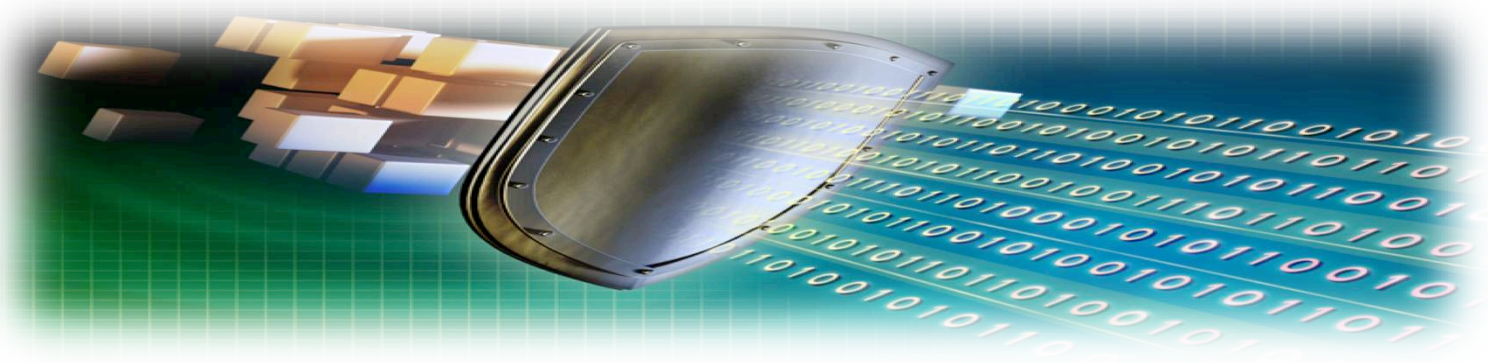
KDM Analytics

Working Together to Build Confidence

# Towards a common OMG Risk Metamodel

Dr. Nikolai Mansourov, CTO KDM Analytics



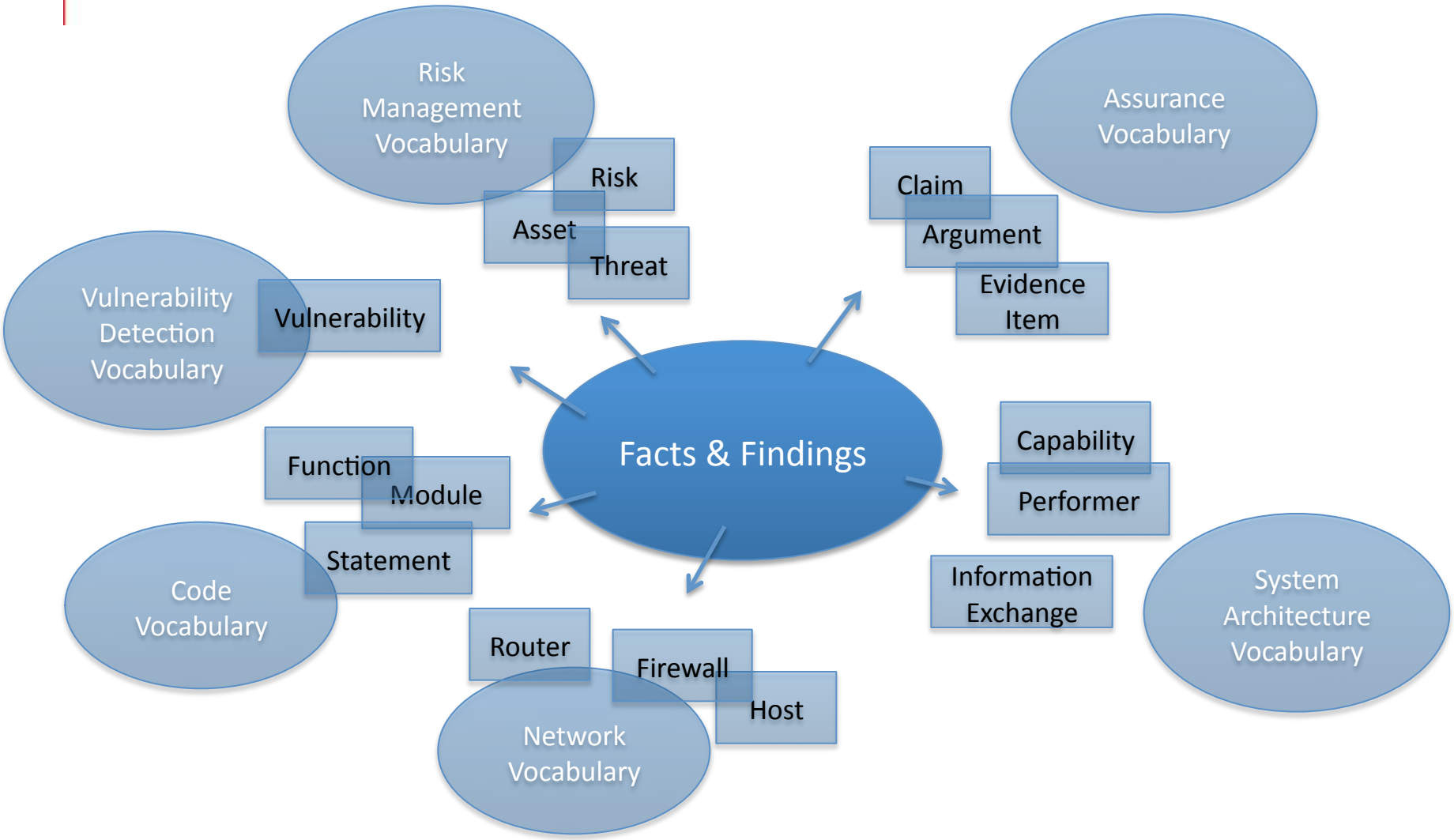
## *Who we are, what we do*

- Security Assurance company
  - Security risk assessments
    - Top-down, in the context of DoDAF views
    - Bottom-up, from source or binary code
  - Emphasis on automation
  - Emphasis on information sharing
    - OMG System Assurance Ecosystem
  - Systematic, repeatable methodology
    - FORSA
  - Efficient tool support
    - KDM Blade
    - Cameo Risk Analyzer, integrated with NoMagic Cameo Enterprise Architect

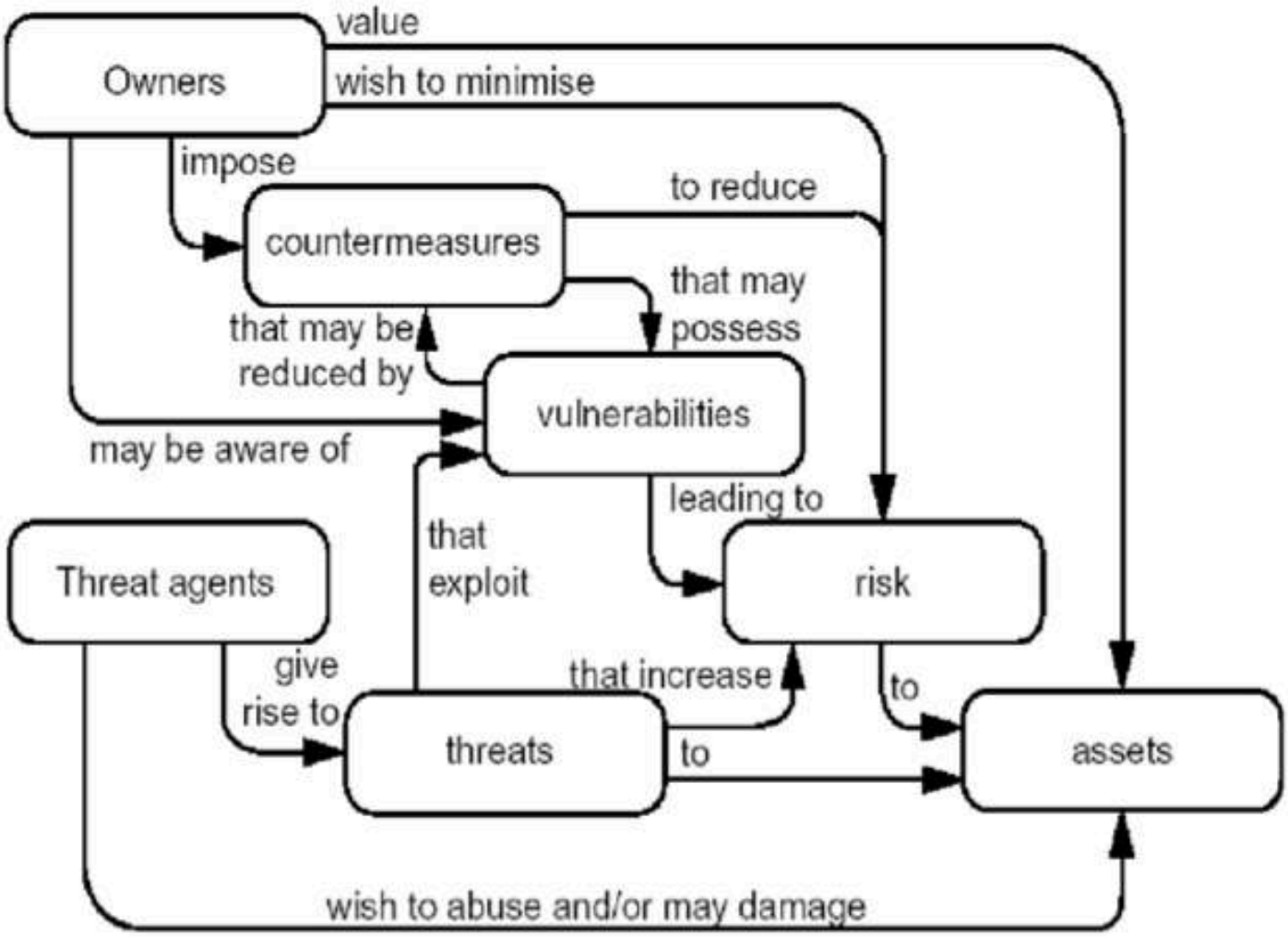
KDM Analytics



# *Tools, Interoperability, "Automated Everything"*



# What is security risk ? (ISO 15408)



# *Existing Threat and Risk Analysis methodologies*

- ISO/IEC 13335
- ISO/IEC 15408
- ISO/IEC 15443
- ISO/IEC 27001
- CRAMM (UK)
- EBIOS (France)
- Mehari (France)
- Magerit (Spain)
- HTRA (Canada)
- NIST SP-800-30 (US)
- Octave (SEI CMU)
- RiskAn (Czech Rep)
- Microsoft Threat Analysis Methodology
- Open Group FAIR
- & others

