SecureCore for Trustworthy Commodity Computing and Communications: Year 2 Accomplishments

Ruby Lee, Princeton University (PI)
Cynthia Irvine, Naval Postgraduate School
Terry Benzel, USC/ISI
Mung Chiang, Princeton University

26 March 2007

SecureCore Vision

- Architectural foundation for trustworthy commodity products for mobile computing and communications
  - commercial and military use
- Approach: Clean-Slate, Integrated, Essential, Minimalist architecture
- Integrated security architecture spanning hardware, software and networking subsystems:
  - Secure Processor hardware (SP architecture)
  - Least Privilege Separation-Kernel (LPSK) and SecureCore Security Services (SCSS) software architecture
  - Ad-hoc network protocols and secure node architecture
  - Architectural mitigation of Covert channels & Side channels
- Goal: Security without compromising performance, cost and usability

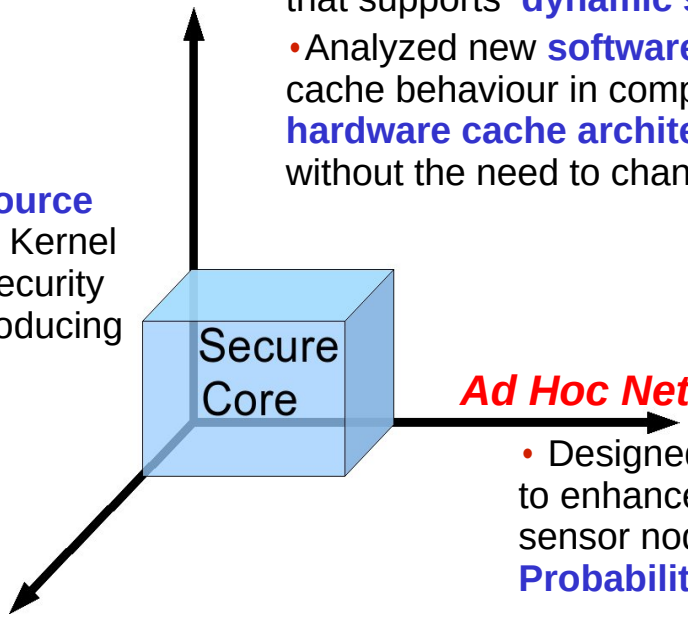
SecureCore: 3 thrusts and integration in Year 2

Security-aware Processor (SP) architecture

- Designed **Authority-mode SP processor architecture** that supports **dynamic security policies**
- Analyzed new **software side channels** due to hardware cache behaviour in computers, and designed **two hardware cache architectures** that block these channels without the need to change the software applications.

HW/SW Integration

- Support **SP hardware resource virtualization** in Separation Kernel architecture for multi-level security (MLS) domains, without introducing covert channels.



