SoS Lablets

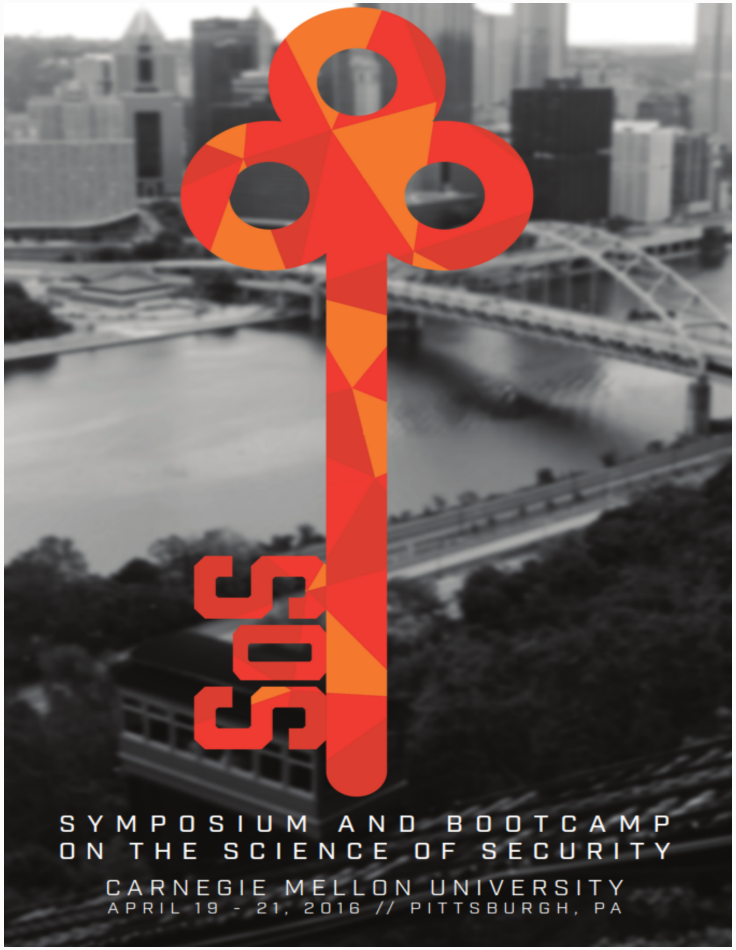
The Carnegie Mellon

Science of Security Lablet

Bill Scherlis (PI)
Quarterly Meeting
Pittsburgh
10-11 July 2017

The CMU Science of Security (SoS) Lablet

- The team
- Mission
- Significance of SoS
- Hard Problems
- Synergetic benefits
- CMU project portfolio



The team

- The CMU research team
  - David Garlan
  - Anupam Datta
  - Andre Platzer
  - Alessandro Acquisti
  - Christian Kästner
  - Travis Breaux – co-PI
  - Lorrie Cranor
  - Limin Jia
  - Rahul Telang
  - Bill Scherlis - PI
  - Jürgen Pfeffer
  - Jonathan Aldrich - co-PI
  - Bradley Schmerl
  - Nicolas Christin
  - *More than seven academic departments, three colleges*
  - *Diverse disciplines: Mathematical logics and models, software architectures/frameworks, graph-theoretic network analytics, human subjects studies, CPS, policy modeling, software devt and evaluation*
- Partner universities
  - Cornell (Dexter Kozen)
  - GMU (Sam Malek)
  - UC Berkeley (Serge Egelman)
  - U Pittsburgh (Scott Beach)
  - Wayne State (Marwan Abi-Antoun),
  - U Nebraska (Matt Dwyer, Witawas Srisa-an)
  - UTSA (Jianwei Niu)
- PhD and MS students
  - 16 PhD and 2 MS in multiple depts
- REU undergraduate students
  - Approx 8 (REU is NSF funded)

Other SoS Lablets
- UIUC
- NCSU
- U Md

Mission concept

(1) Advance **identified specific areas** of cybersecurity research

- **Scalability and composability**
- **Policy-governed secure collaboration**
- Predictive security metrics
- **Resilient architectures**
- **Human behavior**

Not comprehensive coverage of cybersecurity topics

(2) Advance the **scientific coherence** of the multidisciplinary body of cybersecurity technical results