Challenges:

1) no interoperability;
2) few approaches deal with discernable concepts
3) few approaches are systematic enough to provide assurance



# *Failing to understand ALL risks …*

# *JUSTIFIABLE RISK ASSURANCE = RISK MANAGEMENT + ASSURANCE CASE*

*Brief first pass, more details later, time permitting*



# *Justifiable Risk Assurance*

Assurance Case
is related to the level of risk

Assurance Case
is structured according
to Risk Metamodel

Assurance Case

Risk Metamodel

Assurance Process delivers
assurance case
(argument + evidence)

Assurance Case
provides guidance
on how to collect evidence

Assurance Process

Risk Metamodel provides guidance
on how to collect evidence

Assurance Process — Generic guidance & tool

Report — Data specific to system



# *Justifiable Risk Assurance*

Assurance Case
is related to the level of risk

**System Model**

e.g. DoDAF views

Assurance Case
is structured according to
System Model

Assurance Case
is structured according
to Risk Metamodel

System Model
constrains evidence

Risk Metamodel
describes evidence

**Risk Report / Risk Manage-ment Plan**

Assurance Case

**Evidence**

Risk Metamodel

Assurance Case
structures
report

Evidence justifies claims

Assurance Process defines
the steps to collect
evidence

Assurance Process delivers
assurance case
(argument + evidence)

Assurance Case
provides guidance
on how to collect evidence

Assurance Process

Risk Metamodel provides guidance
on how to collect evidence

Assurance Process — Generic guidance & tool

Report — Data specific to system



# *Our FORSA methodology*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis

FORSA:
- Fact Oriented
- Repeatable
- Security
- Assessment

The sequence of steps is designed to increase *confidence*, and therefore increase *assurance*

# *Operational Risks and DODAF 2.0*

General information about the mission, context (not sufficient for automated analysis)

1 — AV1, AV2, OV1

Capability views; useful for analysis In combination with OV views — CV2,CV4

2

OV2

3

Operational resource flow; introduces performers and Information exchanges; suitable for automated analysis; gives operational overview of the entire enterprise; perfect for threat and risk analysis; but may not be sufficient for human interpretation without additional information, which may come either from informal sources or more detailed views

**Operational path**

OV3, OV5, CV6

4

Operational activities

5 — DIV1, DIV2

OV6 (a,b,c)

7

Conceptual and logical data

DIV3

8

Detailed Operational activities

Physical data

**Services path**

**Systems path**

6a — Service decomposition

SvcV1, SvcV2, SvcV3b, SvcV5, CV7

6b — Systems decomposition

SV1, SV2, SV3, SV5b

6c — SvcV3a

Service-System mapping

SvcV4, SvcV6

9a — Service functions

Svc10 (a,b,c)

10a — Detailed Service functions

9b — System functions

SV4, SV5a

SV10 (a,b,c)

10b — Detailed System functions

# *OVERVIEW OF THE RISK METAMODEL FOR JUSTIFIABLE RISK ASSURANCE*



# *Overview of the Risk Metamodel*

Who cares ?

*Owners and criteria*

To what ?

*Assets and Targets*

Identified Risks

How severe ?

*Undesired events*

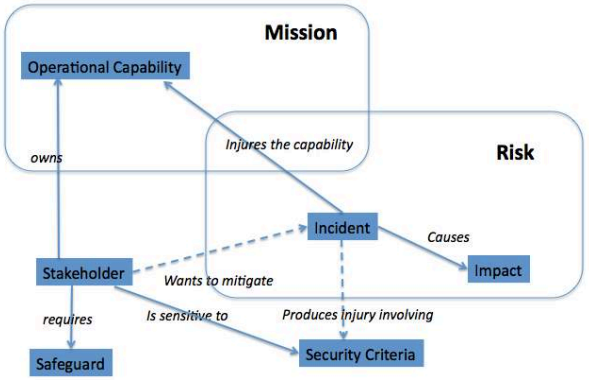
How likely ?

*Attack scenarios*

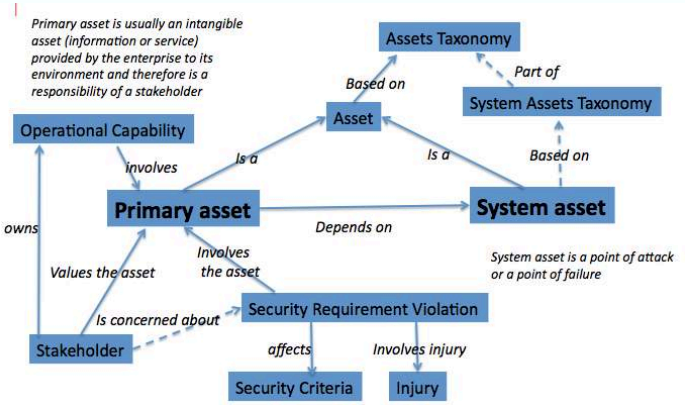
What to do about it ?