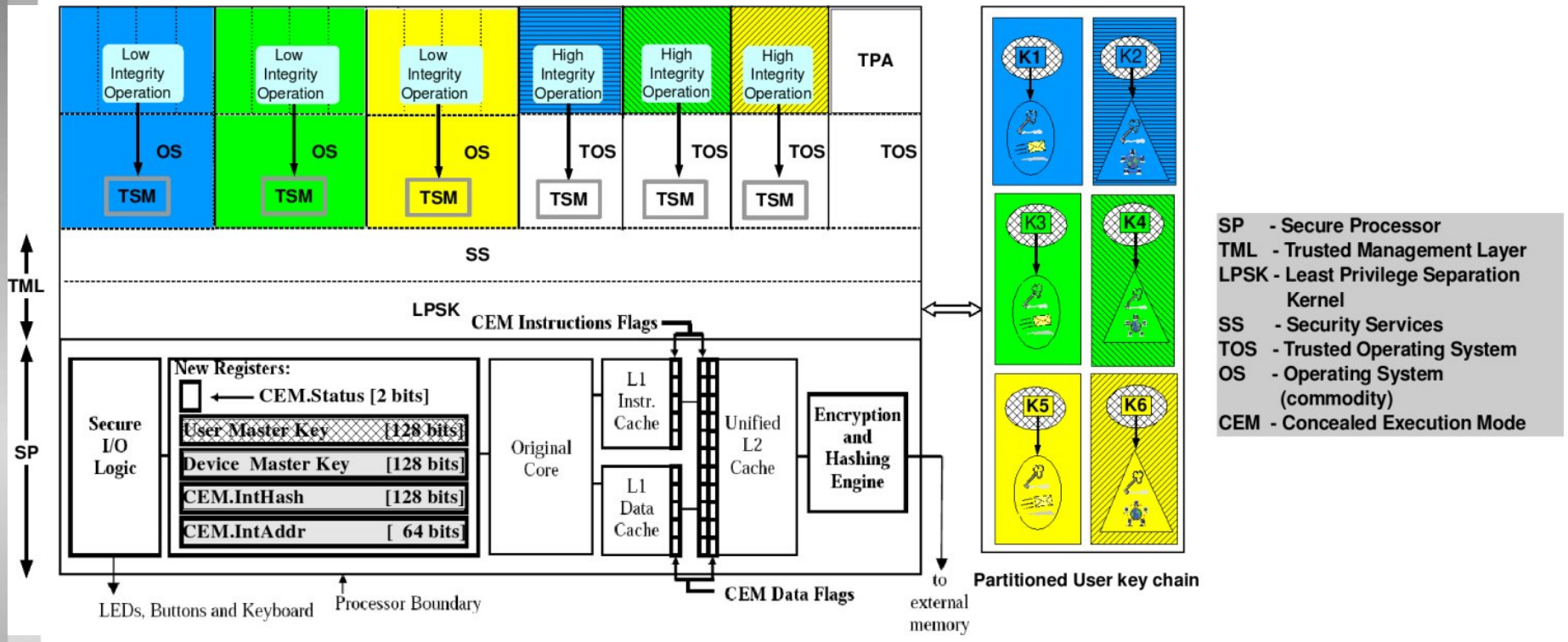
Ad Hoc Network Security Protocols

- Designed **reduced-mode SP architecture** to enhance **secure key management** for sensor nodes, r**educing Successful Attack Probability** by an order of magnitude

Software Security Architecture

- Defined principal subsystem interactions for **SecureCore Architecture Reference Design**
- Designed **Trusted Path Application** for emergency response and **dual use capabilities**
- Defined **new approach for confinement of emergency data**
- Developed **new metrics and model for dynamic security**

HW/SW Integration: Support SP Virtualization for Defense-in-Depth of Key Management



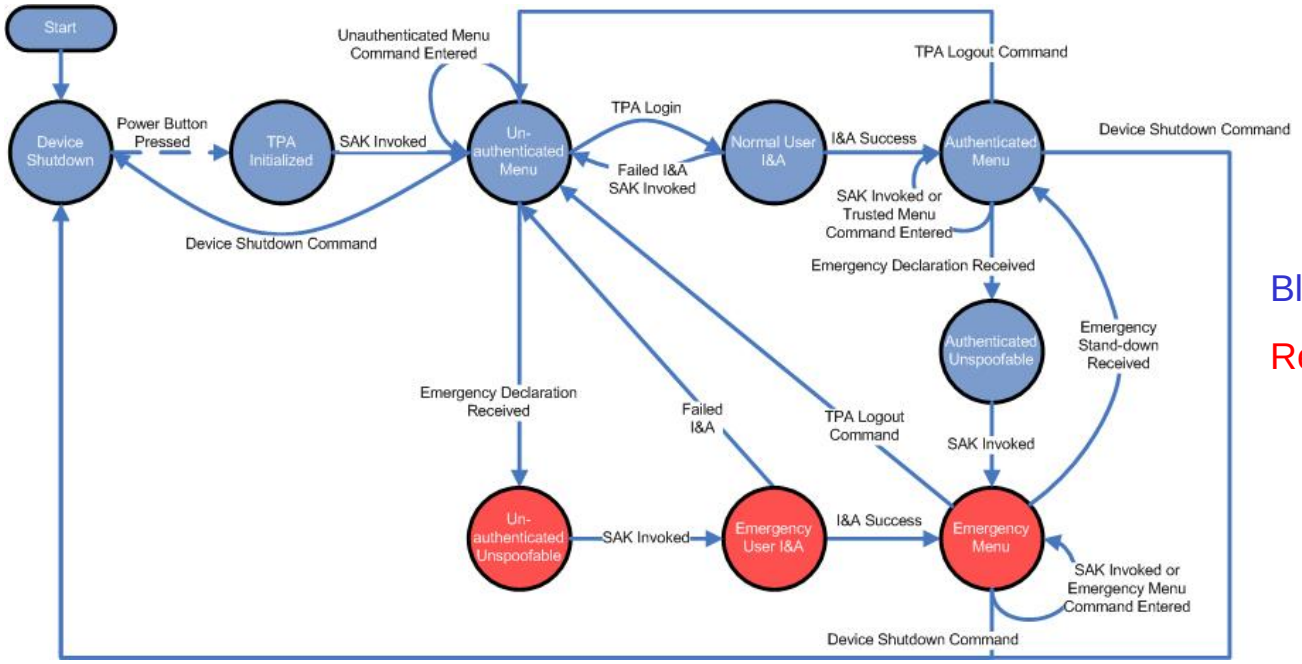
- **LPSK provides separate partitions for different MLS security & integrity levels**
  - Enforces MAC policies on MLS-labelled keys
  - LPSK virtualizes SP resources between partitions, avoiding covert channels, requiring new two new SP instructions
- **SP provides hardware protection of critical master keys that enable**
  - Secure storage and handling of user key chains across devices (User Master Key)
  - Concealed Execution Mode (CEM) when critical software uses these keys

