

## **STC 2014 - The 26th Annual IEEE Software Technology Conference**

The Security Architecture Discipline as Catalyst to  
Effect Secure Design:  
Employing Abuse Case Elaboration for Attack  
Vector Discovery and Countermeasure  
Determination

Murray Rosenthal, CISA, CRISC  
Risk Management & Information Security  
IT Strategic Planning & Architecture  
I&T Division  
City of Toronto

# Context

- ❑ Architecture informs the design of complex objects that, in turn, facilitates their manufacture and implementation.
- ❑ In the absence of architectural rigor, the design, implementation and maintenance of these objects in a steady-state become untenable, as does object reliability.
- ❑ Secure system design is an outcome of security architecture such that the implementable object is well-behaved, resistant to external attack, and free of internal anomalies that would otherwise jeopardize functional integrity.
- ❑ Intelligent, web-based agents represent an emerging trend in delegated trust such that they act, and behave, in a manner congruent with the instructions - and intuitions - of their human counterparts.
- ❑ This session will look at security architecture artefacts in the context of intelligent, web-based agents.

# Session Components

## Context

- Security Architecture Elicitation
- Security Architecture Artefact Manufacture

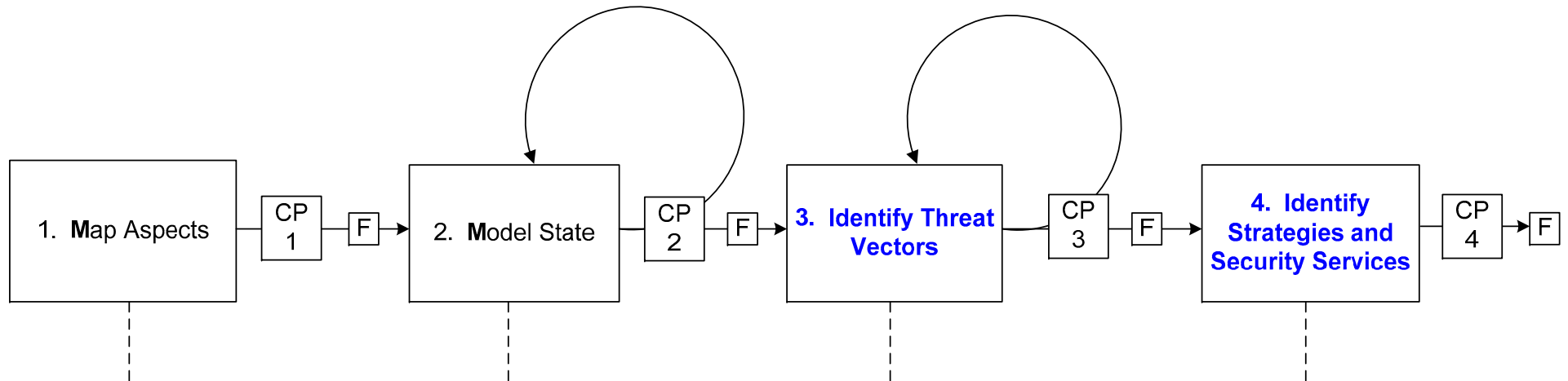
## Business Architecture

- Enterprise Context Model
- Service Integration and Accountability Model

## Logical Architecture (Security Viewpoint)

- Conceptual Architecture Pattern
  - Components View
  - Relationships View
- Class Diagrams and Derivatives
  - NonPersonEntity
  - NonPersonEntity Attack Vector Articulation
  - NonPersonEntity Abuse Case Notation
  - PersonEntity + NonPersonEntity + Service (Normative “Happy Path” Automation Target)
  - PersonEntity + NonPersonEntity + Service (Anomalistic Behaviour)
- Use and Abuse Case Scenarios

# Security State Modeling Elicitation Process



- Data-to-Security Classification Mapping
- Data Disposition-to-Zone(s)
- Role-to-Data-to-CRUD Attribution
- Role-to-Zone
- Location-to-Zone

- Aspect Placement
- Data In-Transit
- Data At-Rest
- Data In-Process

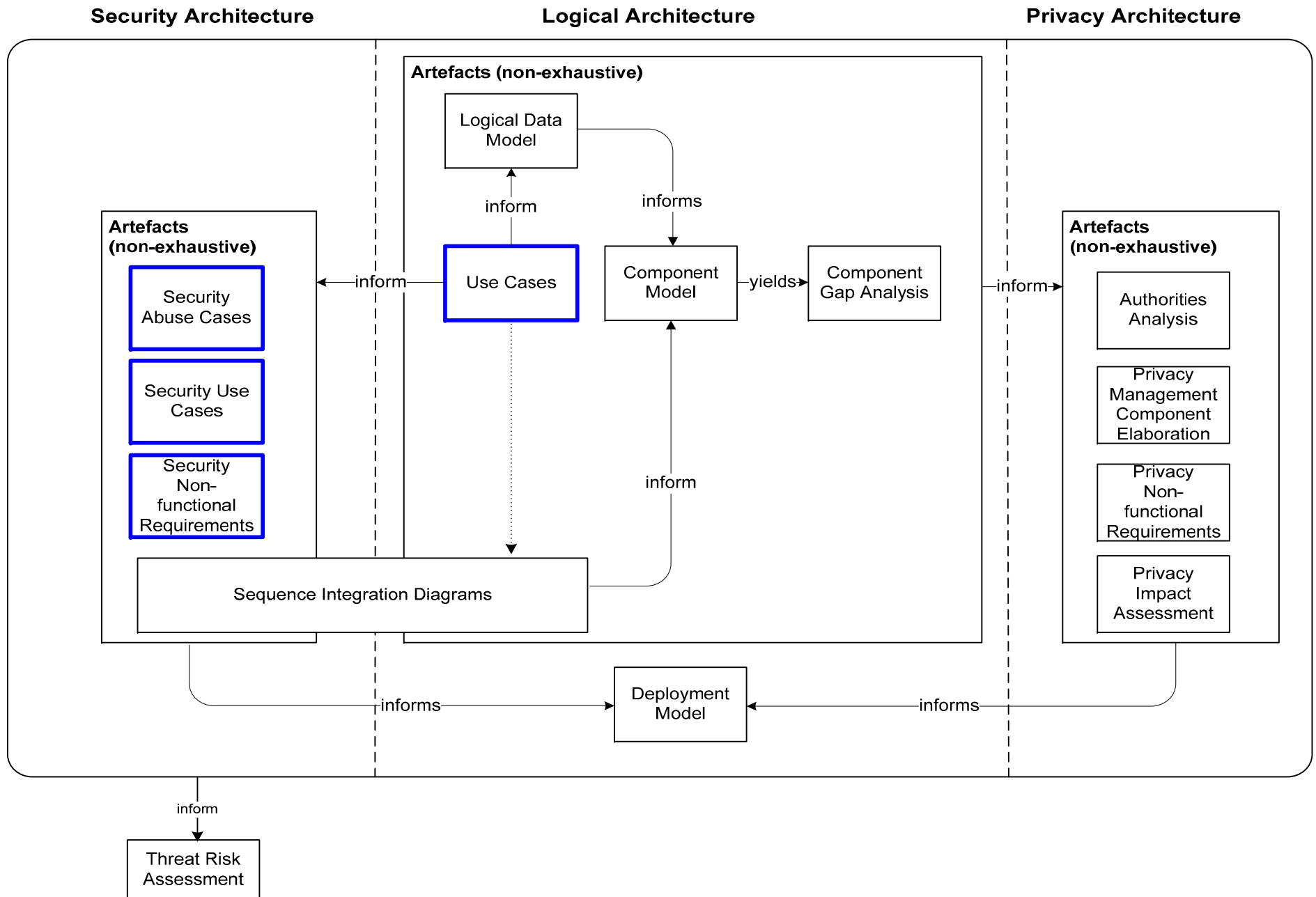
- **Develop Abuse Cases**
  - **Confidentiality**
  - **Integrity**
  - **Availability**
  - **Identification**
  - **Authentication**
  - **Authorization**
  - **Non-Repudiation**