*Controls, mitigation options*

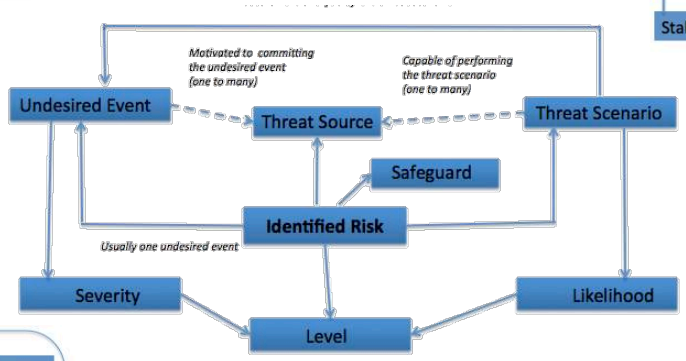
# Overview of the Risk Metamodel



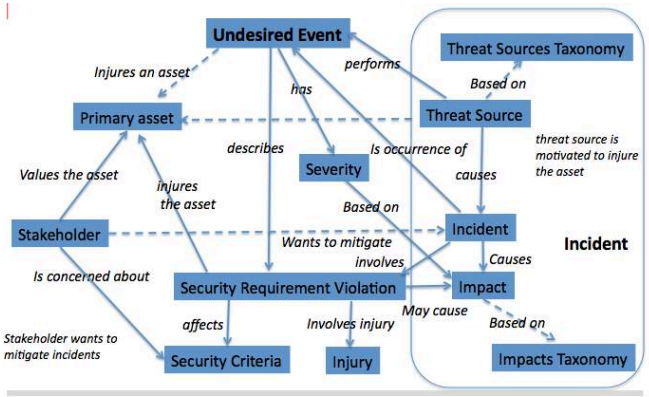

Owners and criteria




Assets and Targets

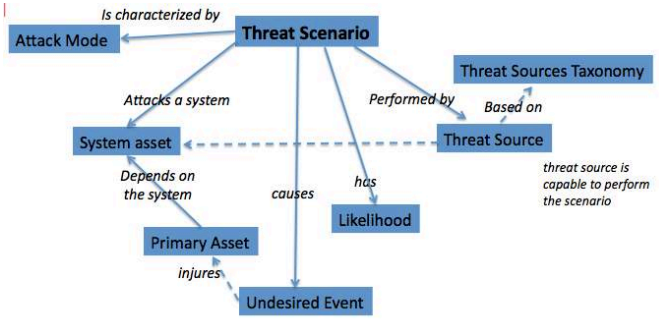



Identified risks




Undesired events

Controls,
Mitigation options



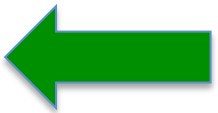

Attack scenarios

# *PART 1: THE OPERATIONAL CONTEXT*

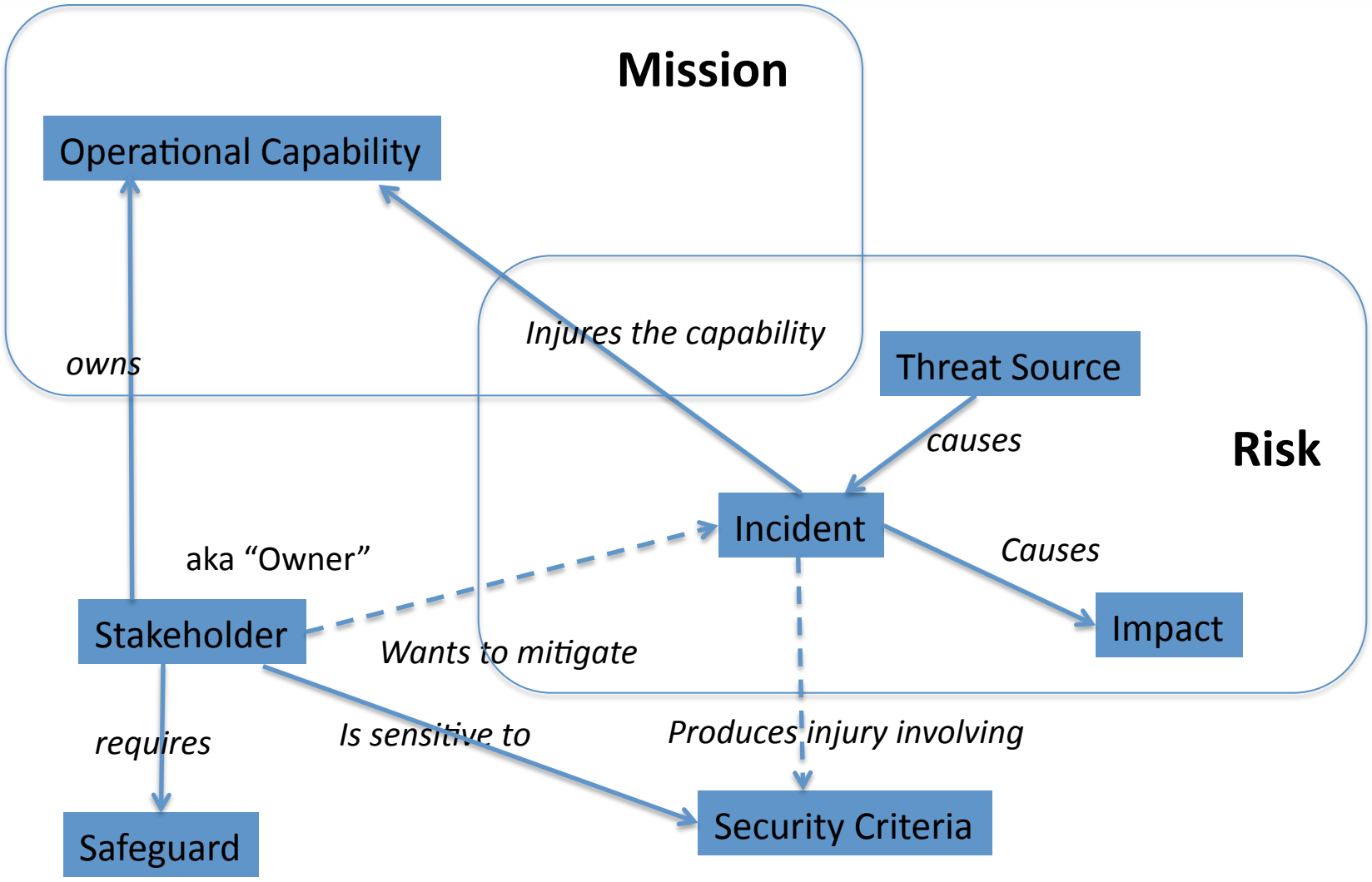


## FORSA STEPS

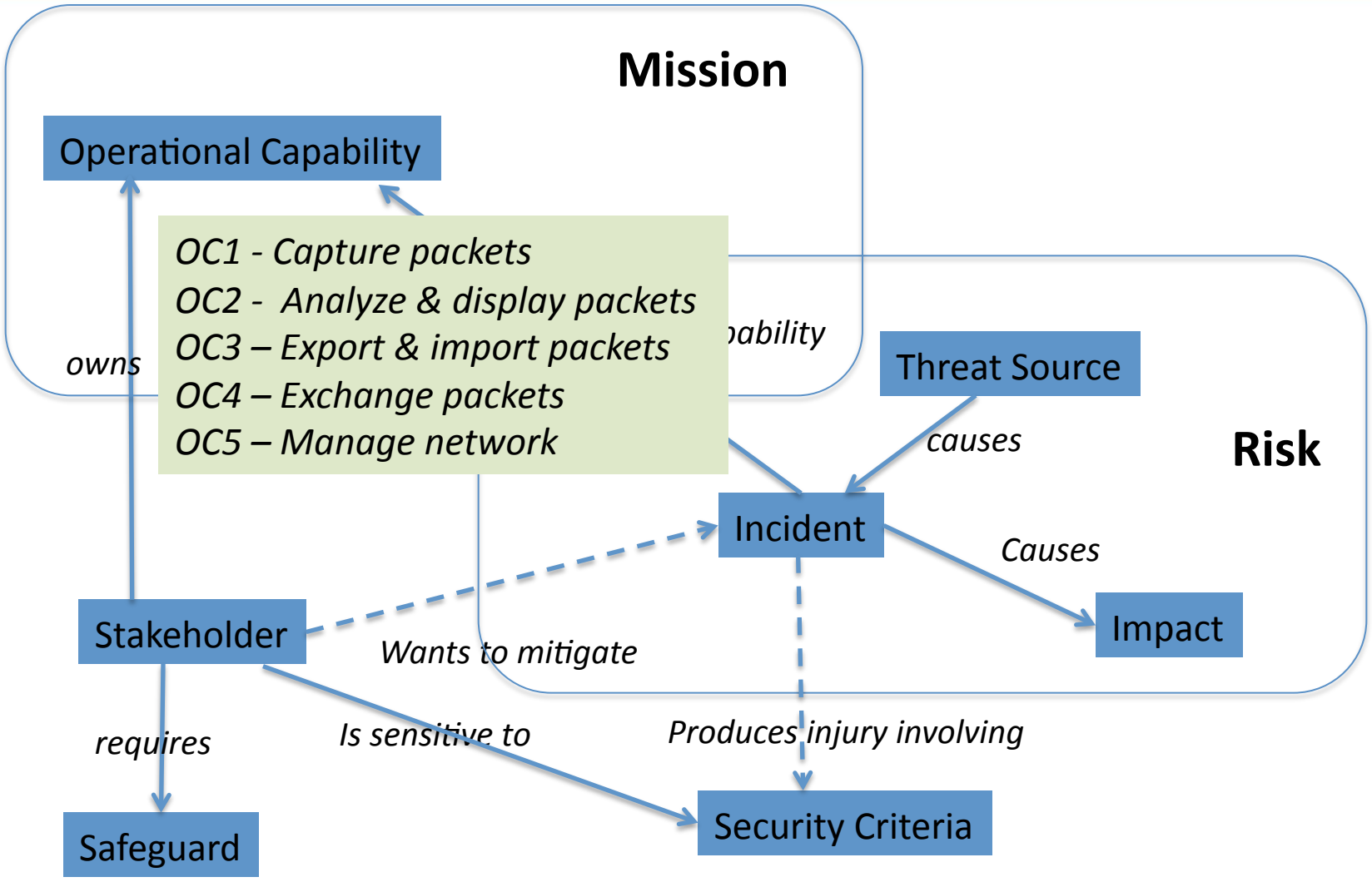
1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



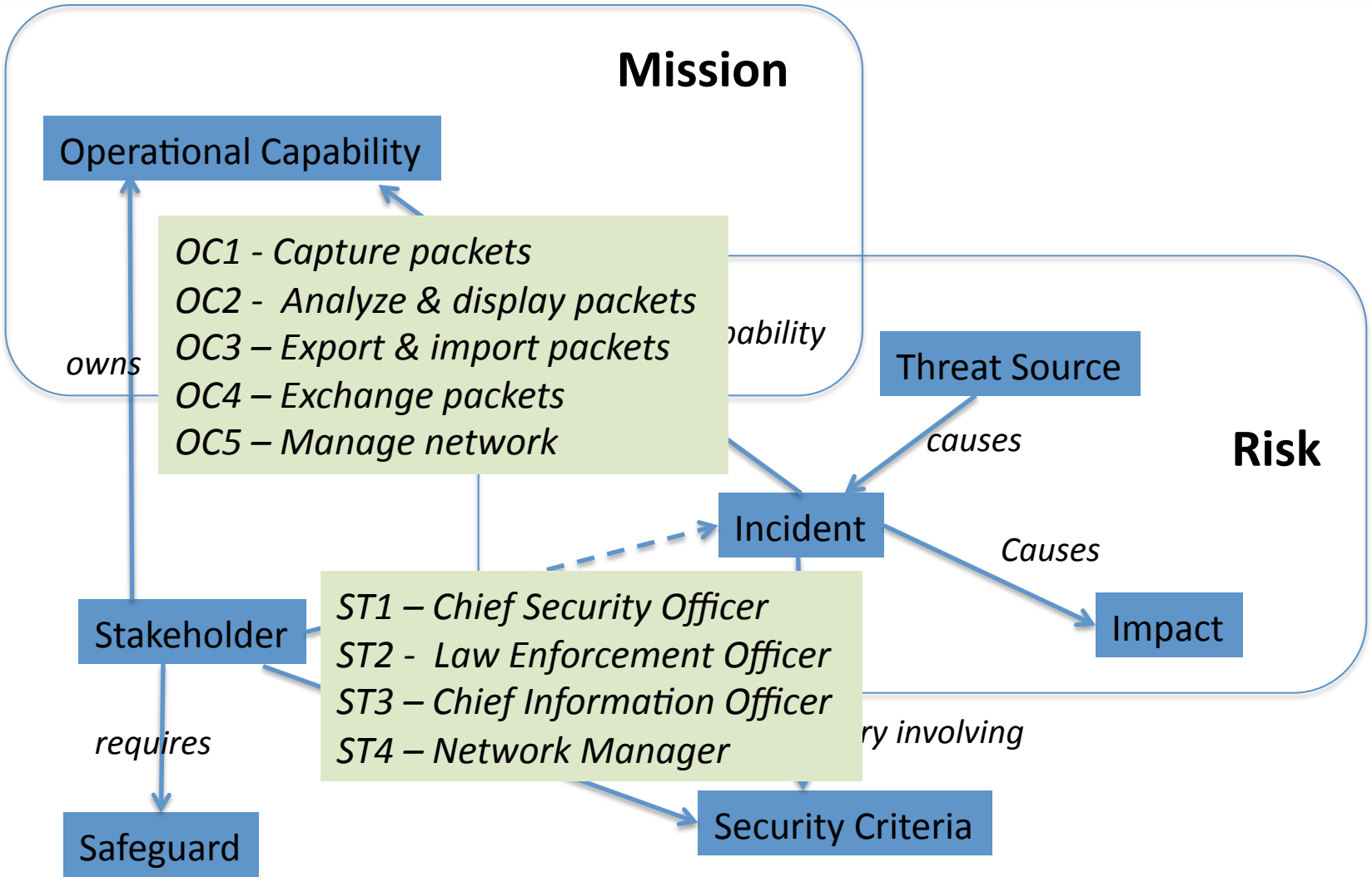
# *Analysis of the operational context is based on the following model*