SecureCore Architecture Reference Design

Layers	Functions	Security Policies
TPA **Trusted Path Application**	Trusted Path interface	None
TOS **Trusted Operating System**	Application Management I&A data management Emergency Management	Application Object reuse Identification & Authentication Integrity of Emergency State
SS **Security Services**	MLS Label Management High-level Scheduling LPSK Resource Virtualization User I/O Focus Management Trusted Channel Management	MLS Support
LPSK **Least Privilege Separation Kernel**	Device & Memory Mgt. Partition Scheduling Communication & Sync primitives	Mandatory Information-flow Enforcement
SP **Secure Processor**	Protection of master keys Concealed execution of code/data Cryptographic access control	Master-Key Confidentiality & Integrity Code Confidentiality & Integrity

- **Key challenge**s
  - Allocation of *functions* and *policies* to layers
  - Translation/extension of minimal LPSK to support applications
- **Key Innovations**:
  - Harmonious allocation of *enforcement* (SP and LPSK), *virtualization* (SS), and *management* (SS and TOS) functions
  - Factoring of MLS label support and MAC enforcement functions
  - Exploiting separation kernel strengths for partition access control

Trusted Path Application for Emergency Response

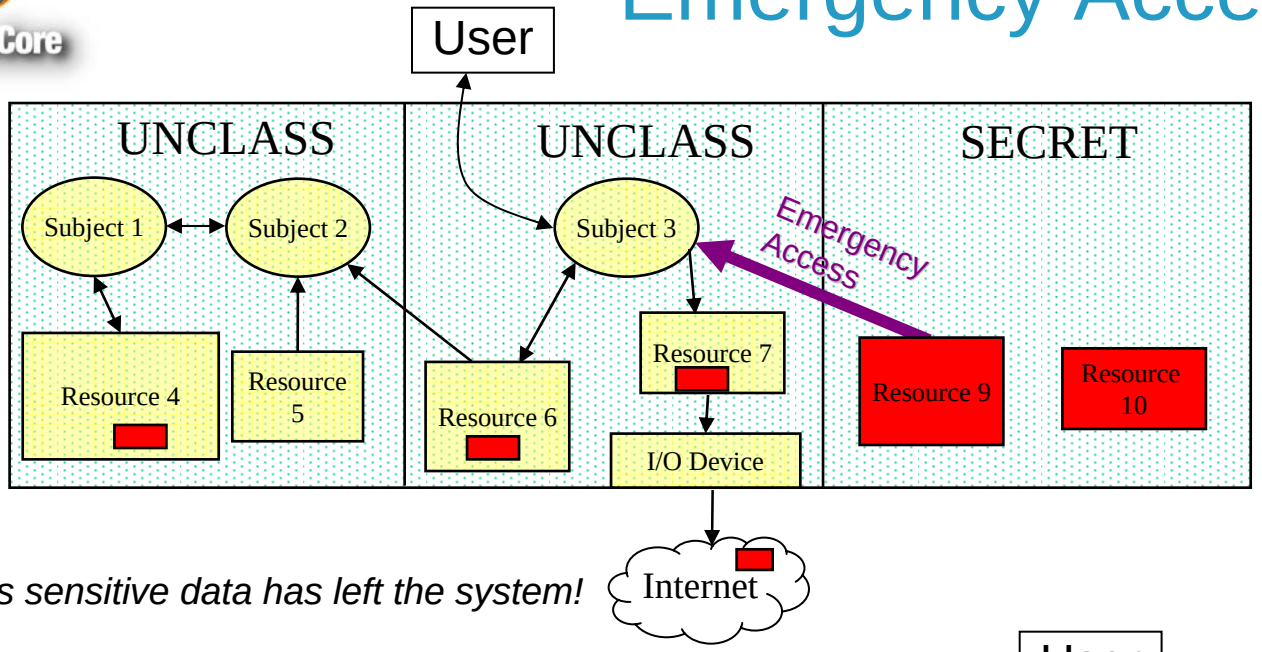


Blue = regular

Red = emergency

- **Trusted Path Application provides use-case driver for *dual-use* platform design**
- **Key Result**:
  - Top-level design for trustworthy management of both
    - Day-to-day processing functions
    - *Transient* access to emergency data
- **Key Progress**:
  - Integrating SP Authority Mode to provide in-depth security for emergency communications and platform attestation