- **Develop Security Use Cases**
- **Identify Strategies**
  - **Prevent**
  - **Contain**
  - **Detect and Notify**
  - **Collect and Track Events**
  - **Recover and Restore**
  - **Assure**
- **Identify Security Services**

## Legend

CP – Checkpoint  
F - Freeze

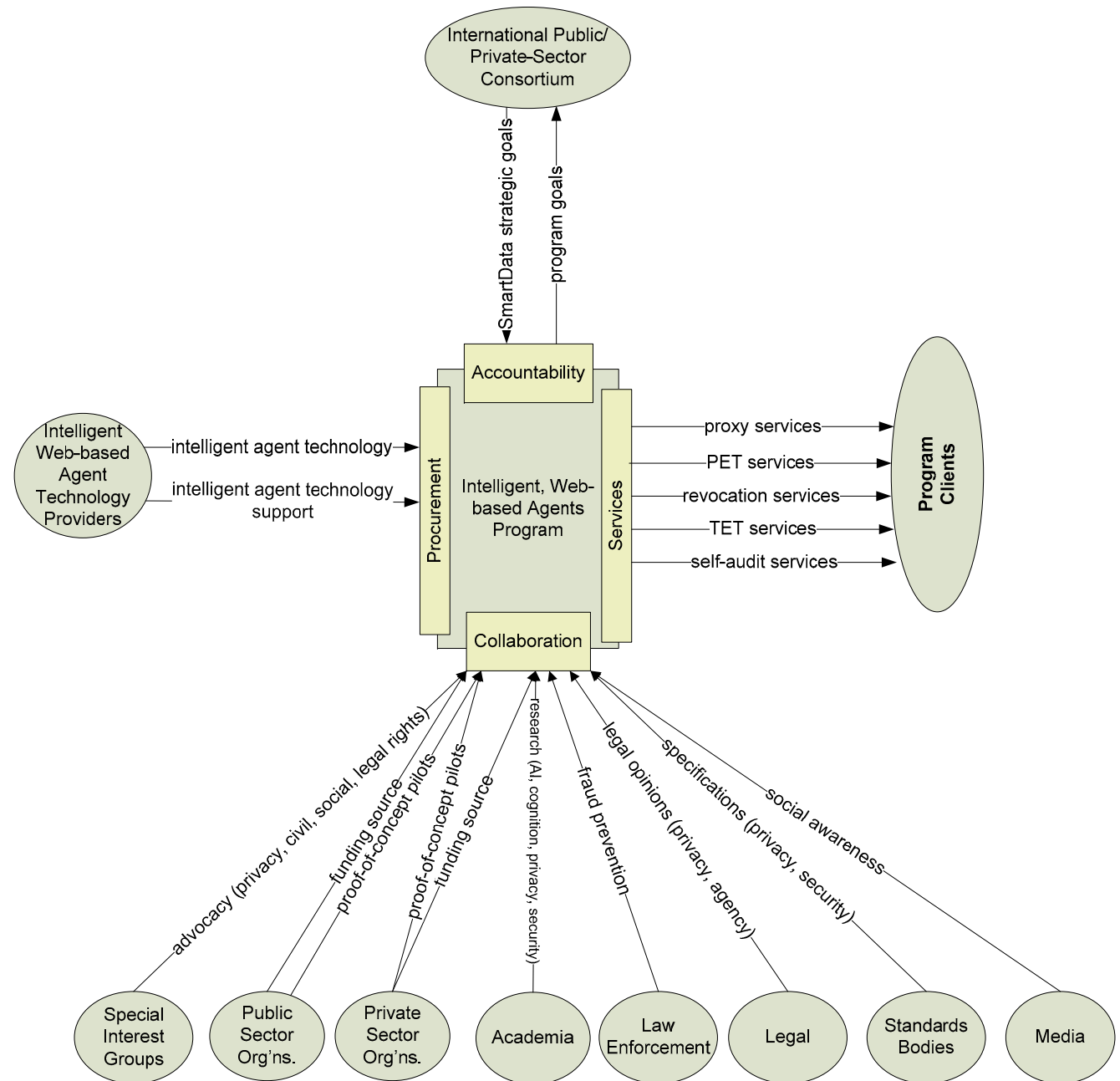
# Security Architecture, Logical Architecture, Privacy Architecture: Artefact Manufacture Relationships



