

# Structured Systems Economics for Security Management

**Adam Beautement**  
UCL

**David Pym**  
University of Aberdeen

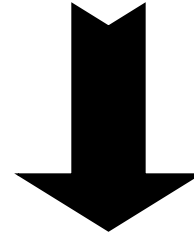
# Background

- Began with a discussion of CIA
- Interested in:
  - Policy decision making
  - Trade-offs
- Various issues arose that current literature didn't satisfactorily cover
- A new system was needed

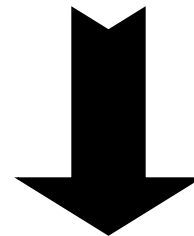
# The Problems with CIA

- Definitions too nebulous
- Essentially impossible to systematically analyze trade-offs with current conceptualizations
- Extensions to the model tend to involve category errors e.g. Parkerian Hexad adds:
  - Possession
  - Utility
  - Authenticity
- Good for discussing security objectives but falls down at the implementation stage

Policy Decisions



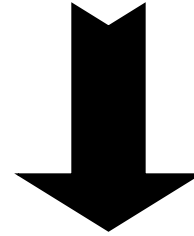
????



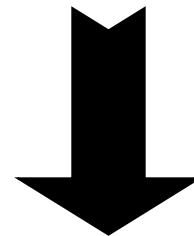
System  
Modelling

Systematic Security  
Analysis  
(e.g. Security Analytics  
developed at HP)

Policy Decisions



Structured Systems  
Economics



System  
Modelling

# Key Aims

- Consciously separate declarative and operational concepts.
- Represent security objectives and the methods used to realized them
- Structure information for ease of transition to modelling
- Minimize collection of additional information in order to achieve the above

# Concepts and Components

- Hierarchy of roles
- Framework layer
- Instantiation layer
- Security Objects (SO)
- Security Components (SC)
- Actors

# Hierarchy of Roles

- Ordered by their influence on the system
- Objectives and actions pertinent to each role characterised as:
  - **Dependencies:** externally mandated, required for the role to function
  - **Priorities:** externally mandated, enacted as resources allow
  - **Preferences:** provided internally by the Actor occupying the role

# Framework Layer

- Declarative
- Maps onto the hierarchy of roles
- Contains Security Objects
- Constructed iteratively and top down
  - This process ends when all dependencies and priorities are associated with a terminated security object
- Static

# Instantiation Layer

- Operational
- Preserves role hierarchy
- Contains Security Components and Actors
- Constructed iteratively and bottom up
  - This process ends when all required resources and processes can be provisioned without recourse to a higher level
- Dynamic

## Framework Layer

## Instantiation Layer

Rôle	Dependency	Priority	Dependency	Priority	Pref.
1					
2					
3					
...					
n					
n+1					