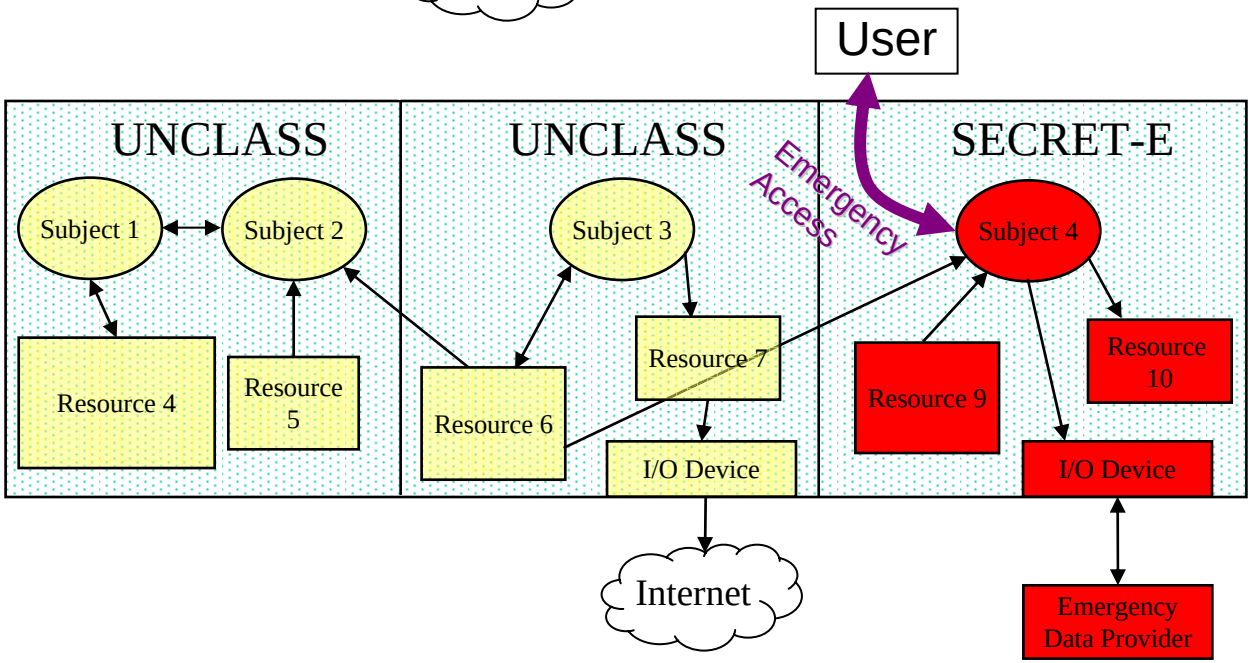
Emergency Access Design



Problem: Propagation of data during and after the emergency

Shows sensitive data has left the system!

Solution: Emergency partition confines data



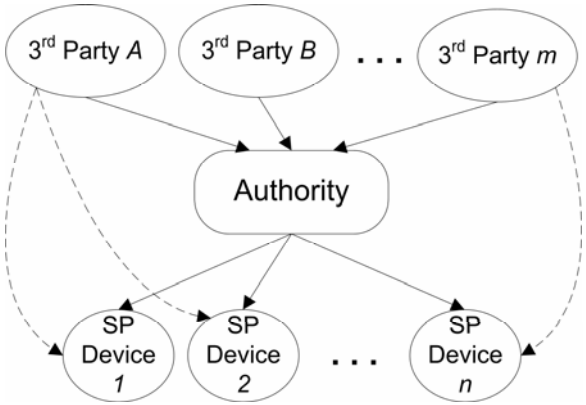
New Metrics and Model for Dynamic Security

	SECURITY ARCHITECTURES		
Security Factors	**MILS**	**Evaluated Policy**	**Least Privilege**
Intransitive Information Flow	Trusted Subjects	Trusted Subject + Categories	Trusted Subject + Categories + Least Privilege
Legend: Colors indicate relative security: yellow is lowest, blue is highest			