# Business Architecture: Enterprise Context Model

## Key Points:

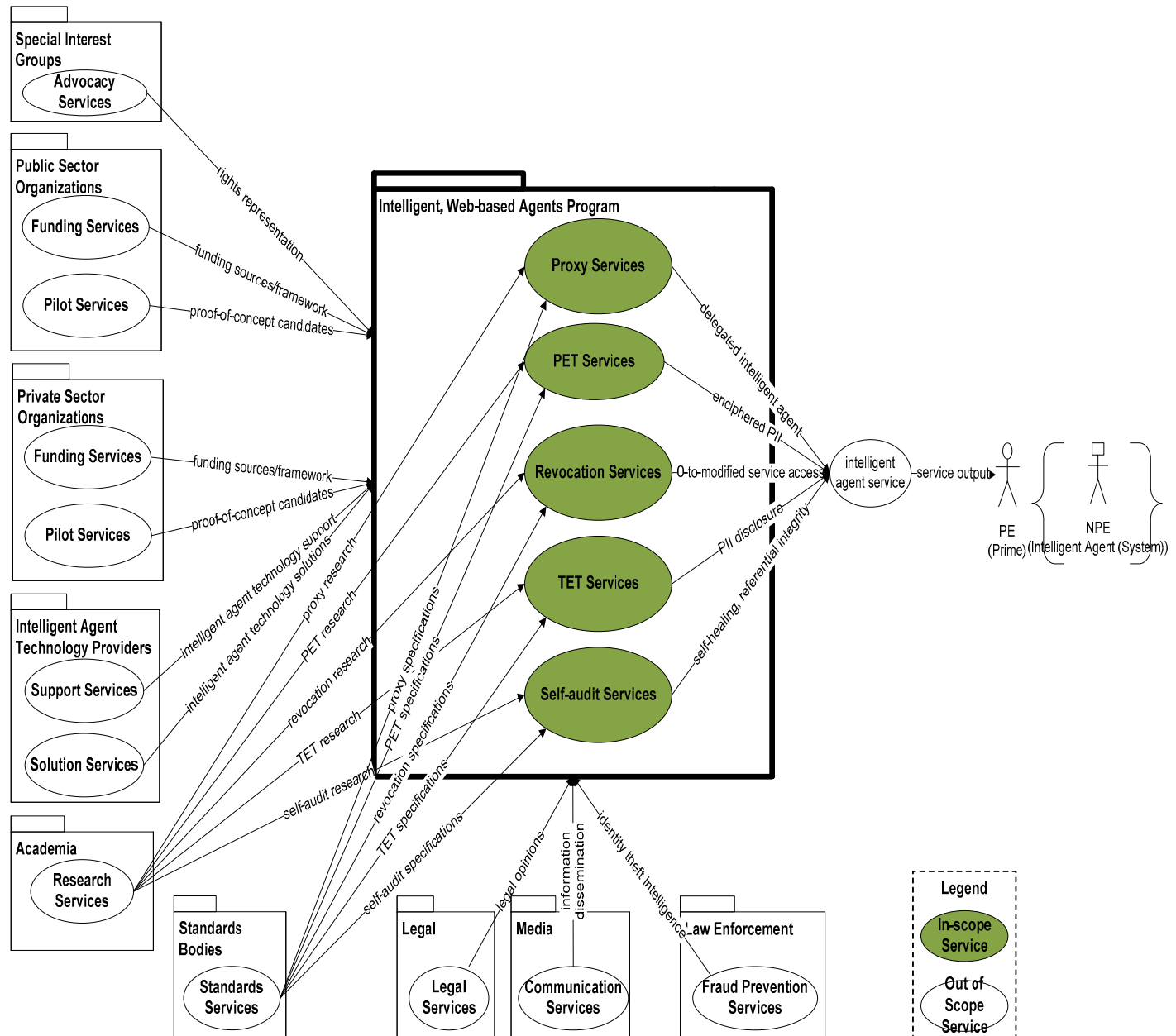
- Scope of interactions between the intelligent, web-based agents program and program clients, viz., persons requiring the services of intelligent, web-based agents to proxy in their stead in situations that include, but are not limited to, service fulfillment and other tasks otherwise undertaken by a person.



# Business Architecture: Service Integration and Accountability Model (SIAM)

## Key Points:

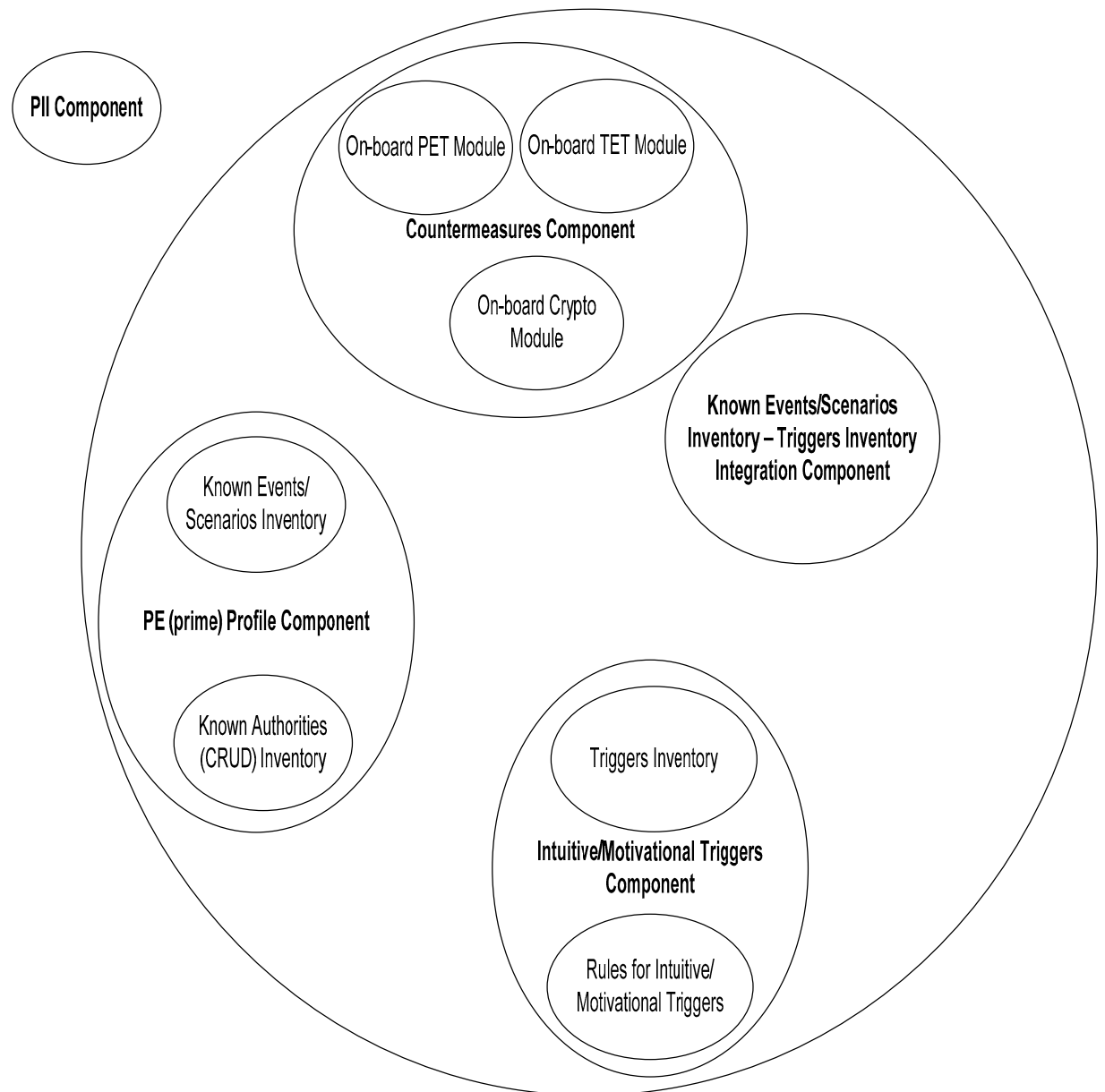
- ❑ The SIAM extends the enterprise context model by depicting accountability for service outputs.
- ❑ Services are associated with contributing organizations that provide input to the intelligent, web-based agents program, a specific service, or more than one service.