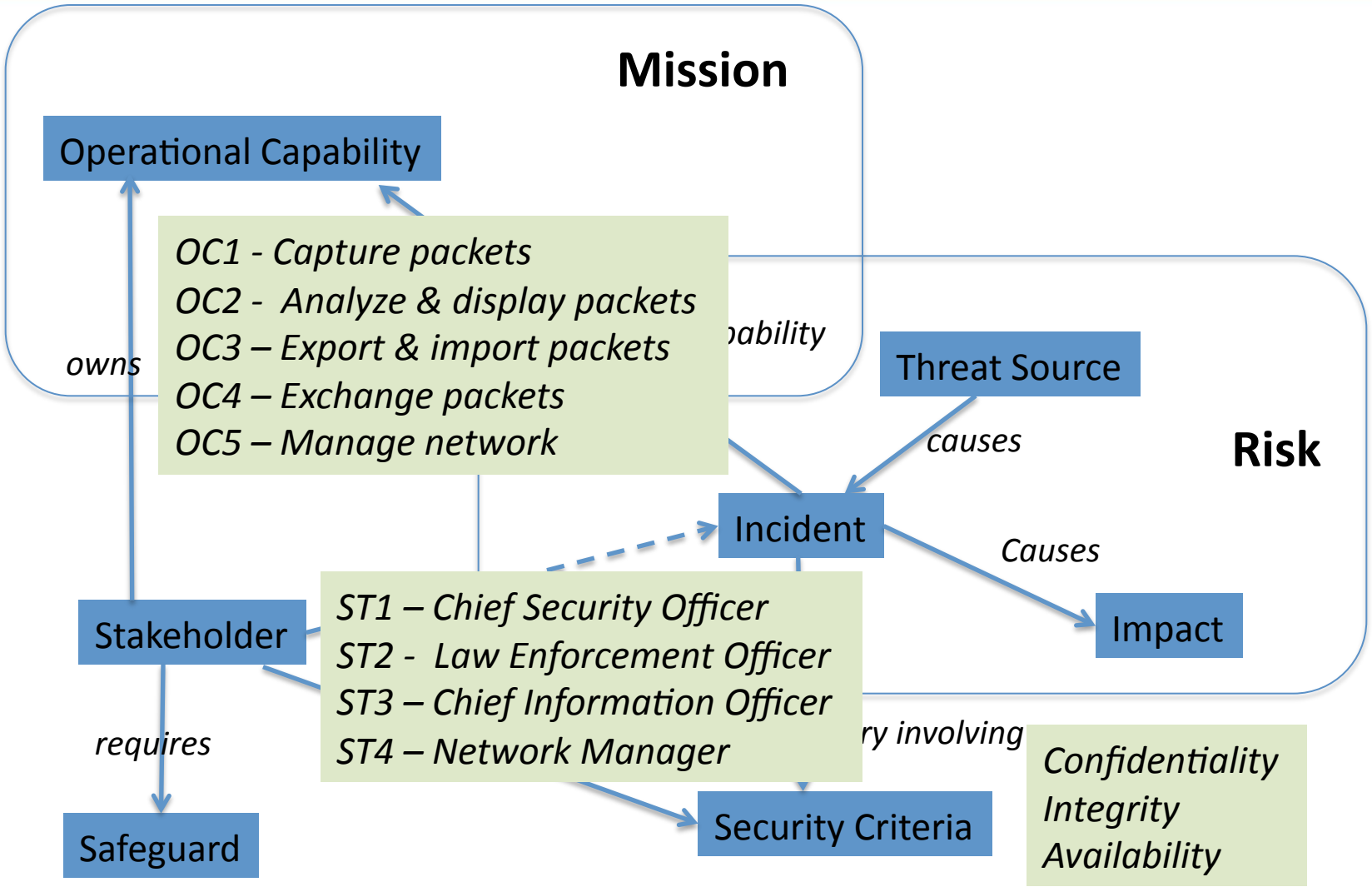
# *Analysis of the operational context is based on the following model*



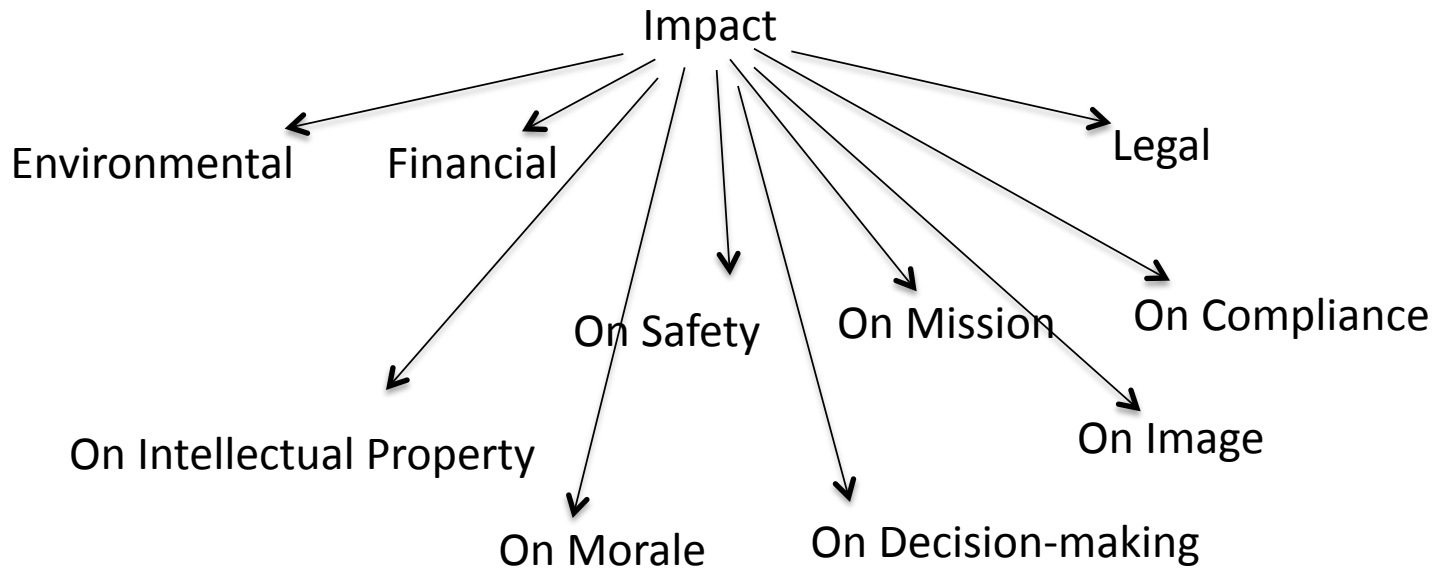
# *Analysis of the operational context is based on the following model*



# *Analysis of the operational context is based on the following model*



# *"Standard" Taxonomy of Impacts*

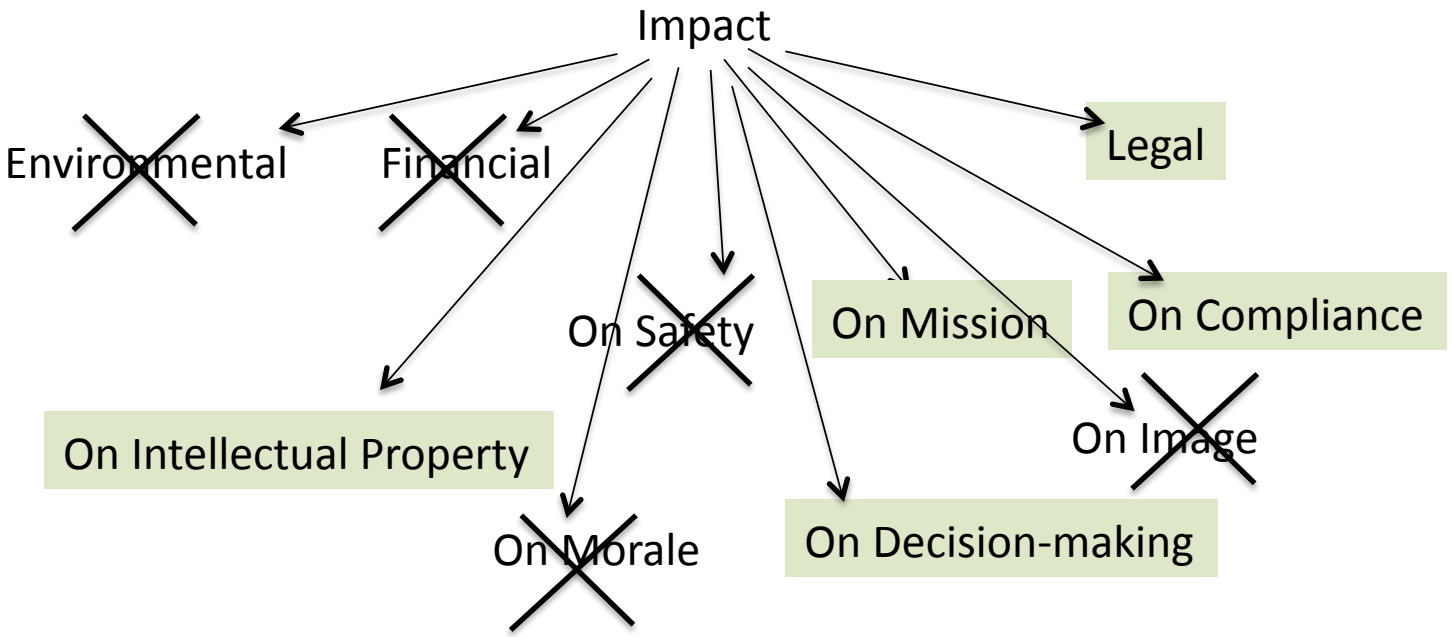



Impacts are related to:
-Identification of the primary assets
-Injury and undesired events