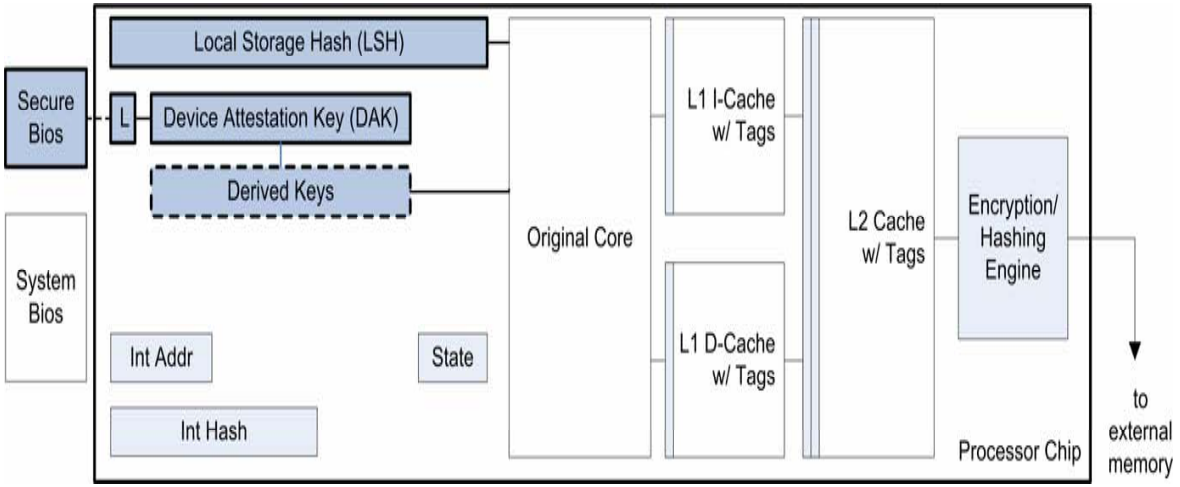
- IT security metrics needed for risk and cost/benefit analyses by:
  - Acquisition managers, system integrators, accreditors, etc.
  - Defined 10 *factors* determining system security, e.g
    - Control transitivity of information flow - restrict unlimited propagation
    - Locus of policy enforcement - which module(s) enforce the policy
    - Granularity with which the *principle of least privilege* can be applied
    - Support for dynamic policies - required for GIG, etc.
  - Analyzed several MLS-capable architectures
    - *MILS* - significant in several current DoD programs
    - *Evaluated Policy* - based on security kernel
    - *Least Privilege* - SecureCore security architecture
- Formal methods increase assurance of policy enforcement
  - Key Progress on new formal model for *dynamic* MLS security policy

Authority-mode SP processor architecture e.g., for emergency response

Trust Model



Hardware Architecture



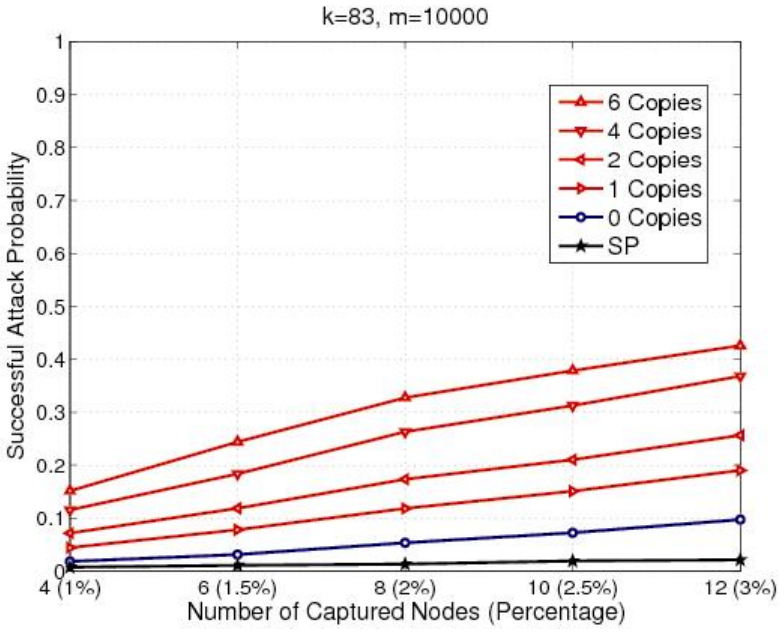
SP hardware architecture enables:

- remote trust in SP-enabled devices
- **dynamic security policies for emergencies ("transient trust")**
- hardware-enforced binding of policies to cryptographic keys
- transient access to third-party protected data, and
- reliable revocation of keys after emergency.

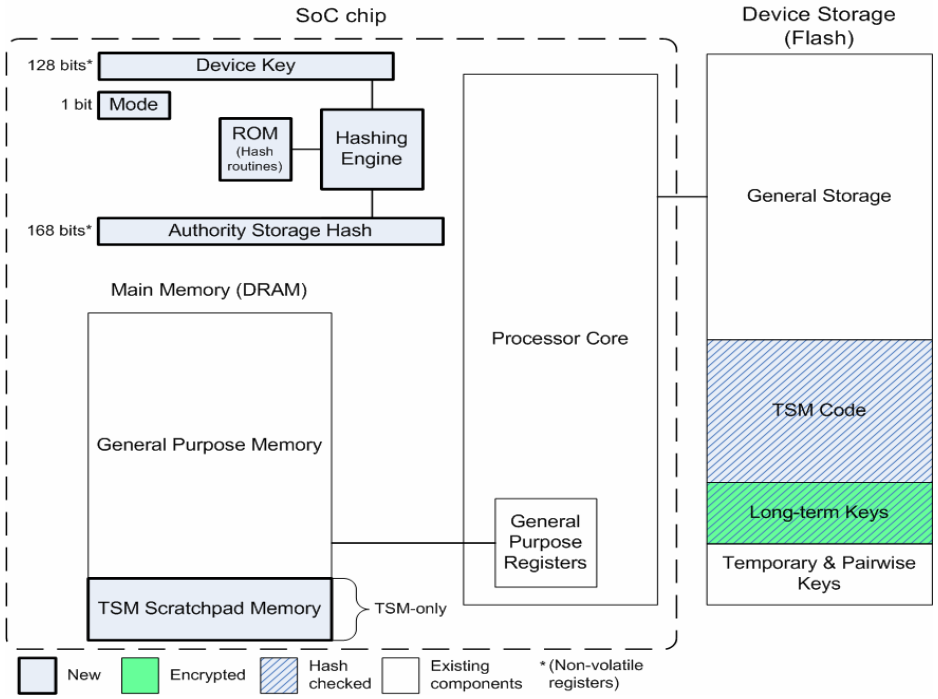
SP Instruction	Description
Begin CEM End CEM	Enter/Exit protected SP mode for TSM
Secure Load Secure Store	Access protected memory (by TSM-only)
Device-Key Derive	Create device-specific attestation keys
Authority Hash Read	TSM-only access to root hash in LSH reg.
Device Key Set Auth Hash Set	Initialize DAK register, Initialize LSH register