# NPE (Intelligent Agent) Conceptual Architecture Pattern: Components View

## Key Points:

- This view of the pattern exposes components required to design the automation target, or system, for an intelligent, web-based agent. Sub-components are grouped, or clustered, within parent components to emphasize like functionality within the automation target.

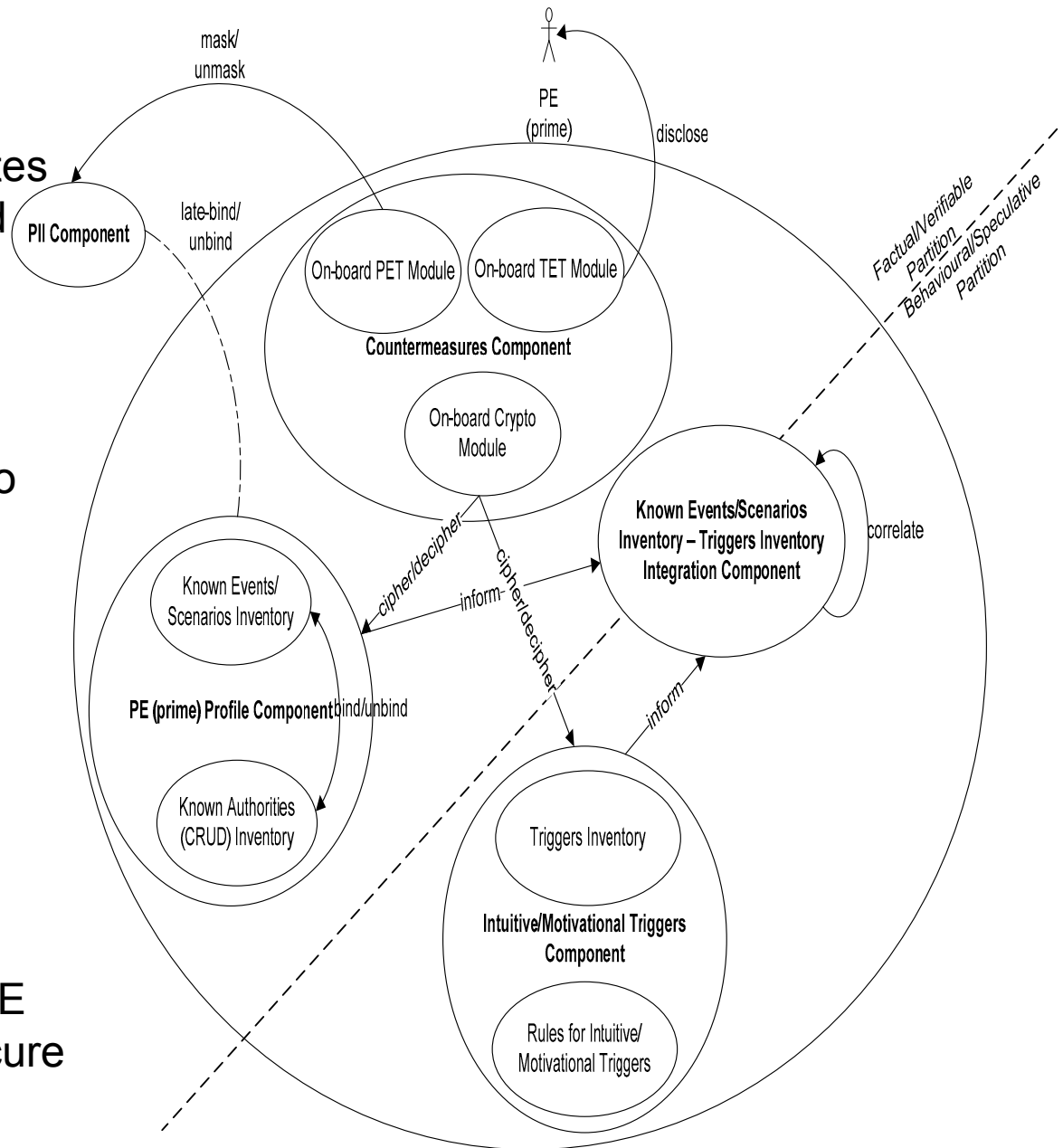




# NPE (Intelligent Agent) Conceptual Architecture Pattern: Relationships View

## Key Points:

- ❑ This view elaborates the components view and articulates the relationships between, and among, components.
- ❑ The PII component, the authoritative store for privacy-related information traceable to the PE (prime), is not encapsulated within the larger NPE frame.
- ❑ Late-binding of the PII component.
- ❑ Deliberate architectural discontinuity between the PII component and the parent NPE frame supports the goal of secure system design.