# Security Objects

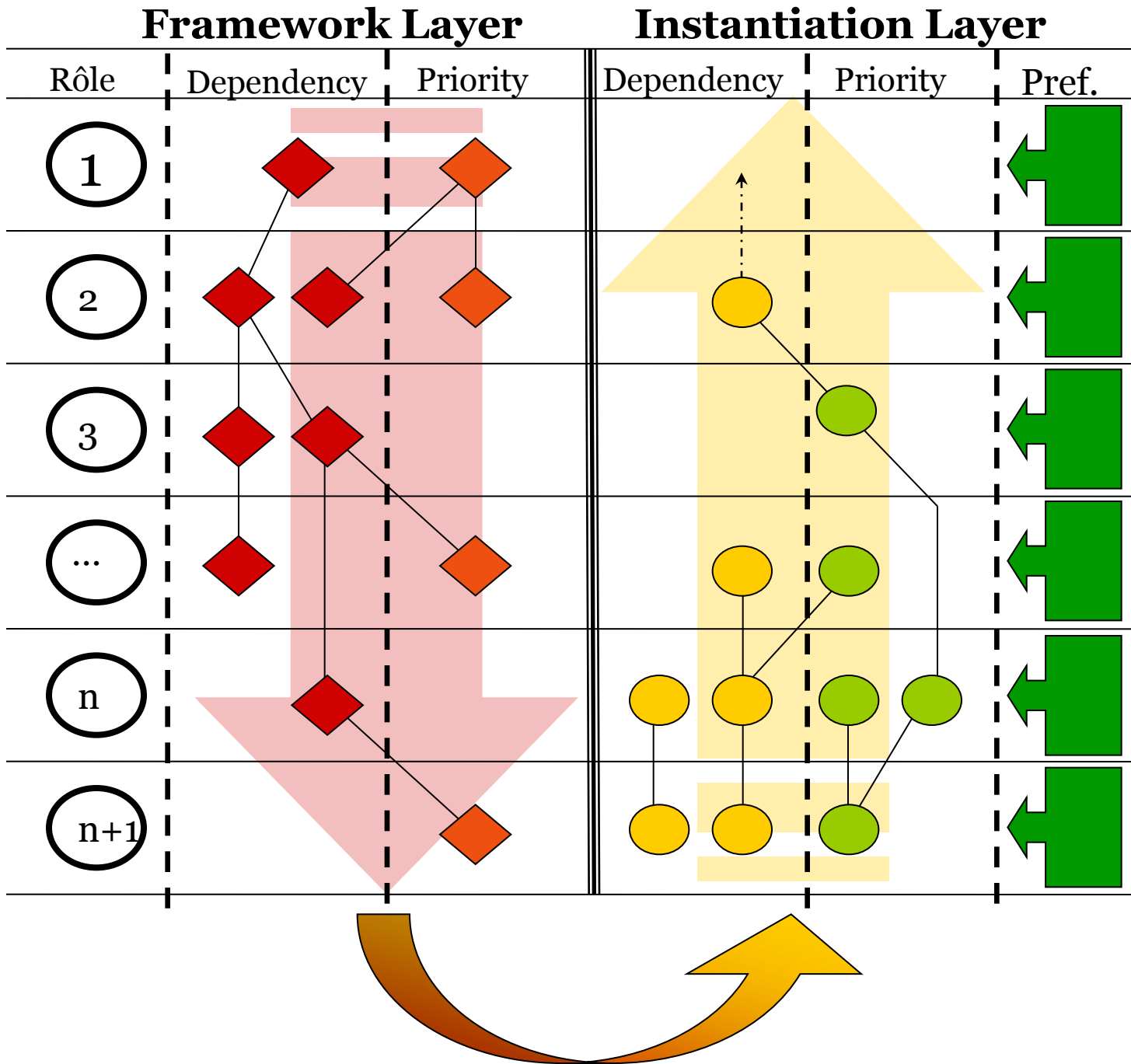
- Represent security objectives
- Characterized by and/or forests associated with dependencies and priorities
- Nodes contain a Boolean value associated with accompanying dependency/priority
- Truth conditions are inherited upwards
- End with a compliance step indicating that the current layer (and those below) are no longer involved in executing the SO

# Security Components

- Represent processes and resources
- Follow the SO they are instantiating
- Each SC implements a checking process that applies to Actors at the level below
- End when all necessary Boolean values (as required by the SO) can be returned without recourse to a higher level

# Actors

- Can transition between hierarchies (and are unique in this capability)
- Represented as a tag-cloud that each hierarchy interrogates
- Assigned to a rôle, based on attributes gleaned from tags (not all of which can be read by each hierarchy)
- Deliver preferences (and motivations) into the hierarchy



**Actors**

# Modelling and Logic

- Systems models are described in terms of:
  - Processes
  - Resources
  - Locations
  - Environment
- SCs in the Instantiation Layer also express these concepts, intentionally so
- This allows SCs to be exported cleanly into a systems modelling language (e.g., Gnosis)
- The logic underpinning SC termination conditions and properties is also captured in such a modelling language, providing a first step in the modelling process

# Application

- A familiar interface (e.g., OWL) can be used to represent the Instantiation Layer
- Existing system information and expert knowledge drawn in at this stage
- Information organised to facilitate system modelling
- Trade-offs between CIA-like objectives can now be explored systematically without recourse to CIA itself

# Future Work

- Fully develop the role of Actors
  - Negative actors
  - Zero-preference actors
- Explore interactions between multiple hierarchies
- Improve methodology to limit the needs for 'hand-crafted' stages