Two-Year Research Activity "System-Aware Security"

Jennifer Bayuk

Barry Horowitz

November, 2011

Perimeter Security Solutions have been the Mainstay for Cyber Security

Advantages	Disadvantages
Separates Out Application Development "Time to Market" Pressures	Application Developers Feel Less Accountable for Built-in Security
Reduces Design Complexity and Can be Added Responsively	Not Designed to Account for Application Specific Designs or Risks
Draws on Available Experience & Knowledge Regarding Solution Values and Limitations	Security Workforce Lacks Knowledge Regarding Business Risks, Resulting In Potential Misappropriations of Risk Reduction
Commercially Available with Economy of Scale Beyond that of Application Specific Solutions	Attackers Can Reuse Exploits Across Different Systems, and Impact Solution Security Through the Supply Chain
Support Exists for Training and Administrative Support	Limited Attention to Operational System Cyber Attack Exercises as Part of Training
Best Practices Exist for Application of Solutions	Oriented to Local Prevention and Detection – Not System-of-Systems Related

System-Aware Security

- We assume that the security community accepts the premise that the evolving threat of cyber attacks cannot be fully addressed with perimeter security alone
- A systems engineering approach, System-Aware Security, is being formulated:
  - Architectural formulation that includes application layer security services
  - Smart reusable security framework:
    - Application layer solutions
    - Local-Information gathering
    - Aware of system mission objectives
    - Aware of current system state
    - Share information among services
    - Capable of adapting security modes based on mission objectives and perceived threats
    - Converted into reusable design patterns for specific security services
    - Design patterns include metrics that relate increases in security to corresponding impacts on system performance

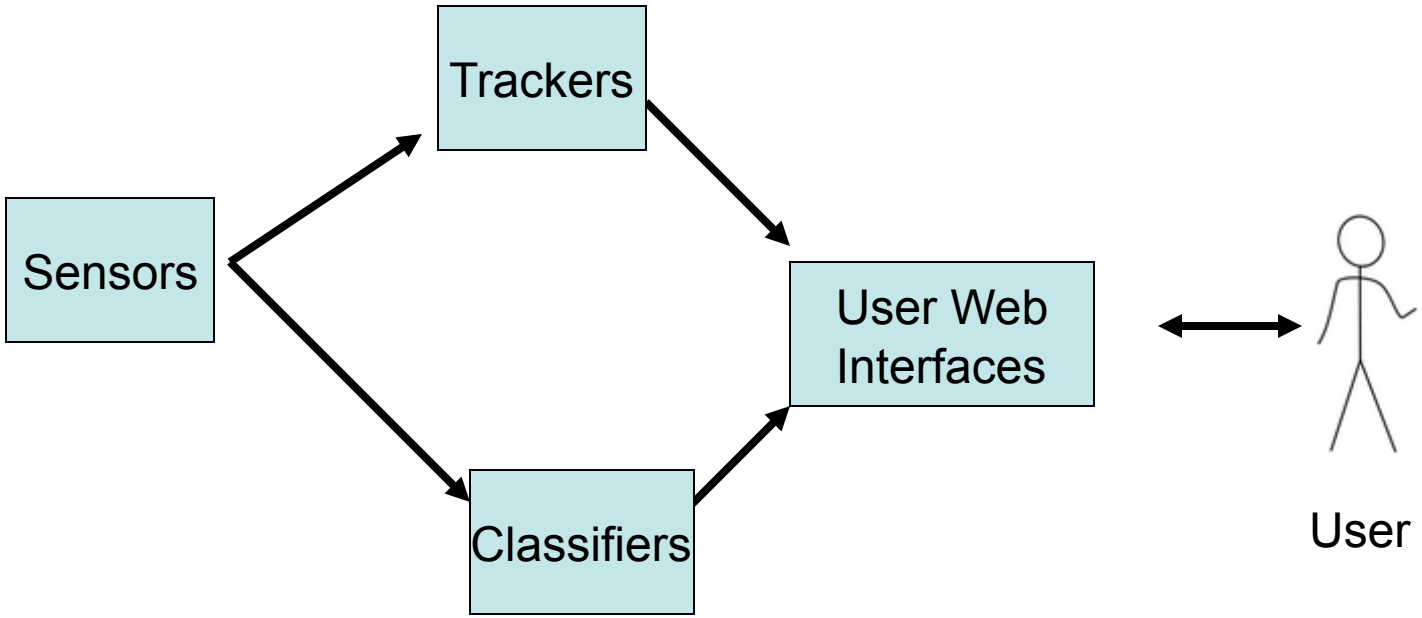
Example Application Layer Services

- Data Continuity Checking
- Virtual Configuration Hopping
- Physical Configuration Hopping
- Automatic Data Replay Discovery
- Honeypots to support real-time threat evaluation
- Adversary-Sensitive System Reconstitution
- Deception Services
- Resilience Services

Proposed Research(1)

- Develop an exemplar System-Aware Security Architecture for a “pipe-line surveillance system” including the integration of a number of application layer services:
  - Data continuity checking
  - Real-time virtual configuration hopping of selected application functions across multiple operating systems
  - Real-time physical configuration hopping of selected application functions
  - Honeypots to divert threats to these capabilities