# Class Diagram: NonPersonEntity

## Key Points:

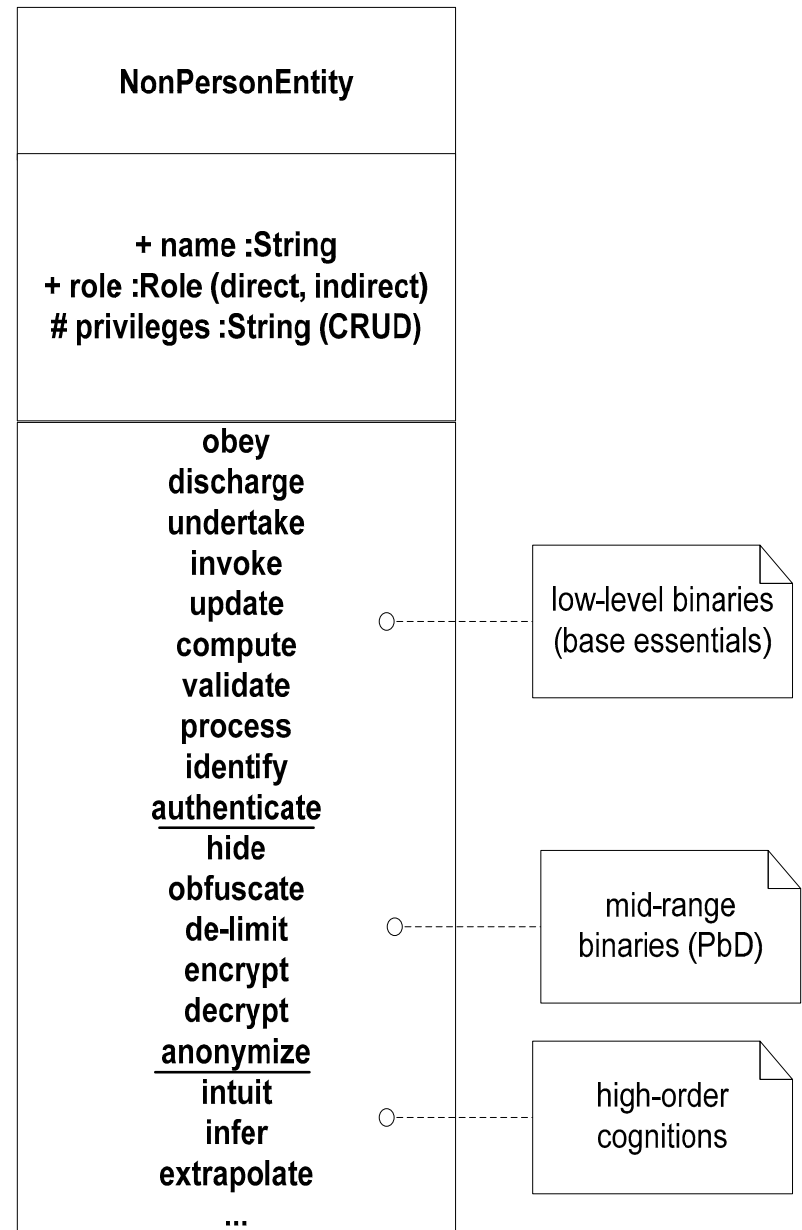
- ❑ The class diagram formalism is used to document intelligent, web-based agent elements, using the generally accepted namespaces of <<ClassName>>, <<Attributes>> and <<Operations>>, respectively.

<< **ClassName** >>

<< **Attributes** >>

<< **Operations** >>

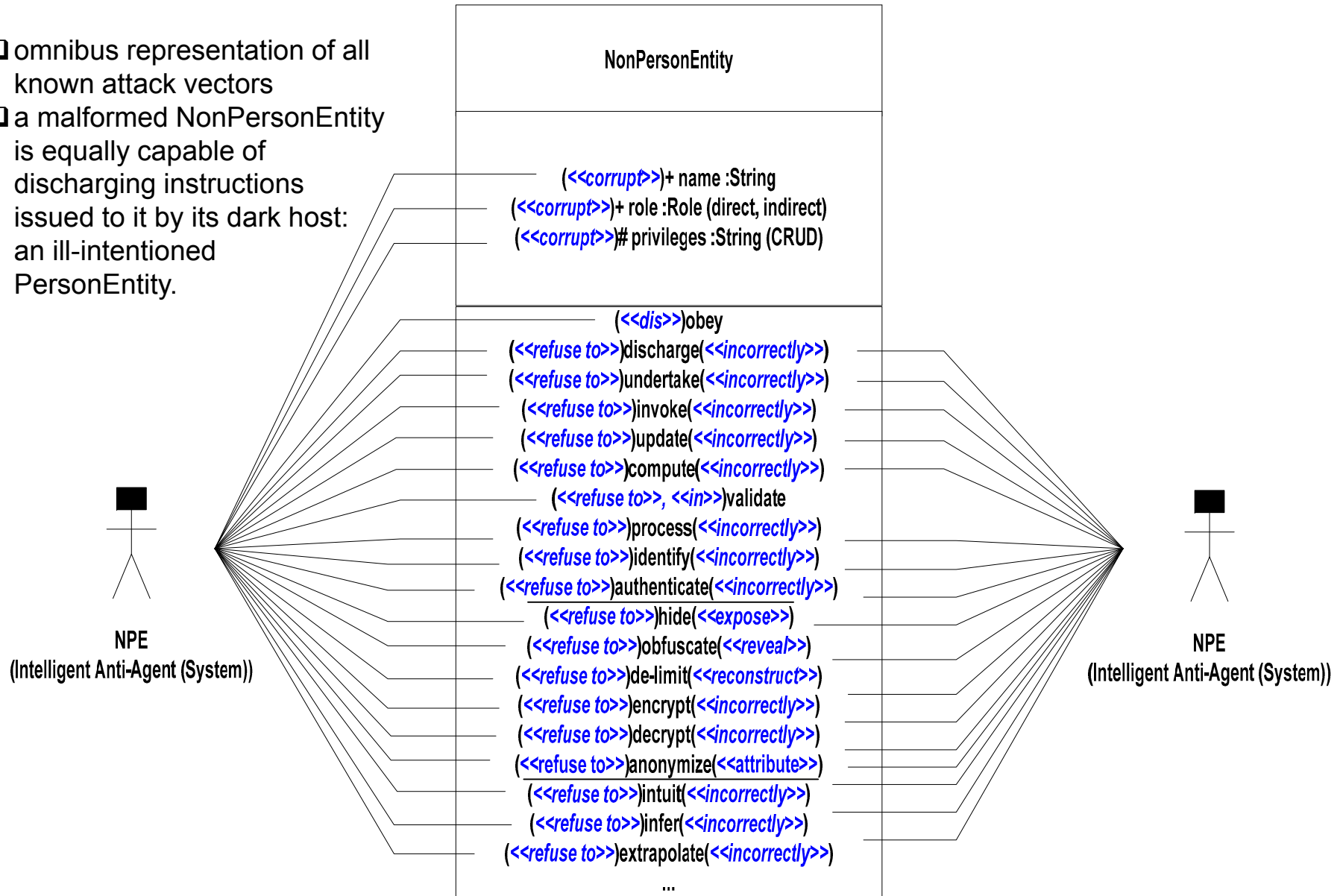
*Operations are constrained to the delegation context.*