*Based on EBIOS and HTRA*

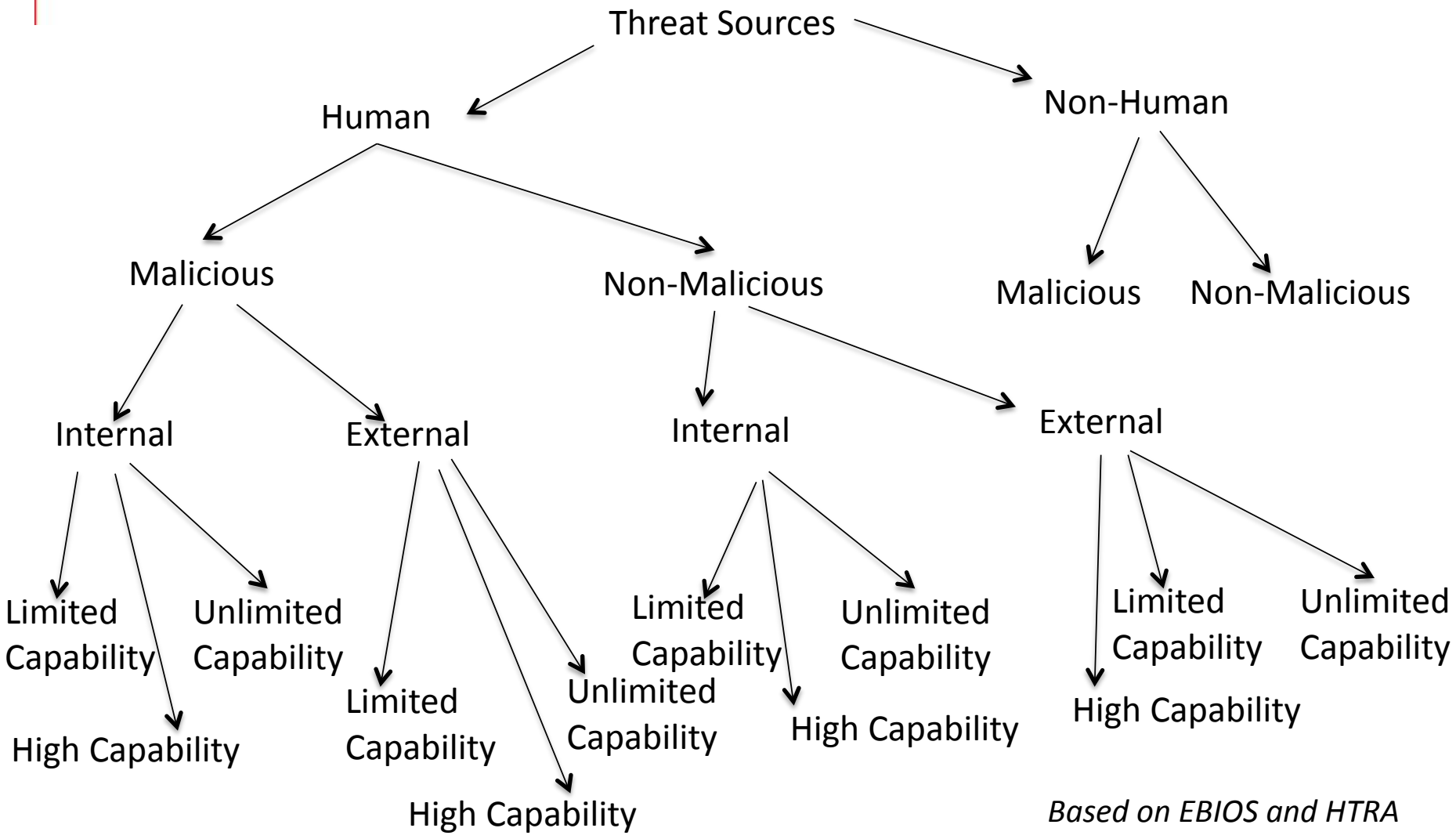


# *Taxonomy of Impacts in scope*



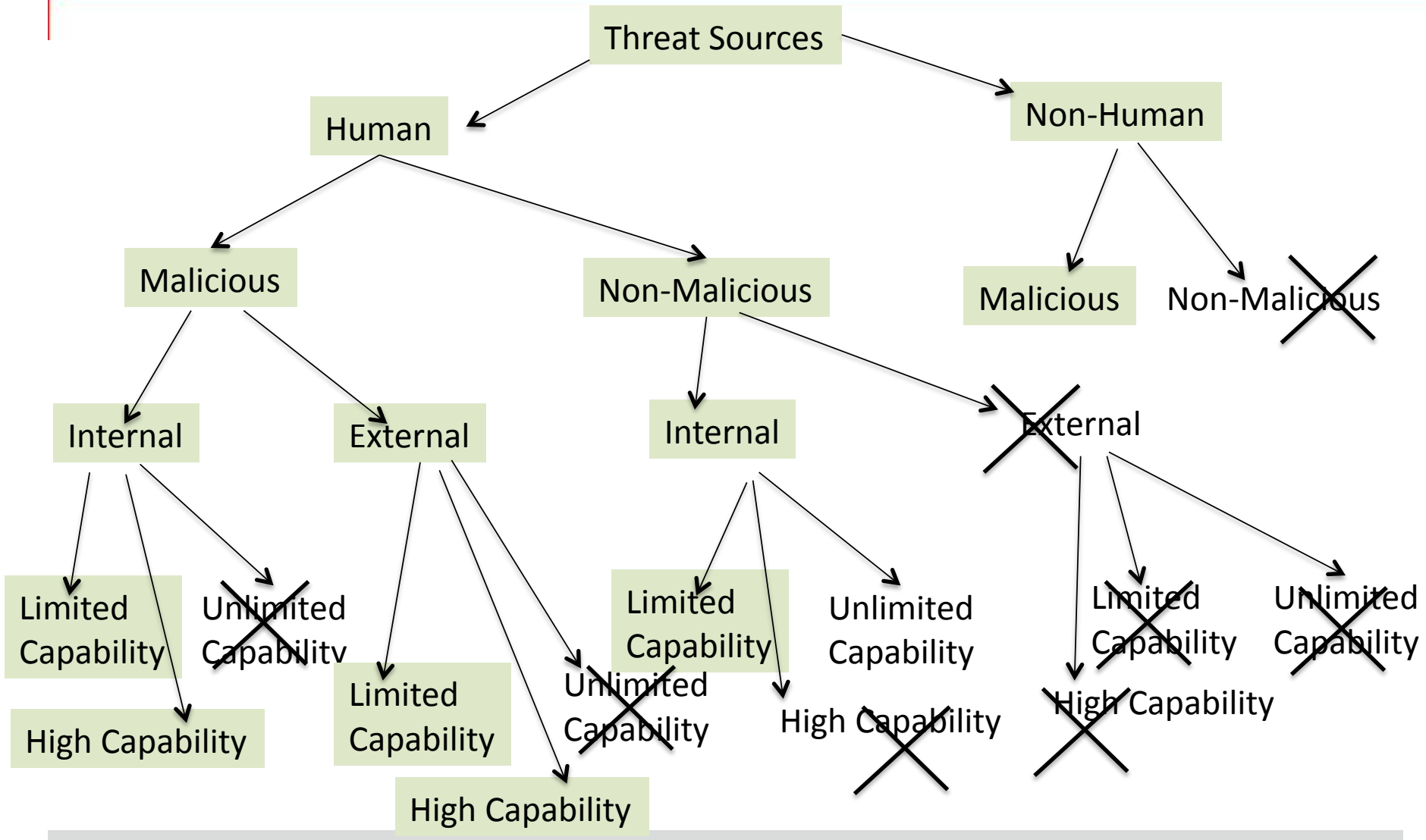

Impacts are related to:
-Identification of the primary assets
-Injury and undesired events

# *"Standard" Taxonomy of threat sources*



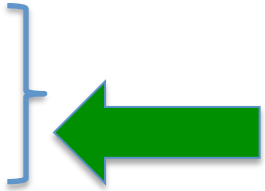

*Based on EBIOS and HTRA*

# *Taxonomy of threat sources in scope*



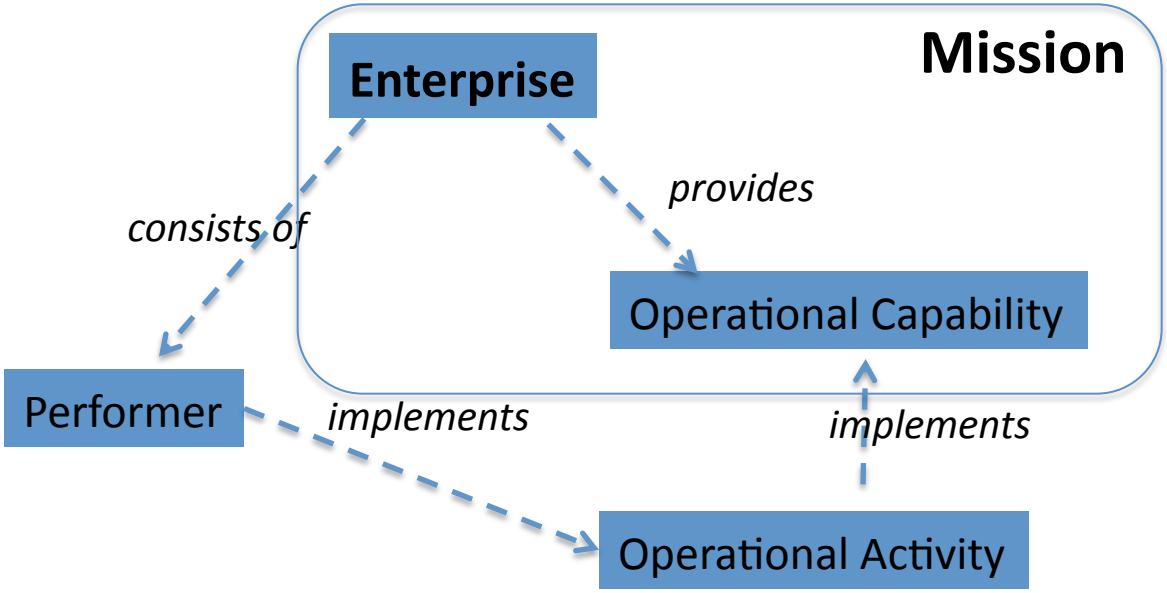
## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Threat Scenario Identification
6. Threat Scenario Analysis
7. Risk Identification
8. Safeguard Identification
9. Vulnerability Analysis
10. Risk Analysis