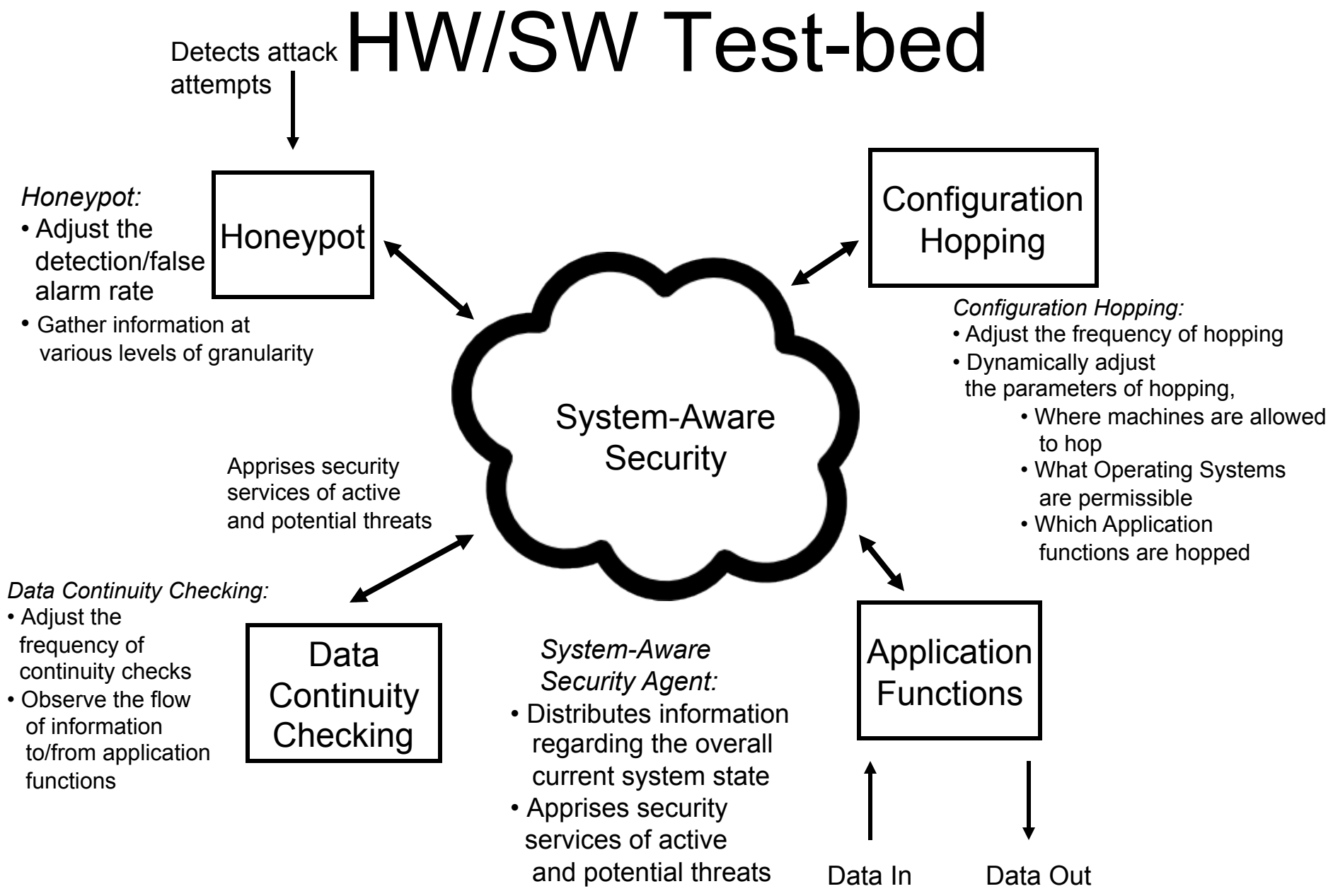
HW/SW Test-bed



Support Tools

Scenario Generation
Metrics and Measurements
Analysis

HW/SW Test-bed



Proposed Research (2)

- Test-bed will be used to develop a model for:
  - Elucidation of the tradeoffs between the level of security and system performance
  - Dynamically adjusting the level of protection based on local information from security services and the current system state
  - Exploring the relationships among security services in order to support the design of adaptive System-Aware security features based on security data fusion
- Test-bed results will be used as a basis for creating design patterns for smart reusable security services.

Design Patterns Characteristics

Name:	Descriptive name for the pattern.
Context:	Framework to which the pattern applies.
Problem:	Description of the problem.
Forces:	Tradeoffs, value contradictions, key dynamics of tension and balance, constraints.
Solution:	Description of the solution.
Graphic:	A depiction of response dynamics.
Agility:	Evidence of characteristics that qualify the pattern as agile and adaptable.
Examples:	Referenced cases where the pattern is employed.

Metrics as Part of Design Patterns

Verification: Configuration specification measures.

Functional tests for correct behavior.

Validation: Ability of system to perform mission under design basis threat.

Measure of reusability within security framework.

V and V is interpreted as:

Correctness – Do the security features work?

Effectiveness – Do they contribute to system security?

Products from Research Effort

- Description of System-Aware Security Architectural Concept
- Example Implementation
  - Emulated Surveillance System Application
  - Implemented Prototype System-Aware Security Services
  - Integration of Services to Dynamically Adapt Performance Based upon Risk and System Performance Implications
  - Experimental Results from Emulation-Based Sensitivity Analysis
- Design Patterns with Embedded Metrics Regarding Security/System Performance Tradeoffs