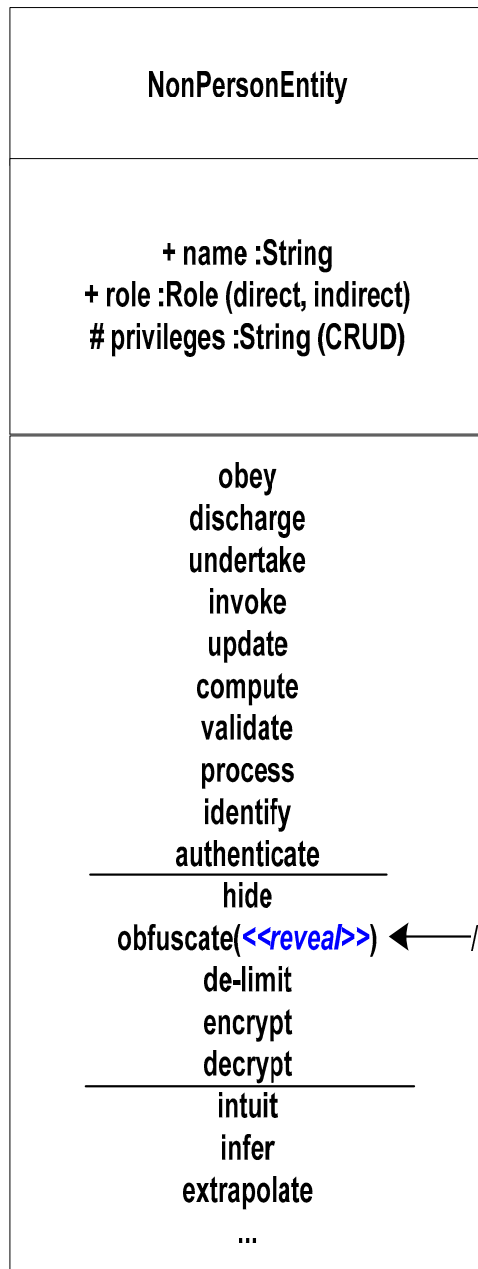
# Class Diagram: NonPersonEntity – Attack Vector Articulation

## Key Points:

- ❑ omnibus representation of all known attack vectors
- ❑ a malformed NonPersonEntity is equally capable of discharging instructions issued to it by its dark host: an ill-intentioned PersonEntity.