- Methods
- Validation
- Productivity

(3) Engage and broaden the cybersecurity **technical community**

- Facilitate community and educational engagement
- Workshops and conference events: HotSoS conference

Significance of **SoS**

- Premises – security **operating** environment
  - Growth in urgency and criticality of cybersecurity
    - Common vulnerable tech base
  - The "natural world" of cybersecurity is unusual
    - Synthetic terrain
    - Systems we build but do not understand
    - Presence of active adversaries
    - Rapid pace of change of systems and operating environment and threat
  - Multidisciplinary character of research necessary to advance capability
    - *Diverse technical domains*:
      - Biometrics, human behavior, crypto math, protocol analysis, language foundations, logics and models, systems architecture, threat analysis, cyber-physical models, networking, hardware, API design, measurement,…
    - *Adversary escapes the abstractions*: We must continually broaden the scope of our models – and make side channels more expensive
  - Diverse scientific approaches underlie the research
    - Mathematically based theory
    - Data-driven empirical studies
    - Empirical behavioral studies: observational and interventional

Significance of **SoS**

- Premises – security **engineering** environment
  - Challenge to interweave science and engineering
    - Foundations for engineering practice
    - Techniques to assess and understand what we are building
  - Diverse points of potential intervention to improve security
    - Requirements, architecture, development, operations, sustainment
    - Evaluation and measurement
  - Complexity and interconnection in systems and organizations
    - Rich and diverse supply chains
    - Socio-technical ecosystems
      - Framework-and-app models
      - *Payloads and platforms*
    - Dynamism, AI-based systems, IoT and new CPS models, etc.
  - Product families – in time and space
    - Configurations (e.g., Linux on Android)
    - Ongoing evolution and need to rapid recertification
  - Rapid pace of change in systems
    - Rapid iteration in response to mission, technology, threat
    - Broad recognition of need for iteration

Consequence: Three part approach to SoS

1. *Address the most challenging **Hard Problem** areas*
2. *Advance the **process and methods by which science is done** and the engineering that builds on it*
3. *Engage with the broader research and technical **community** to address these goals*

Three part approach to SoS

1. ***Address the most challenging Hard Problem areas***

- ***We focus primarily on HP 1 and 5***
- ***Strong activity related to HP 2 and 4.***
- ***(Details in HP Report)***

Opportunities:

- Obtaining and benefitting from common framing
- Identify transitions *already possible* to *engineering* practice
  - SoS then supports continued refinement of these

Risks

- Uncertainty regarding extent of benefit from common framing
- Difficulty of achieving common framing

Five Hard Problems in the Science of Security

1. **Scalability and composability**
2. **Policy-governed secure collaboration**
3. Predictive security metrics
4. **Resilient architectures**
5. **Human behavior**

CMU emphasis

Not comprehensive:

Focus on the engineering and evaluation of systems

Selection criteria for the problems

- High level of technical challenge
- Significant operational value
- Likelihood of benefiting from emphasis on scientific research methods and improved measurement capabilities

Three part approach to SoS

2. *Advance the process and methods by which science is done*

- ***Explicit focus on methods***
- ***Synergies in the Lablet approach***
- ***Towards a common base for analysis and engineering***

Opportunity: Methods and systematization

- Definition and validation of diverse “methods”
- Systematization of practices through study teams
  - (Cf. HP report process)
- Development of links with engineering practice
  - Analysis and synthesis

Risks: Management by the numbers

- Light under lamppost: TRLs, scientometrics, etc.

Synergies in the Lablet approach

- Data meets models
  - E.g., social network structures, developer usability, end-user usability, API complexity
- Semantics-based approaches meet real engineered systems
  - E.g., hypervisors, Web apps, framework+apps, large components
- Empirical science (data, people) meets mathematical reasoning
  - E.g., language design, API design, model design, tool design

Three part approach to SoS

3. ***Engage with the broader research and technical community to address these goals***

