Trustworthy Commodity Computation and Communication

Enables dynamic, "transient trust" security policies for achieving the appropriate availability of highly sensitive information during emergencies in the face of determined adversaries.

Accomplishments

- Concept of operation
  - Multilevel-secure (MLS) multi-use handheld device
  - Different functional contexts correspond to different user roles:
    - Everyday and emergency
    - Normal or trusted
    - Support inter-context secure sharing of information
  - Trustworthy security architecture that can support dynamic security policies and services
  - Core building blocks
    - Security-aware processor extensions
    - Least privilege separation kernel
    - Trusted security services
    - Secure operating-system services
    - Trusted path application

SecureCore Software Architecture

<table>
<thead>
<tr>
<th>Layer</th>
<th>Functions and Policies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trusted Path Application</td>
<td>Trusted Path interface to security-critical services</td>
</tr>
<tr>
<td>SCOS</td>
<td>Application/Management, Identification and Authentication Operating System Services</td>
</tr>
<tr>
<td>SCSS</td>
<td>MLS Support and Interpretation, Resource Virtualization, Object Management, Focus Management, Trusted Channel Management, Inter-Partition Rounding</td>
</tr>
<tr>
<td>LP/SP</td>
<td>Partitioning of Resources, Resource Management, TAR Enforcement, Partitioning Scheduling, Cross-Partition and Inter-Process Communication</td>
</tr>
</tbody>
</table>

Concept of Operation

- SecureCore
  - Trusted Path Application
  - SCOS
  - SCSS
  - LP/SP

Functional Prototype Design

- Three partitions
  - Software-emulated SIP module
  - LPSK utilizes hardware security mechanisms
    - Segmentation
    - CIC gates
    - Hardware privilege levels
    - Task state management
    - SCSS and LPSK co-locate in same privilege level
    - Secure Attention Key (SAK)
      - Keyboard input
      - Focus switch via SAK
    - Simple crypto key management application

SP HW Architecture

- User-mode: enables controlled and secure access to user's secrets
- Authority mode: enables transient, policy-controlled access to third-party protected information, remotely
- Reduced mode: for use in low power applications

Contributing Members

(Alphabetically ordered)

Ganesh Bhaskara*, Paul Clark*, Timothy Levin*, Thuy Nguyen*, Mark Orwat*, David Shifflett*, Timothy Vidas*

Princeton University, Naval Postgraduate School, Information Science Institute/USC

NSF Grant No. CNS-0430487, CNS-0430566 and CNS-0430598