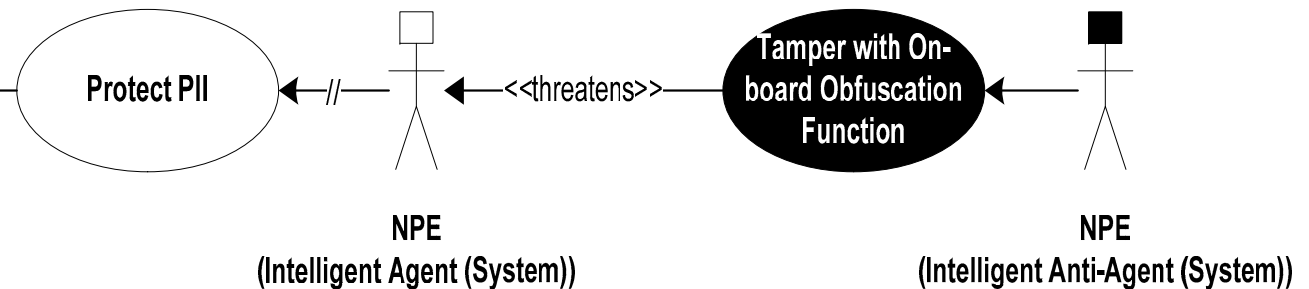


# Class Diagram: NonPersonEntity – Abuse Case Notation

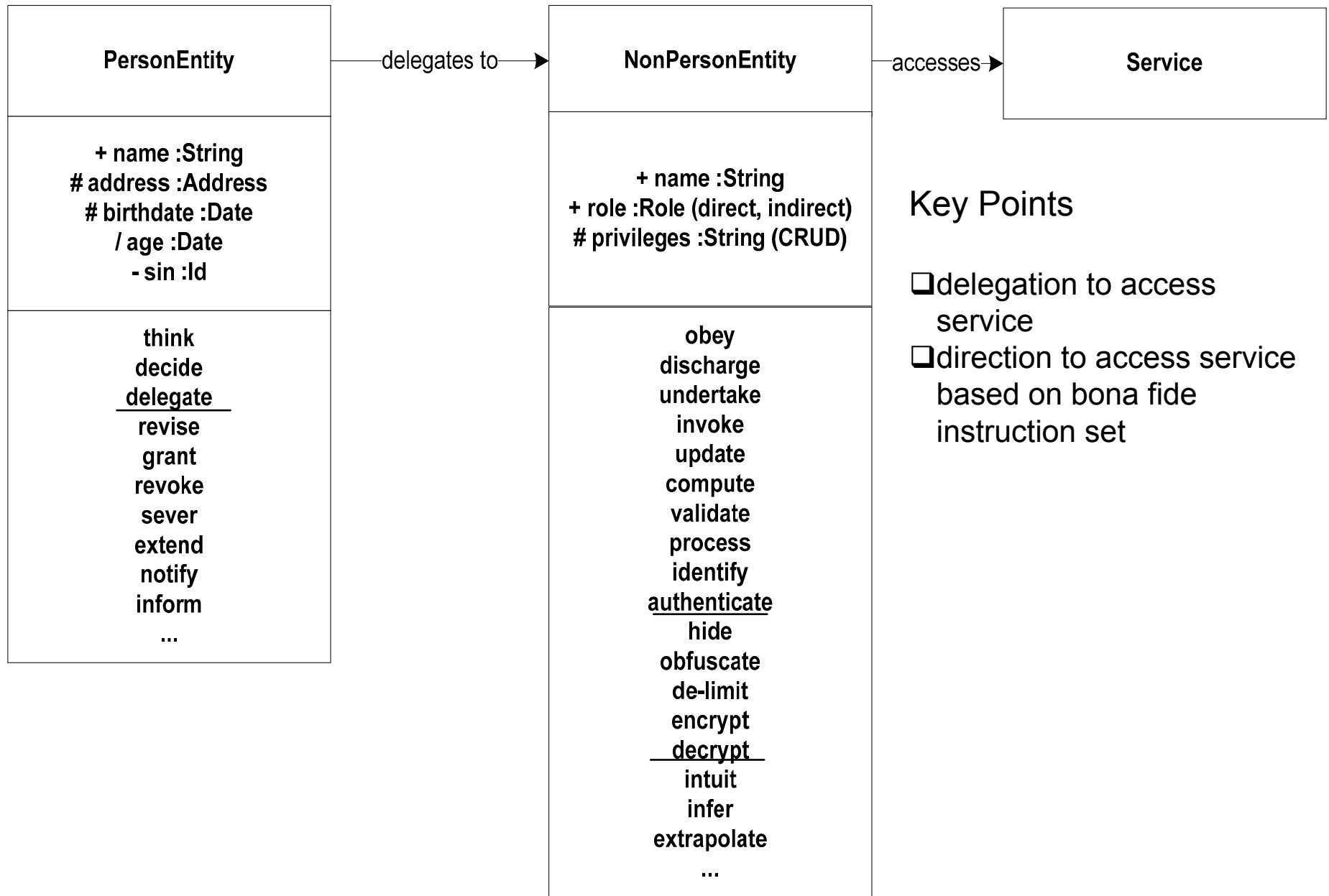


## Key Points:

- normative, “happy path” instruction set interrupt
- PII under attack via unauthorized manipulation of the obfuscation function
- PII disclosure to a rogue NonPersonEntity



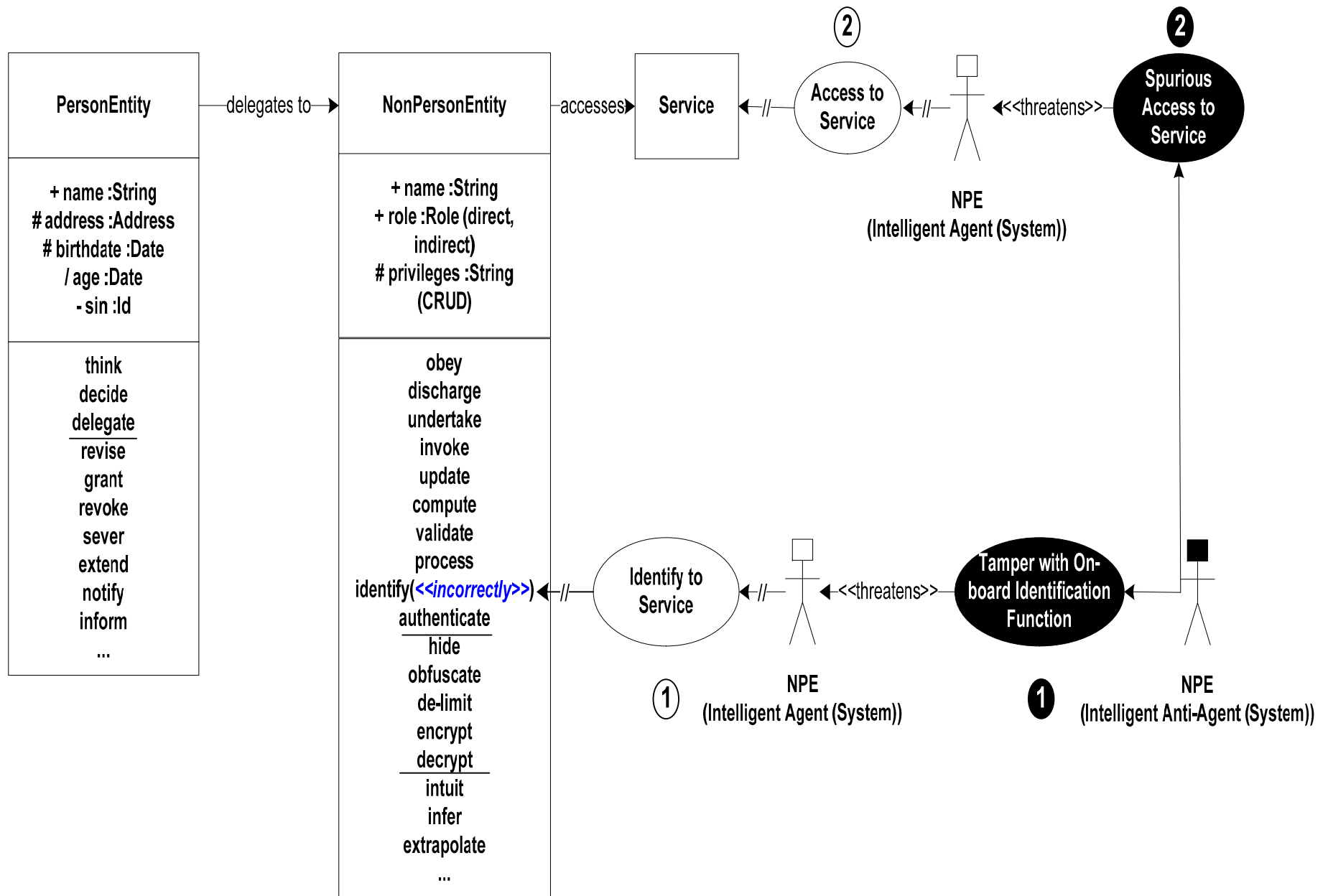
# Class Diagram: PE + NPE + Service (Normative, “Happy Path” Automation Target)