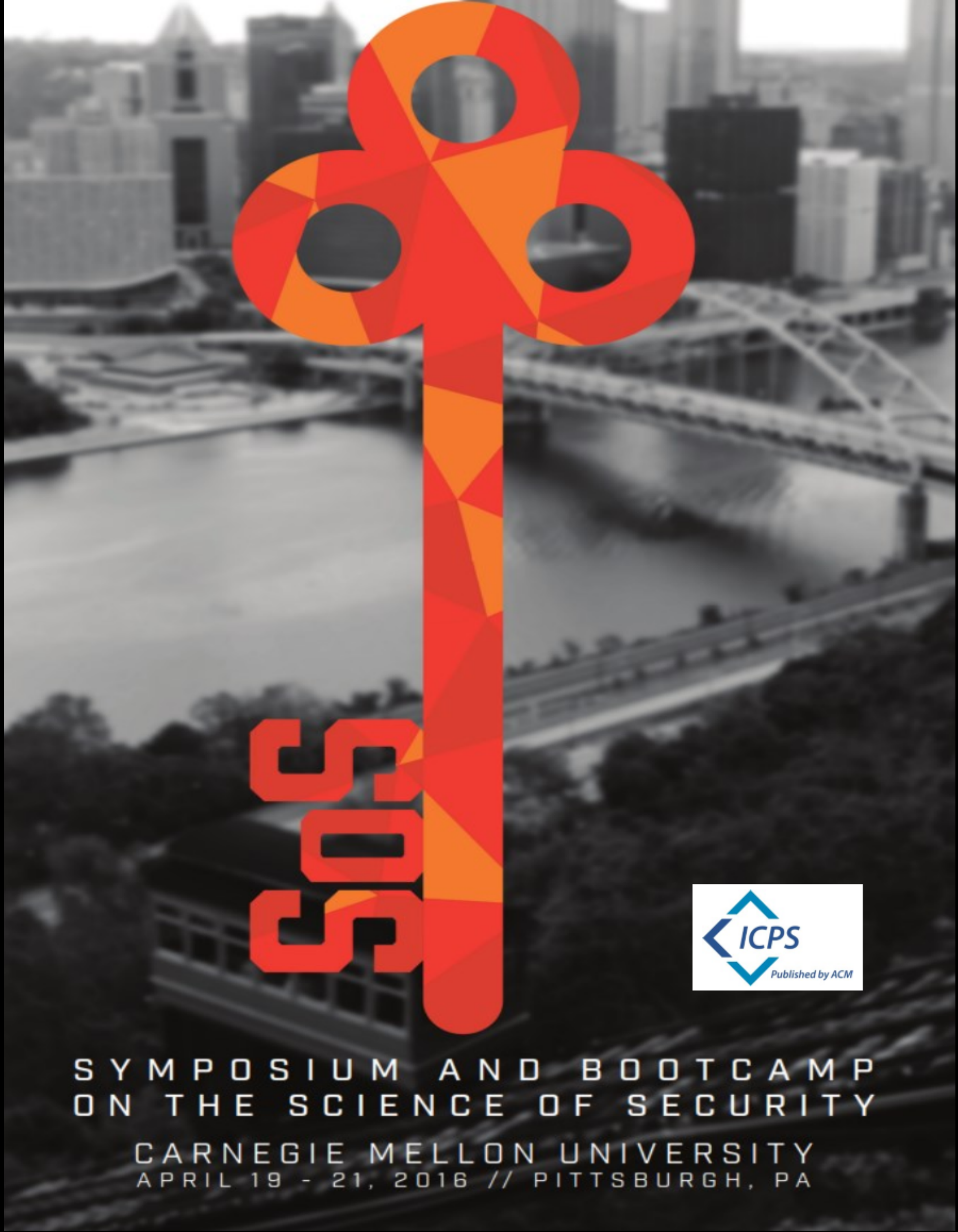
- ***HotSoS 2016 Conference***
- ***Conference on Safety and Control for AI (with OSTP)***
- ***Workshop on Safety and Control for AI (with Lablet)***

Opportunity: More active/dedicated community processes

- Connect more explicitly with engineering practice
  - Identify and test engineering principles
- Framing the Hard Problems
- Identification of common elements: methods, features, etc.