SP architecture provides Secure Key Management for mobile ad-hoc networks

Probabilistic Key Mgmt
(smaller SAP is better)



Reduced-mode SP architecture



- Nodes use their long-term keys to establish pairwise keys with other nodes for encrypting link traffic. If 2 nodes do not share a common long-term key, they use a relay node.
- When nodes are captured, their long-term keys can be pooled into a super-node and # copies can be made of this fabricated node.
- SP architecture protects these long-term keys, preventing fabrication of new nodes from captured nodes, and hence reducing the Successful Attack Probability on network links to negligible levels.

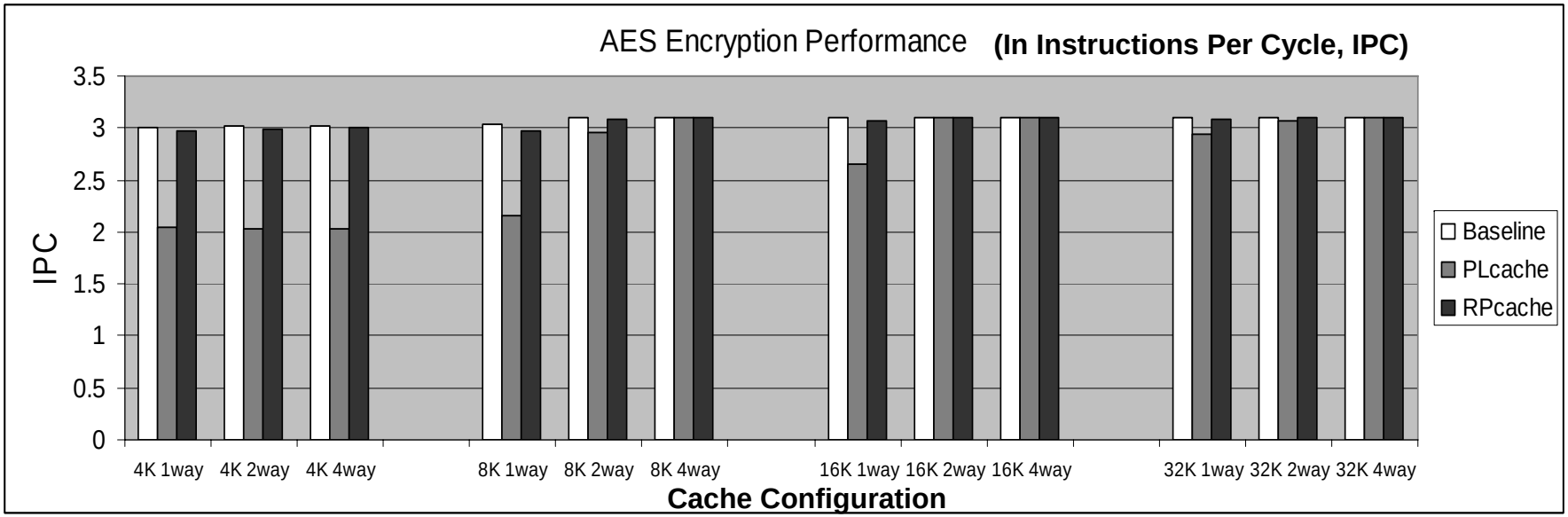
Cache Side Channels and 2 HW Solutions

Security & Performance	Partitioned Cache	Our PLcache	Our RPcache
Prevents external Interference?	Yes	Yes	Yes
Prevents Internal Interference?	No	Yes	Yes
Relative performance	Low	Medium	High

Interference in cache usage creates side channels that can leak cipher keys.

