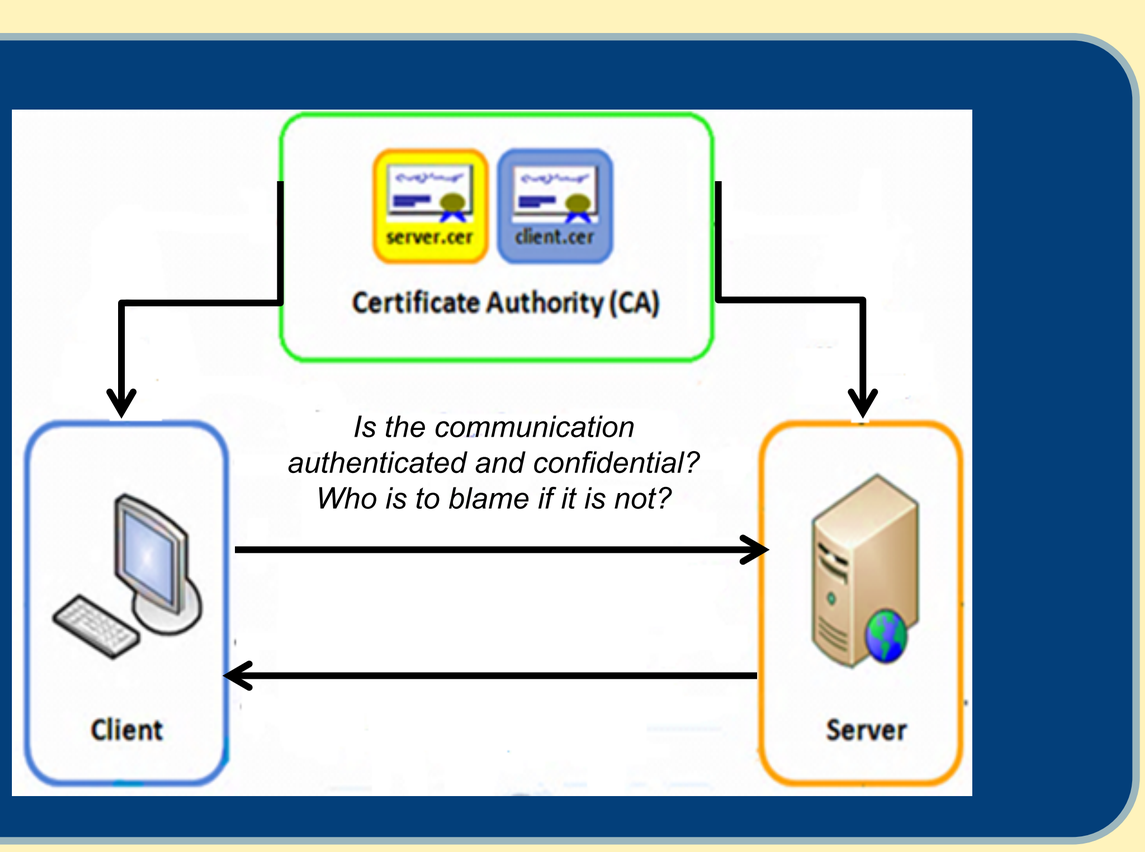
Accountability via Deviance and Causal Determination

PI: Anupam Datta Co-PIs: Limin Jia, Dilsun Kaynar, Frank Pfenning

Computer Science Department and Electrical and Computer Engineering Department, CMU

Project Goals

- Develop a language for distributed computing in which contracts are specified via types and a type-directed distributed monitoring infrastructure for detecting deviance from such contracts.
- Develop an extensional blame semantics based on causation
- Apply the developed methods to study accountability in the public key infrastructure



Approach

- Prescribe communication behavior between processes by **session types**
- **Dynamically monitor** communication to detect undesirable behavior
- Correctly **blame** the violating party
- Model protocol parties as programs and assume the availability of evidence of their actions in the form of a **log**
- Formalize **cause** using ideas from prior counterfactual definitions of actual causation

Blame Theorems

- When a violation is detected dynamically, one of the indicated set of culprits must have been compromised
- Dynamic monitoring does not change system behavior in well-typed processes

A set of program actions A on *log* L is a cause of violation V if:
Occurrence: Violation occurred on L
Sufficiency: Removing actions not in A still leads to V
Minimality: No proper subset of A satisfies the above condition

Selected Publications

- *Monitors and Blame Assignment for Higher-Order Session Types.* Jia, Gommerstadt, Pfenning In Proceedings of Principles of Programming Languages, 2016.
- *Program Actions as Actual Causes: A Building Block for Accountability.* Sharma, Datta, Garg, Kaynar, In Proceedings of Computer Security Foundations Symposium (CSF), 2015.
- *Interaction-aware Actual Causation: A Building Block for Accountability in Security Protocols,* CMU-ECE PhD Thesis by Divya Sharma

Interested in meeting the PIs? Attach post-it note below!