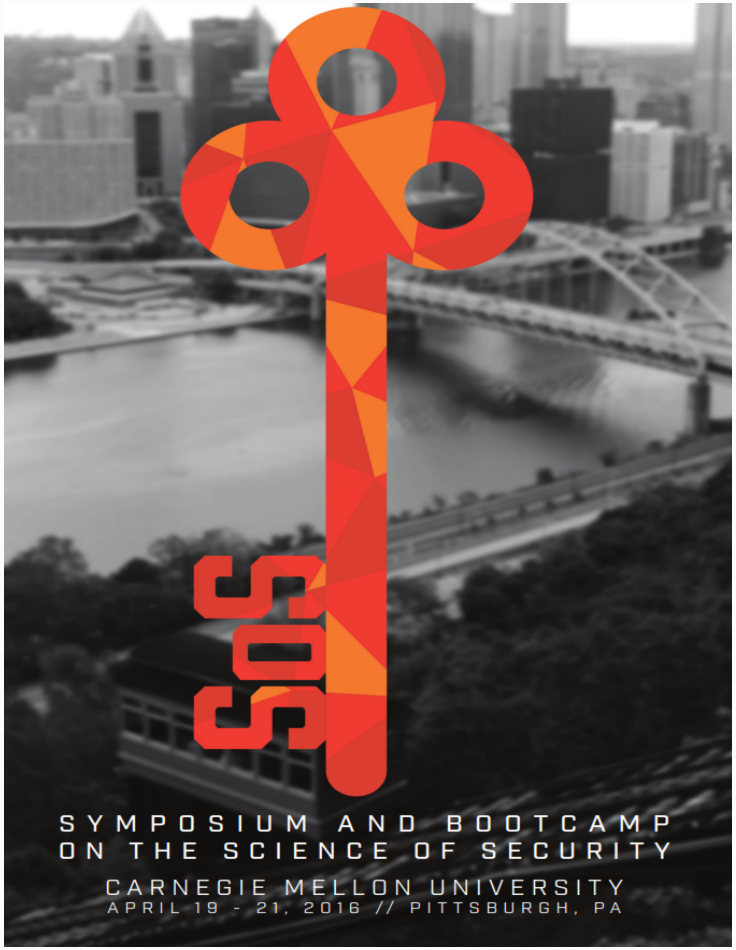
Risks:

- Failure to transition into engineering/evaluation practice


SoS
ICPS
Published by ACM
SYMPOSIUM AND BOOTCAMP
ON THE SCIENCE OF SECURITY
CARNEGIE MELLON UNIVERSITY
APRIL 19 - 21, 2016 // PITTSBURGH, PA

The CMU Science of Security (SoS) Lablet

- The team
- Mission
- Significance of SoS
- Hard Problems
- Synergetic benefits
- **CMU project portfolio**



CMU Projects (9 projects)

	Com	Pol	Met	Res	Hum		
1	X		X	X		Aldrich, Garlan, Malek (GMU), Abi-Antoun (Wayne State)	Frameworks, APIs, and Composable Security Models
2	X		X			Kästner	Limiting Recertification in Highly Configurable Systems: Interactions and Isolation among Configuration Options
3	X			X		Platzer, Kozen (Cornell)	Security Reasoning for Distributed Systems with Uncertainties
4	X	X				Datta, Jia	Compositional security
6	X				X	Aldrich, Sunshine	A Language and Framework for Development of Secure Mobile Applications
7	X		X	X		Garlan, Schmerl	Multi-model run-time security analysis
8	X				X	Aldrich, Dwyer (Nebraska)	Race Vulnerability Study and Hybrid Race Detection
9	X	X			X	Breaux, Niu (UTSA)	Usable Formal Methods for the Design and Composition of Security and Privacy Policies
10			X		X	Cranor, Acquisti, Christin, Telang, Egelman (Berkeley), Beach (U Pitt)	Understanding user behavior when security is a secondary task

The Carley (#11) and Harper (#5) projects are concluded.

2: Highly configurable systems

- **Leader: Christian Kästner, (Jürgen Pfeffer)**
- HPs: Composability. Metrics.
- Scope
  - Scalability of assurances for ***highly configurable*** systems
    - Exponential configuration spaces
    - Massive reuse of third-party libraries that evolve independently
  - Compositional analysis of configuration options enables **scaled** analysis
    - How are options implemented? How do they interact?
  - Support for modular and timely **recertification** of judgments
    - Support change and variation
- Recent example results
  - Safe updates for server-side JS (node.js)
    - Static analysis to assure the absence of certain malicious package updates in ***npm*** packages
    - Dynamic sandboxing of JS/node.js packages
  - SME study of software certification (CC, DO-178c)
    - (*Finalist, ICSE section ACM student research competition*)