2 new solutions:

- PL cache: lock sensitive cache lines
- RPcache: randomize cache mappings



Year 3 Plans

- Support Emergency Response and High Threat Incidents - *transient trust*
  - Integrate Trusted Path Application with Authority-mode SP processor hardware support, and compare alternative approaches.

Secure System reference design

- Complete *dynamic security* design for transient trust
- Complete the design for Security Services layer
- Complete the *System* Formal Security Policy Model (kernel + services layers)

Develop **new** metrics from our (yr. 2) measurement factors

Reference design for Secure OS and Secure Processor

Design Secure Adhoc Network Protocols with:

- joint end-to-end key management and routing
- joint trust and resource management via back pressure
- Availability provisioning

- Design new Secure Processor (SP) hardware architecture for:
  - Enabling a secure initial application state, *dynamically*
  - Preventing memory-replay attacks with *deployable* memory integrity mechanisms
  - Reducing *capacity* of processor and cache induced covert channels

Backup slides

Backup: Interesting questions addressed in the slides

- Does the project support advanced features such as:
  - Multiple emergency mode partitions?
    - Yes, one for each group of 3rd parties who share emergency data
  - Multiple sensitivity levels in partitions?
    - Yes, but only in the trusted partition
  - Movement of info between partitions?
    - Yes, for commodity and trusted partitions, as limited by TML policy
    - During emergency, emergency partition can read from lower-class partitions
- What is dynamic during an emergency?
  - Emergency partition available to user only during emergency
  - User may be provided *more sensitive* info only during emergency
  - **Key Result**: TML constrains emergency behavior

Backup: Measurement and Assurance

- Other Measurement Factors
  - *Controlled interference*
    - New fine-grain control of trusted subjects
  - *Label-space scalability*
    - Support for lots of labels (e.g., Intelligence Community)
  - *Evaluation effort*
    - Are complex evaluations required?
  - *Cohesion of evaluation*
    - Are multiple evaluations required?
  - *Assurance via kernel control of trusted subjects*
    - More assurance than application control
  - *Locus of Dynamic Resource Management*
    - which module(s) provide dynamic resource management
    - Fewer is better, lower in the software stack is better
      - because it limits cost of assurance

Year 2 Accomplishments: A Summary

1. Key functions/services provided by SecureCore Kernel (LPSK & SS):
   - High-level specification for 8 Kernel interface categories
   - 40 Kernel interfaces
2. Defined 10 measurements to compare Secure MLS architectures – SecureCore's LPSK/SS scored better than MILS and traditional Security Kernels.
3. Integrated user-mode SP hardware architecture in Reference Secure System Architecture - new SP instructions allow covert channel-free use by MLS domains.
4. Defined new hardware (Authority-mode SP processor) and Secure Kernel and trusted services to support dynamic security policies for PDAs used by emergency first-responders.
5. Significantly improved security of mobile ad-hoc network key management mechanisms with sensor-mode SP architecture.
6. Identified very fast processor-based covert channels and cache-based side channels and designed two new hardware solutions.
7. Refined SP hardware architecture design via different usage scenarios: User-mode (year 1); Crisis response Authority-mode; Reduced-Mode for sensors, and MLS needs (year 2).

Broad Impacts

- Foundation for secure computing, secure information management and secure ad-hoc communications
  - Emergency response / transient trust
  - Contextual-adaptive security
  - Dynamic coalitions
  - Multi-use technology
    - Decreased costs
    - Broadens availability of security
  - Economical
    - Feasible transfer to commercial/military
  - Research and education vehicle