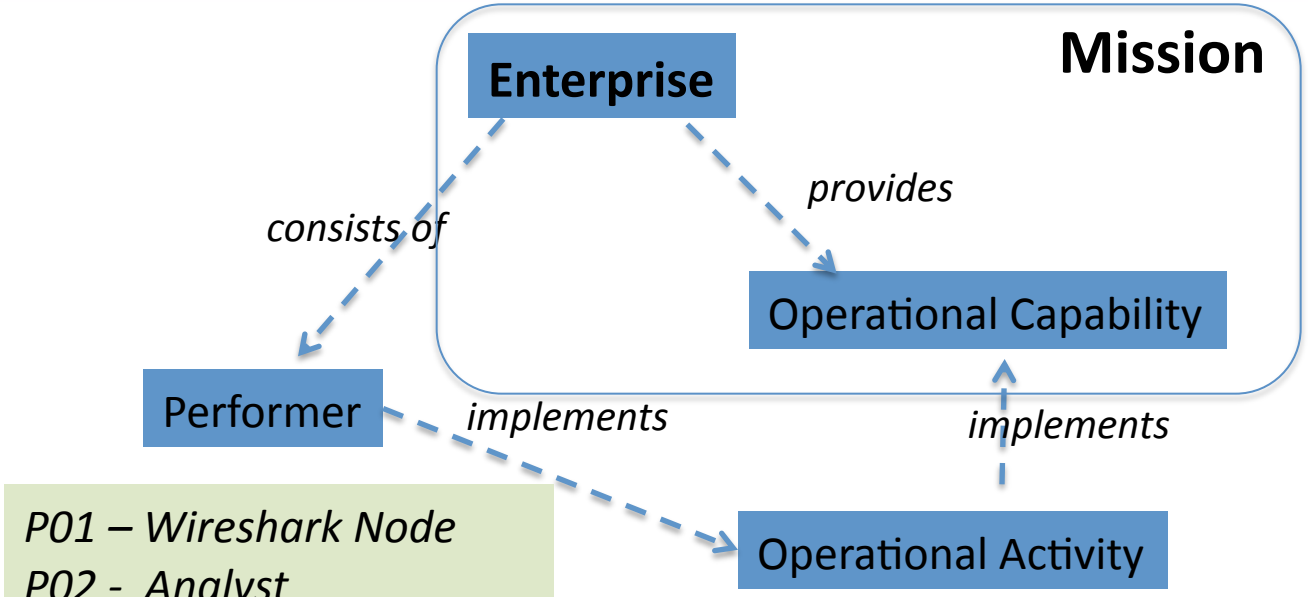




# *Analysis of System Facts is aligned with Business Models*

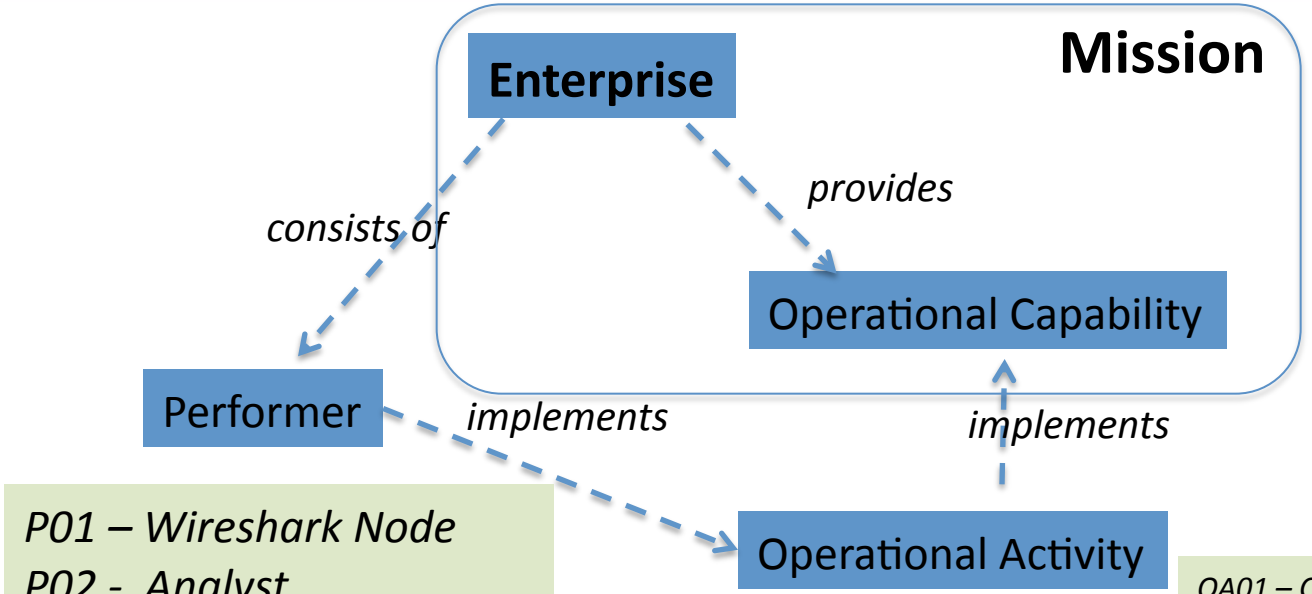


# *Analysis of System Facts is aligned with Business Models*




*P01 – Wireshark Node*
*P02 - Analyst*
*P03 – Network*
*P04 – Network User*
*P05 – Stored capture files*
*P06 - Admin*

# Analysis of System Facts is aligned with Business Models




*P01 – Wireshark Node*
*P02 - Analyst*
*P03 – Network*
*P04 – Network User*
*P05 – Stored capture files*
*P06 - Admin*

*OA01 – Configure Network Interfaces*
*OA02 - Start Capture*
*OA03 – Stop Capture*
*OA04 – Save Capture Trace*
*OA05 – Analyze Trace*
*OA06 – Open Capture Trace*
*OA07 – Annotate Trace*
*OA08 – Send Network Packet*
*OA09 – Receive Network Packet*
*OA10 – Tap Network Packet*
*OA11 – Dissect Network Packet*
*OA12 – Add Network Packet to Trace*
*OA13 – Manage Network*

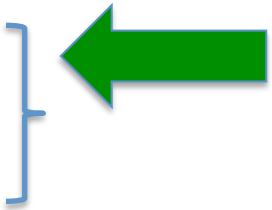


# *PART 2: THE EFFECT ANALYSIS*



## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



# *Analysis of Assets is based on the following model*

*Primary asset is usually an intangible asset (information or service) provided by the enterprise to its environment and therefore is a responsibility of a stakeholder*

