.
Course Milestones

• **Milestone 5 – Results (due 12/6 11:59:59pm):**
  • Perform all the experiments you’ve listed in milestone 4 and document your results.
  • Make inferences about your results with appropriate rationale.
  • Deliverable: 3-5 page evaluation report.
  • You have exactly one month for this.
Agenda

• Course Learning Goals
• Course Structure, Prerequisites, Grading, and Logistics
• Course Milestones
• Quick Processor Architecture Review
• Research Themes
The classic 5-stage MIPS pipeline

Characteristics:

• RISC Architecture
• Simple Decoding
• Inorder Pipeline
• Scalar Pipeline
Superscalar MIPS pipeline

Characteristics:

• RISC Architecture
• Simple Decoding
• Inorder Pipeline
• Superscalar Pipeline
Modern CISC Architectures

Characteristics:

• Large Front-End

• Complex Decoders

Native Instructions (e.g., inc [0x803ac])
Modern CISC Architectures

Characteristics:

• Large Front-End

• Complex Decoders

Native Instructions (e.g., inc [0x803ac])

Modern CISC Architectures

Characteristics:

• Large Front-End

• Complex Decoders

Native Instructions (e.g., inc [0x803ac])
Typical Intel Front-End

L1 I cache

Instruction fetch unit

16 byte line

Instruction Length Decoder / Pre-decode

18 entry Instruction Queue

Can dispatch up to 5 instructions per cycle

Decoders:
- 1:1 decoder
- 1:1 decoder
- 1:1 decoder
- 1:4 decoder

MSROM

Micro-op queue

Maximum of 4 micro-ops per cycle

To Execution

start of loop

end of loop

loop stream detector

micro-ops

N

Y

hit?

Program Counter (PC)

Micro op Cache

8-way 1.5K micro ops
Dynamic Speculation and OOO Execution

- Execution follows data dependences
- Instructions may execute out-of-order
- More instruction-level parallelism may be exploited across control-flow boundaries via dynamic branch prediction
Security-Aware Processor Architecture Design

CS 6501 – Fall 2018

Ashish Venkat