7: Multi-model runtime analysis

- **Leaders: David Garlan, Bradley Schmerl**
- HPs: Resiliency. Composability. Metrics.
- Scope
  - **Resiliency architecture**
    - Attack scenario based on Target breach and related APT analysis
    - Testbed support to explore resiliency in this kind of setting
    - Defense tactics in the presence of threats
  - **Anomaly detection** algorithms on traces
    - Precision/recall analyses
      - Effectiveness, signal to noise, architecture size effect, abstraction function effect
  - **Flows of information** in socio-technical networks and in social networks
- Example results
  - Architecture evaluation techniques
  - Architecture generation (large scale) and resiliency scenario evaluation

1: Science of secure frameworks

- **Leaders: Jonathan Aldrich, David Garlan, Sam Malek (UCI), Marwan Abi-Antoun (Wayne State)**
- HPs: Composability. Resiliency. Metrics.
- Scope
  - Security assessment of software in a framework-based ecosystem (such as Android)
  - Uncertainty-aware decision making for resilient responses
- Example results
  - *DelDroid*: Analysis to automatically extract the least privileges required by each component in a program (ICSA 2017)
  - Comprehensive taxonomy of analytic techniques for Android software (IEEE TSE June 2017)
  - *Savasana*: Analysis to identify inter-/intra-component dependencies to ensure safe adaptation in a complex ecosystem (ACM TOSEM May 2017)
  - Architecture extraction for Android systems using semi-automated analysis

3: Security in distributed systems with uncertainty

- **Leaders: Andre Platzer, Dexter Kozen (Cornell)**
- HPs: Composability. Resiliency.
- Scope
  - Apply optimization techniques to security planning, compromising optimality for rapid solution
    - Application to anomaly detection, policy synthesis
      - Policy synthesis: How to adapt/escalate access control in response to anomalous system behavior
    - Builds on earlier work on #E-SAT solving
    - Based on Markov decision processes
  - Application (ongoing) is the 4-dimensional plane collision avoidance problem (FAA)
- Example results
  - Diverse technical results enabling large anomaly detection and security policy synthesis problems (thesis near completion)

4: Secure composition of systems and policies

- **Leaders: Anupam Datta, Limin Jia, Amit Vasudevan, Sagar Chaki (SEI), Petros Maniatis (Google)**
- HPs: Composability. Resiliency.
- Scope
  - Secure-object abstractions verifiable in low-level systems software
    - C99 and assembly using **UberSpark** models and analyses.
    - Exploit CompCert stack to create executable binaries
    - Enables "interface confinement" for analysis of adversary code
  - Apply to assuring security invariants in a performant hypervisor
- Example results
  - Rigorous integration of UberSpark, CASM (verifiable assembly code), CompCert to achieve verified properties in binaries
  - Architecture concepts to support application of UberSpark and its abstractions to heterogeneous systems (IoT, mobile, etc.)
  - http://uberspark.org

10: Security Behavior Observatory (SBO), 1 of 2

- **Leaders: A Acquisti, LF Cranor, N Christin, R Telang, S Egelman (Berkeley)**
- HPs: Humans. Metrics.
- Scope
  - Observe behavior of end users “in the wild” rather than in lab settings
    - Focus on security- and privacy-related activity
- Data collection:
    - >2 years of security and privacy behavior data from SBO human participants (~500 total, ~200 currently active)
    - Survey data from ~500 SBO participants
    - Several months of password behavior data for more than 200 enrolled human participants

10: Security Behavior Observatory (SBO), 2 of 2

- Example results
  - Users who claim to be more engaged with security practices ***do not*** necessarily have more secure outcomes

  - Assessments
    - Susceptibility of users to phishing attacks
      - Based on signal detection theory and risk homeostasis theory
    - Assessment of privacy-related behavior in browsing and shopping
      - Use of privacy plugins, incognito mode, etc.
    - Assessment of password-creation behavior, including degree of reuse

  - Seemingly effective compare-and-select crypto-key fingerprint representations (visual comparisons to thwart MITM) are generally ***not*** effective (CHI 2017)

  - There are patterns of password reuse, and it is endemic (*see tech talk*)

