What Formal Methods?

- Formal methods enable reasoning from logical or mathematical specifications of the behaviors of computing devices or processes.
- They offer rigorous proofs that all system behaviors meet some desirable property.
- Exemplars:
  - Contemporary cryptography relies on formal methods in this broad sense.
  - Synthesis of secure programs and other correct-by-construction mechanisms (e.g., zero knowledge proofs) also use formal methods.
- Mature set of tools and used in many other domains.
What Secure Architecture?

- Just like constructing a building, we start off by creating a blueprint
- Architecture → Detailed Design → Implementation → Testing and verification
- Make explicit the security goals and adversary model
- Illuminate the dependencies on other infrastructures
  - What are the trust relationships?
  - What are the control and data flows across infrastructures?
  - What are the personnel allocation?

Why Would We Want Them for Security?

- **Formal methods:**
  1. Way to break out of “cat and mouse” game between adversaries and defenders
  2. Privacy guarantees need to be that – guarantees
  3. Rich set of scalable verification tools being developed

- **Secure architecture:**
  1. Brings necessary rigor to the design process
  2. Can help align business processes with security processes
  3. Can be a pathway to application of formal methods
Why We Should Stay Away from Them for Security

- **Formal methods:**
  1. Needs too much expertise: modeling, formal specification, use of the tools
  2. Tools are not mature enough: breaks at large scale, cannot handle dynamism, too many false alarms
  3. Lengthens development time

- **Secure architecture:**
  1. Needs too much consensus when payoffs are in the distant future
  2. Needs too much expertise
  3. Lengthens development time

Presentation available at:
Dependable Computing Systems Lab (DCSL) web site
engineering.purdue.edu/dcs