Year 2 Publications – page 1 of 2

1. Jeffrey Dwoskin and Ruby B. Lee, “Processor Architecture for Remote, Transient, Policy-controlled Secrets,” Princeton University Department of Electrical Engineering Technical Report CE-L2006-007, November 2006.
2. Jeffrey Dwoskin and Ruby Lee, “Enabling Transient Access to Protected Information for Crisis Response”, Princeton University Department of Electrical Engineering Technical Report CE-L2006-001, May 2006.
3. Ruby B. Lee, Jeffrey Dwoskin, and David Champagne, “Fundamental Architectural Features in SP processors for Protecting Sensitive Information,” submitted to IEEE Micro, December 2006.
4. Peter Kwan and Ruby B. Lee, “Minimalist Security Architecture in SP-processors”, Book Chapter, Hardware-Based Security Anthology, publisher review, 2006.
5. Zhenghong Wang and Ruby B. Lee, “Covert and Side Channels due to Processor Architecture”, Proceedings of the 22nd Annual Computer Security Applications Conference (ACSAC'06), pp.473-482, December 2006.
6. Michael Neve, Jean-Pierre Seifert, and Zhenghong Wang, “A refined look at Bernstein's AES side-channel analysis”, Fast abstract in the Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security, pp. 369, March 2006.
7. Zhenghong Wang and Ruby B. Lee, “New Cache Designs for Thwarting Software Cache-based Side Channel Attacks”, International Symposium on Computer Architecture, ISCA'07, June 2007, to appear.
8. Zhenghong Wang, Jing Deng, and Ruby B. Lee, “Mutual Anonymous Communications: A New Covert Channel Based on Splitting Tree MAC”, 26th Annual IEEE Conference on Computer Communications (Infocom '07), Minisymposium, May 2007, to appear.
9. Zhenghong Wang and Ruby Lee, “Cache-based side channel attacks: analysis and countermeasures,” Princeton University Department of Electrical Engineering Technical Report, May 2006.
10. Reouven Elbaz, David Champagne and Ruby B. Lee, “TEC-Tree: A Low Cost and Parallelizable Tree for Efficient Defense against Memory Replay Attacks”, Technical Report CE-L2007-002, Department of Electrical Engineering, Princeton University, March 12th 2007. Submitted to conference.
11. David Champagne and Ruby B. Lee, “Memory Integrity for Secure Computing Platforms”, Draft Technical Report, Department of Electrical Engineering, Princeton University, June 2006 (under revision)
12. David Champagne and Ruby B. Lee, “Scope of DDoS Countermeasures: Taxonomy of Proposed Solutions and Design Goals for Real-World Deployment”, 8th International Symposium on Systems and Information Security, November 2006.
13. David Champagne and Ruby B. Lee, “Remote Takeovers: Software Vulnerabilities and Architectural Countermeasures”, Book Chapter, Hardware-Based Security Anthology, publisher review, 2006.

Year 2 Publications – page 2 of 2

14. Ganesha Bhaskara, Timothy E. Levin, Thuy D. Nguyen, Cynthia E. Irvine, Terry V. Benzel, Jeffrey Dwoskin, Ruby Lee, *Virtualization and Integration of SP Services in SecureCore*, University of California, Information Sciences Institute Technical Report ISI-TR-623, September 2006
15. Thuy D. Nguyen, Timothy E. Levin, Cynthia E. Irvine,Terry V. Benzel, and Ganesha Bhaskara, *Preliminary Security Requirements for SecureCore Hardware*, Naval Postgraduate School Technical Report NPS-CS-06-014, and University of California, Information Sciences Institute Technical Report ISI-TR-621, September 2006
16. Timothy E. Levin, Cynthia E. Irvine, Thuy D. Nguyen, Terry V.Benzel, Ganesha Bhaskara, *Initial SecureCore Security Architecture,* University of California, Information Sciences Institute Technical Report NPS-CS-07-003, March 2007
17. Timothy E. Levin, Cynthia E. Irvine and Thuy D. Nguyen, "Least Privilege in Separation Kernels," *Proceedings International Conference on Security and Cryptography*, Setubal, Portugal, August 2006, pp. 355-362
18. Timothy E. Levin, Cynthia Irvine and Thuy Nguyen, *An Analysis of Three Kernel-based Multilevel Security Architectures*, Naval Postgraduate School Technical Report NPS-CS-06-001, August 2006
19. Ganesha Bhaskara, Timothy E. Levin, Thuy D. Nguyen, Terry V. Benzel, Cynthia E. Irvine, Paul C. Clark, Integration of User Specific Hardware for SecureCore Cryptographic Services, Naval Postgraduate School Technical Report NPS-CS-06-012, July 2006
20. Thuy D. Nguyen, Timothy E. Levin, Cynthia E. Irvine, "High Robustness Requirements in a Common Criteria Protection Profile," Proceedings of the 4th IEEE International Information Assurance Workshop, Royal Holloway, UK, April 2006, pp. 66-75
21. Timothy E. Levin, Cynthia E. Irvine and Evdoxia Spyropoulou, "Quality of Security Service: Adaptive Security," Handbook of Information Security, Vol.3, pp 1016-1025, ed. H. Bidgoli, John Wiley and Sons, 2006
22. D. Xu, J. Dwoskin, J. Huang, M. Chiang, and R. Lee, 'Re-examining probabilistic versus deterministic key management', IEEE International Symposium on Information Theory, June 2007, to appear.
23. Jeffrey Dwoskin, Dahai Xu, Jianwei Huang, Mung Chiang, and Ruby Lee, "Secure Key Management Architecture Against Sensor-node Fabrication Attacks." Submitted to conference, 2007.
24. M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures", Proceedings of the IEEE, Jan. 2007
25. D. Xu, Y. Li, M. Chiang, and A. R. Calderbank, 'Optimal provisioning of elastic service availability', Proc. IEEE INFOCOM, May 2007, to appear.
26. J. W. Lee, A. Tang, J. Huang, M. Chiang, and A. R. Calderbank, 'Reverse-engineering MAC: A game-theoretic model', IEEE Journal of Selected Areas in Communications, 2007, to appear.