10: Race vulnerabilities

- **Leaders: Jonathan Aldrich, Josh Sunshine, Witawas Srisa-an (U Nebraska Lincoln)**
- HPs: Composition. Humans.
- Scope
  - Analysis of concurrent systems to detect race-related security vulnerabilities
  - Techniques for preventing race-related vulnerabilities through secure-by-construction development techniques and tools
- Example results
  - *Glacier*: a type system for enforcing immutability in Java. We report the first user studies demonstrating that a type system helps developers avoid security issues and implement immutability correctly, with applications to race vulnerability mitigation (ICSE 2017a)
  - *Jitana*: an efficient and scalable approach to analyzing whether inter-app communication in Android apps follows security constraints (ICSE 2017b)
  - *SimExplorer*: a testing framework that better controls nondeterminstic applications in order to more effectively find concurrency faults (J. Software: Testing, Verification, and Reliability)

8. Framework for secure mobile applications

- HPs: Composition. Humans.
  - Composition: Composable techniques for secure-by-construction software
  - Humans: Influencing developer behavior in constructing secure code

- Scope
  - Programming languages, type systems, and software frameworks that enable construction of mobile applications with **known security properties**

- Example results
  - A new formal model of authority in object capability systems, and a module system that facilitates capability-based reasoning about resource use in software systems (ECOOP 2017)
  - Integration of type safety into structure editors, enhancing editor services that can facilitate built-in security properties (POPL 2017, SNAPL 2017)

8. Modules as Capabilities for Resource Control

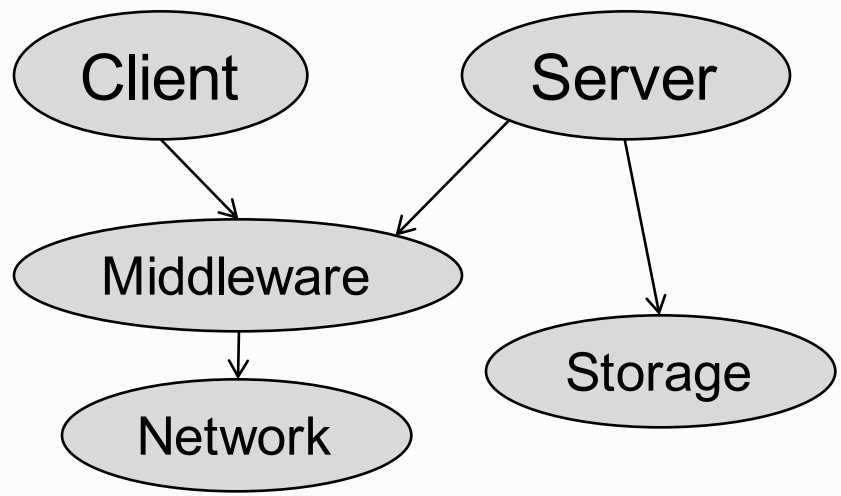
How can an architect maintain effective control over system architecture?

- In the example, what if the Client opens *other*, unsecured, connections?

Conceptual Architecture [SG94]

Solution: resources as **capabilities**

- Capability: an unforgeable token controlling access to a resource [DV66]
- No ambient capabilities
  - By default, Client and Server have no network capability
- Capability delegation
  - Explicitly pass capabilities to modules, such as Middleware, that need them



Capability / Module Structure

9. Formal methods for composing policies

- **Leaders: Travis Breaux, Jianwei Niu (UTSA)**
- HPs: Metrics. Humans.
  - Metrics: Empirical privacy risk score; iterative, measured security improvement framework
  - Humans: Influencing developer behavior in constructing secure code
- Scope
  - Facilitate assessment of consistency of privacy and security policies with actual app behaviors
- Example results
  - Privacy risk predicted by information type, user demographics (PLSC'17)
  - Privacy policy information type ontology (RE'17)
  - Framework to estimate security requirements improvement (RE'17)
  - Mapping of policy terminology to API functions (ICSE'16)
  - Case study of 501 top Android apps – discovered 63 policy violations
  - http://polidroid.org/
    - Tools for developers

The CMU Science of Security (SoS) Lablet

- The team
- Mission
- Significance of SoS
- Hard Problems
- Synergetic benefits
- CMU project portfolio