Assets Taxonomy

*Part of*

System Assets Taxonomy

*Based on*

Asset

Operational Capability

*involves*

*Is a*

*Is a*

*Based on*

**Primary asset**

**System asset**

*owns*

*Depends on*

*Involves the asset*

*System asset is a **point of attack** or a point of failure*

*Values the asset*

Sensitivity

*Is concerned about*

Stakeholder

*affects*

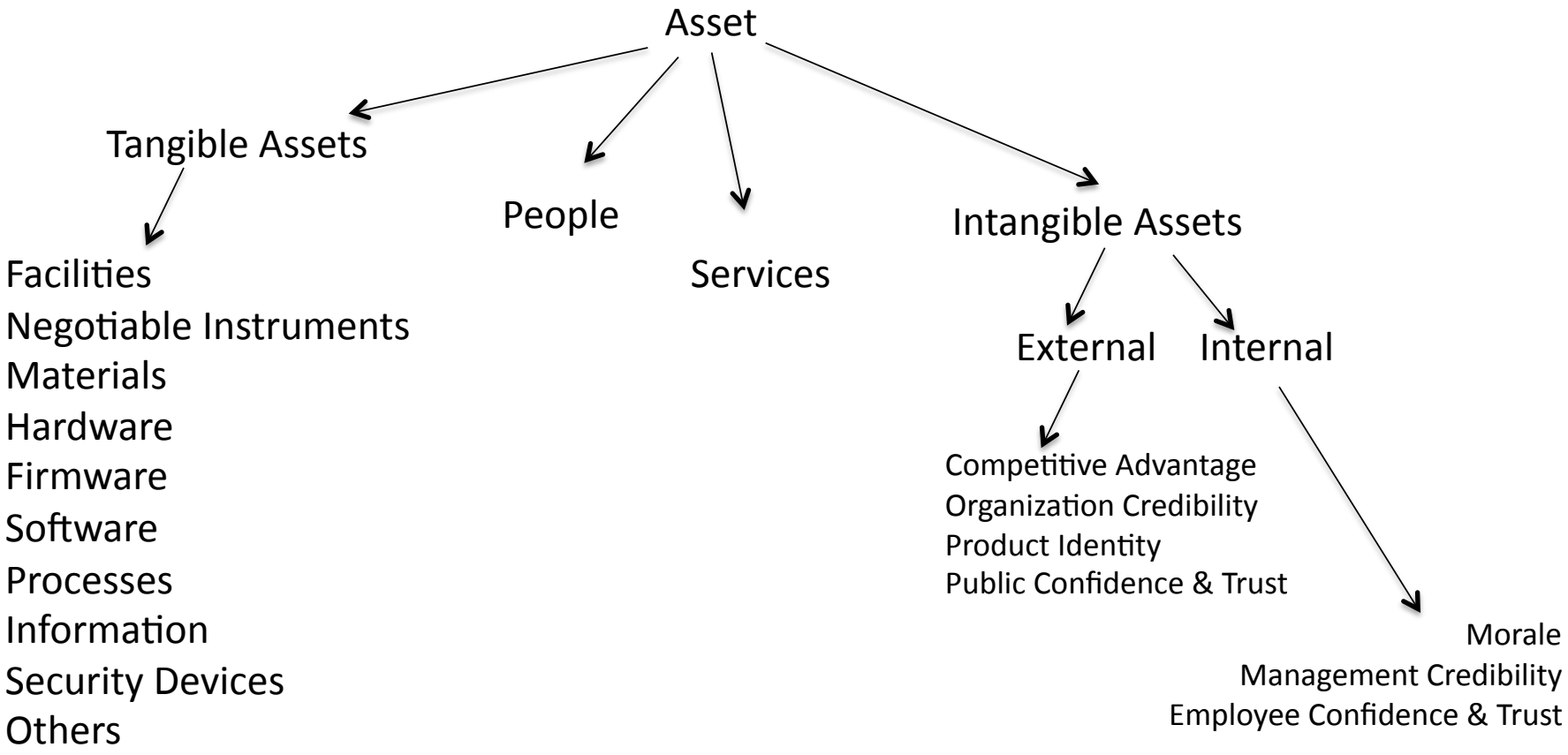
*Involves injury*

Security Criteria

Injury

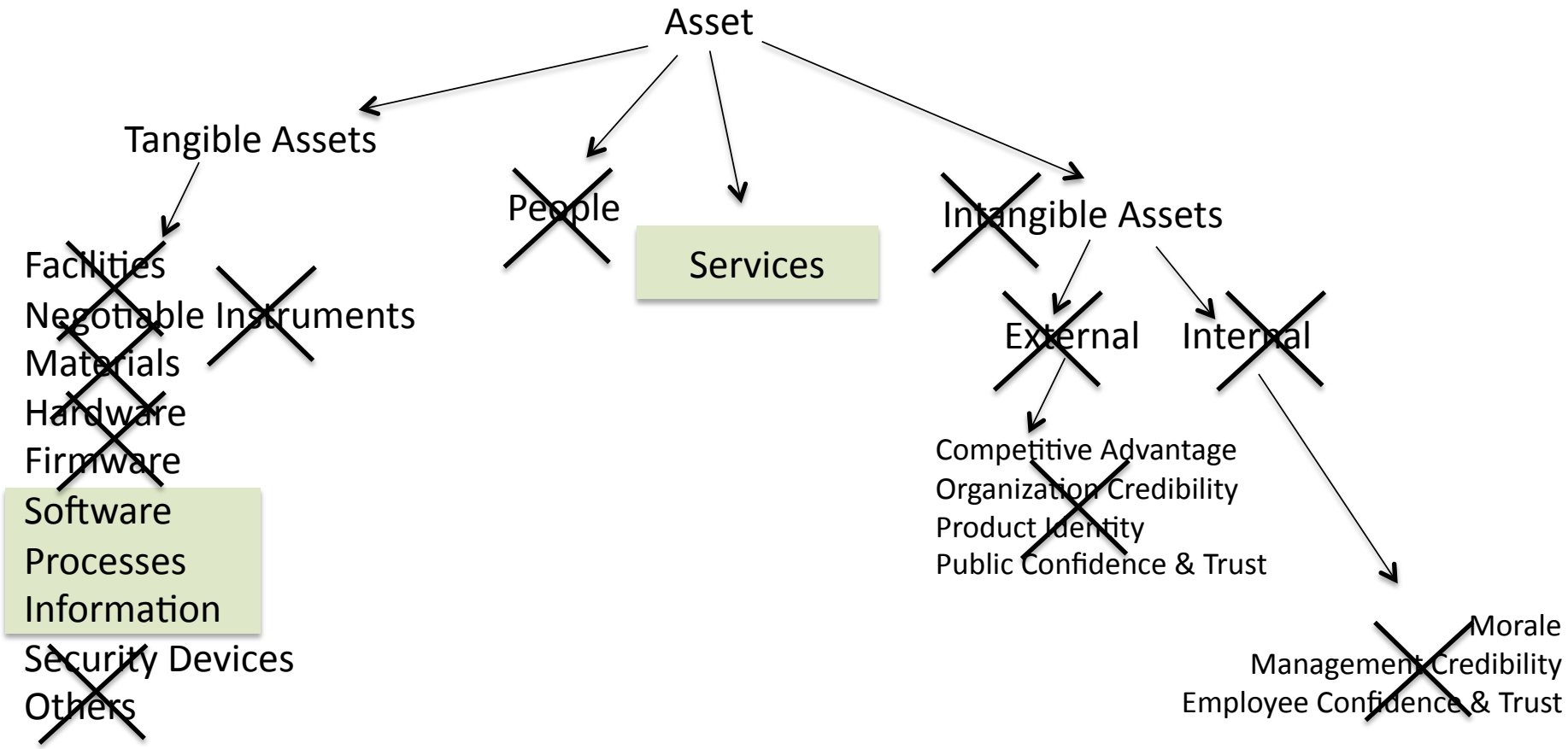


# *Taxonomy of Assets*




*Based on HTRA*

# *Taxonomy of Assets in scope*






# *Analysis of Assets is based on the following model*

*Primary asset is usually an intangible asset (information or service) provided by the enterprise to its environment and therefore is a responsibility of a stakeholder*

Assets Taxonomy

*Part of*

System Assets Taxonomy

*Based on*

Asset

Operational Capability

*involves*

*Is a*

*Is a*

*Based on*

**Primary asset**

**System asset**

*owns*

*Depends on*

*CA1 - Network packets*
*CA2 - Wireshark node*
*CA3 – Network*
*CA4 – Other information assets*
*CA5 – Knowledge of the capture*

*System asset is a point of attack or a point of failure*

*Values the*

*Is c*

Stakeholde

*volves injury*

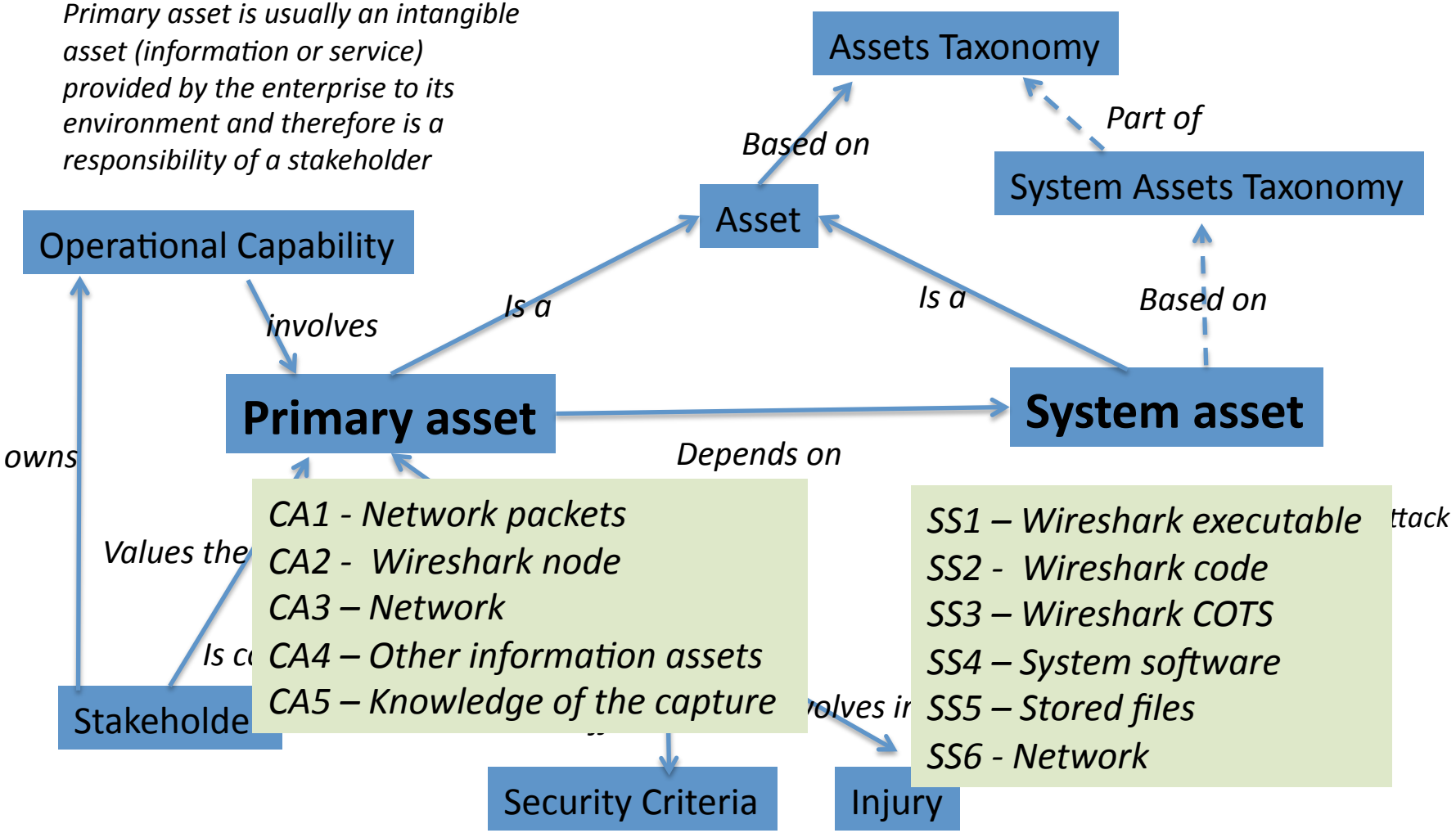
Security Criteria

Injury



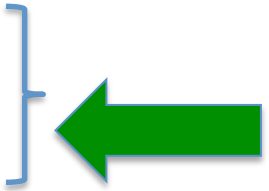
# *Analysis of Assets is based on the following model*

*Primary asset is usually an intangible asset (information or service) provided by the enterprise to its environment and therefore is a responsibility of a stakeholder*

