Trustworthy Commodity Computation and Communication

Perform research into design of secure integrated core architectures for trustworthy operation of mobile computing devices.

Including:
- Security-aware SecureCore Hardware
- SecureCore Least Privilege Separation Kernel
- SecureCore Security Services
- and secure communications

For use in resource-constrained, ubiquitous computing platforms exemplified by secure embedded systems and mobile computing devices

### Comparison to state-of-the-art

**Current approach**
- ad hoc revocation mechanisms
- temporal policies lack low level support
- VMs provide no sharing
- trusted subjects all or nothing
- isolated design of layers
- security with coprocessor

**New Approach to**
- revocation
- temporal access control
- read down from VM
- modeling & assured control of trusted subjects
- codesign of HW/Kernel/Services
- unified processor

### Technical Summary

**Anticipated technical advances**
- Kernel-based fine grain control of trusted subjects
- A trusted subject may only access certain objects in its trust range – minimizes reliance on the correctness of application-domain security services
- Formal model and architectural solution define “controlled interference” for trusted subjects.
- Subjects can “read down” to blocks at lower levels, as allowed by kernel
- Also, kernel-controlled controlled write-up ("hilled" write)
- Traditional separation kernel architectures lack these abilities
- Exportation of hardware interrupts to the client OS
- Enables OS-specific interrupts handling regarding subjects' access violations to individual resources
- Traditional separation kernel architectures only provide block-level notification
- Kernel-based "intrusive information flow" enforcement
- Traditionally requires trusted subjects
- SecureCore supports, for example, a policy whereby each subject may only read down one level, because of data integrity or system assurance concerns.

**Innovation**
Utilization of hardware/kernel/SCSS co-design to construct SCSS interface such that SecureCore unique security features do not require modifications to the client OS.

**Recent Developments**
- Hardware and software architecture and authorization model to support temporal access controls
- Hardware and software mechanisms to support object reuse requirements
- Re-examination and synthesis of security principles relative to current technology trends and target platform

---

NSF Cyber Trust Annual Principal Investigator Meeting
Sept. 25 - 27th, 2005
Newport Beach, California