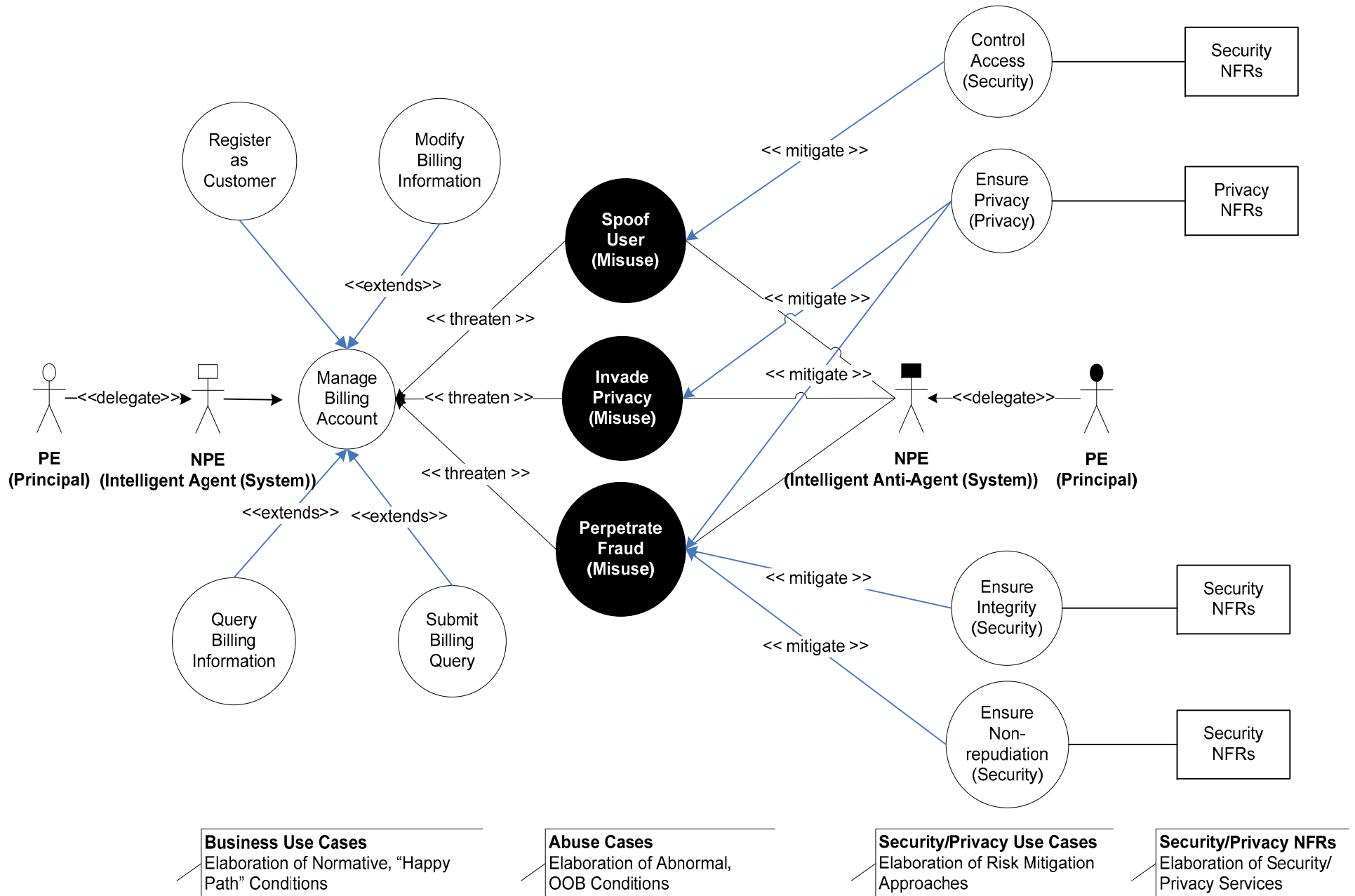
## Key Points

- delegation to access service
- direction to access service based on bona fide instruction set

# Class Diagram: PE + NPE + Service (Anomalistic Behaviour)



# Use and Abuse Case Scenarios



# Conclusions

- ❑ The operation of systems in a steady state is best served through architectural rigour, especially the security architecture domain.
- ❑ Upfront investment in deep articulation of a given scope of interest – some complex object - provides reasonable assurance that the instruction set (code) is:
  - well-behaved; and
  - conforms to the architectural representations on which it is based.
- ❑ Failure to describe a complex object at the level at which it can be comprehended will invariably lead to anomalies in expected behaviour and security gaps.
- ❑ The effort required to do so takes time and costs money; however, the absence of these models and artefacts means that the reliability, sustainability and authoritativeness of the complex object cannot be vouchsafed.
- ❑ “Pay me now or pay me later.”



## **STC 2014 - The 26th Annual IEEE Software Technology Conference**

The Security Architecture Discipline as Catalyst to  
Effect Secure Design:  
Employing Abuse Case Elaboration for Attack  
Vector Discovery and Countermeasure  
Determination

Murray Rosenthal, CISA, CRISC  
Risk Management & Information Security  
IT Strategic Planning & Architecture  
I&T Division  
City of Toronto

# Glossary of Terms

<b>Acronym</b>	<b>Long-form Definition</b>
CRUD	“C”reate, “R”ead, “U”pdate, “D”elete – generally accepted data-related actions
ECM	Enterprise Context Model – a business architecture artefact
NPE	Non-Person Entity – a virtual object
OOB	Out Of Band – an abnormal condition, situation or circumstance
PbD	Privacy by Design
PE	Person Entity – a human being
PET	Privacy Enhancing Technologies
PII	Personally Identifiable Information
SIAM	Service Integration and Accountability Model – a business architecture artefact
TET	Trust Enhancing Technologies