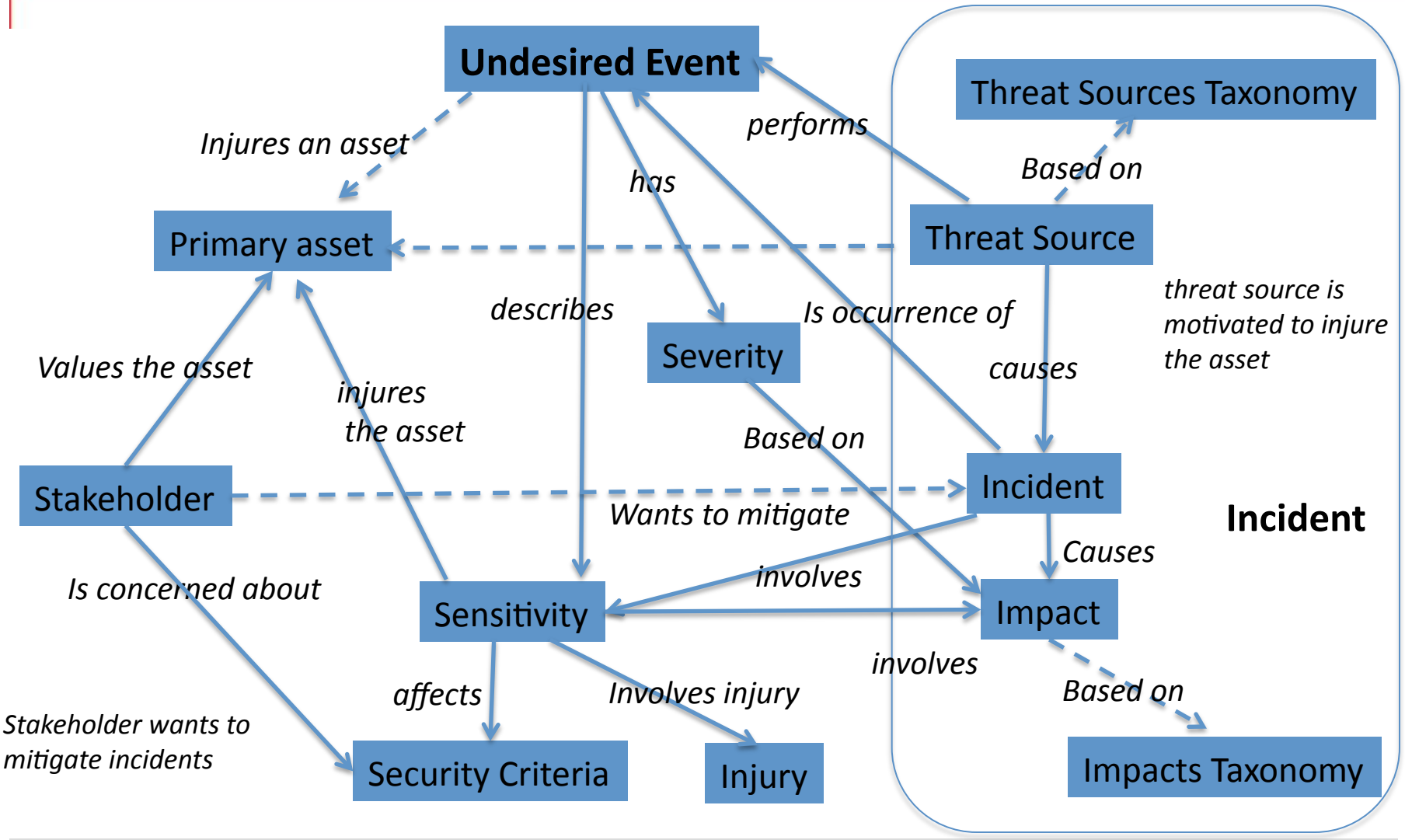


## FORSA STEPS

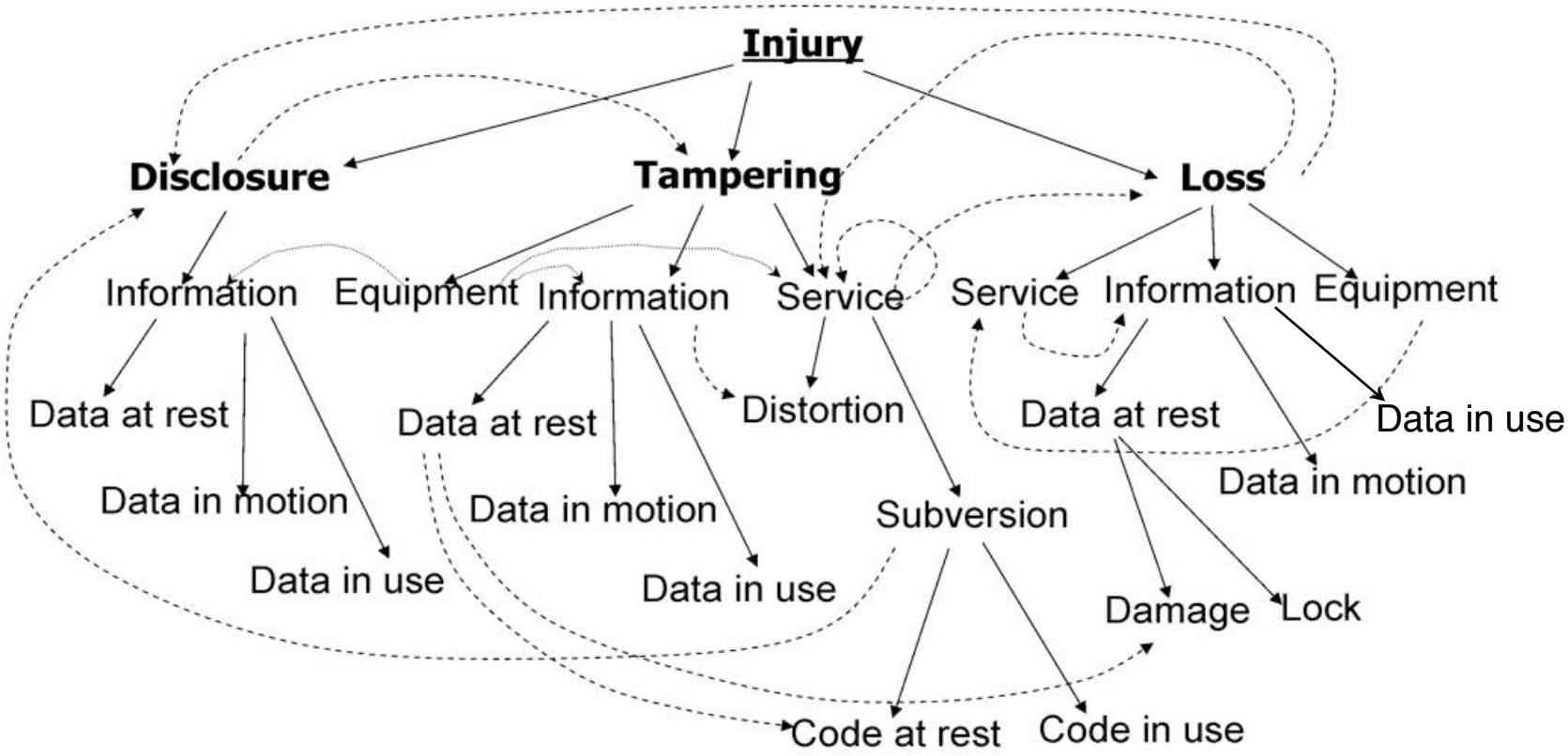
1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



# *Analysis of Undesired Events is based on the following model*

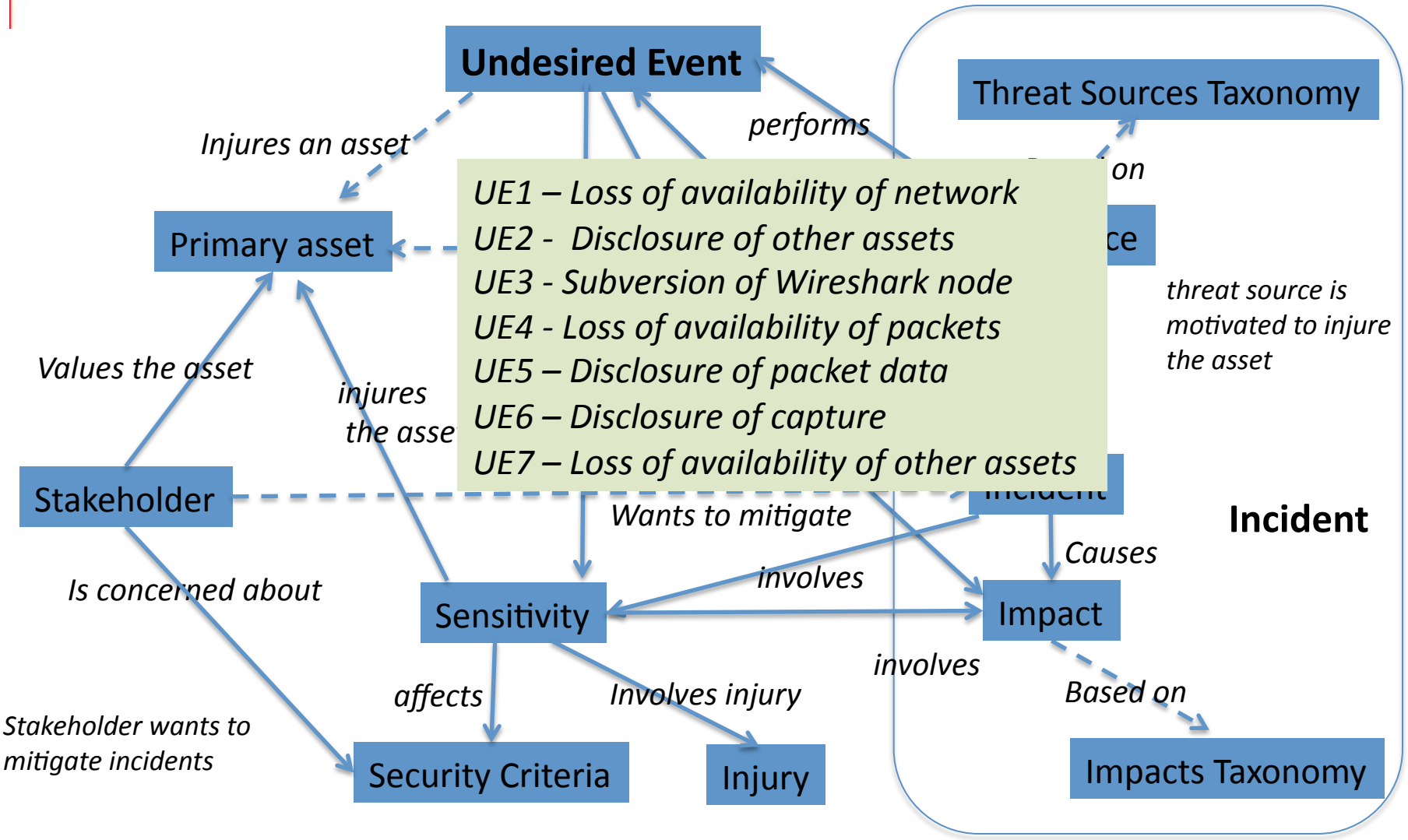


# *Taxonomy of injuries for cyber security*






# *Analysis of Undesired Events is based on the following model*



# *FORSA: The Effect analysis*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis

Mostly import (if possible) or interviews

+ validation & verification

+ validation & verification

At this point we can justify that we understand ALL classes of Undesired Events that can occur, and justify their severity



# *PART 3: THE CAUSE ANALYSIS*



## *Cause and effect in a risk statement*

**What?** Loss of confidentiality *of* confidential assets

Security criterion

**Undesired event**

Primary asset: sensitivity subject & attack goal

Threat source

Injury

R01 Hacker *gains access to* confidential assets *by* information gathering *on* stored files

**Threat scenario**

**How?** information gathering *on* stored files

Attack mode

System asset: attack target



# *The corresponding FORSA steps*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis

What?

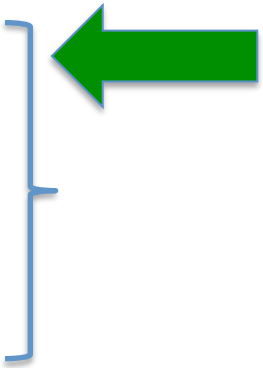
*Analysis of valuables, sensitivities and Impacts*
*=>* **severity**

How?

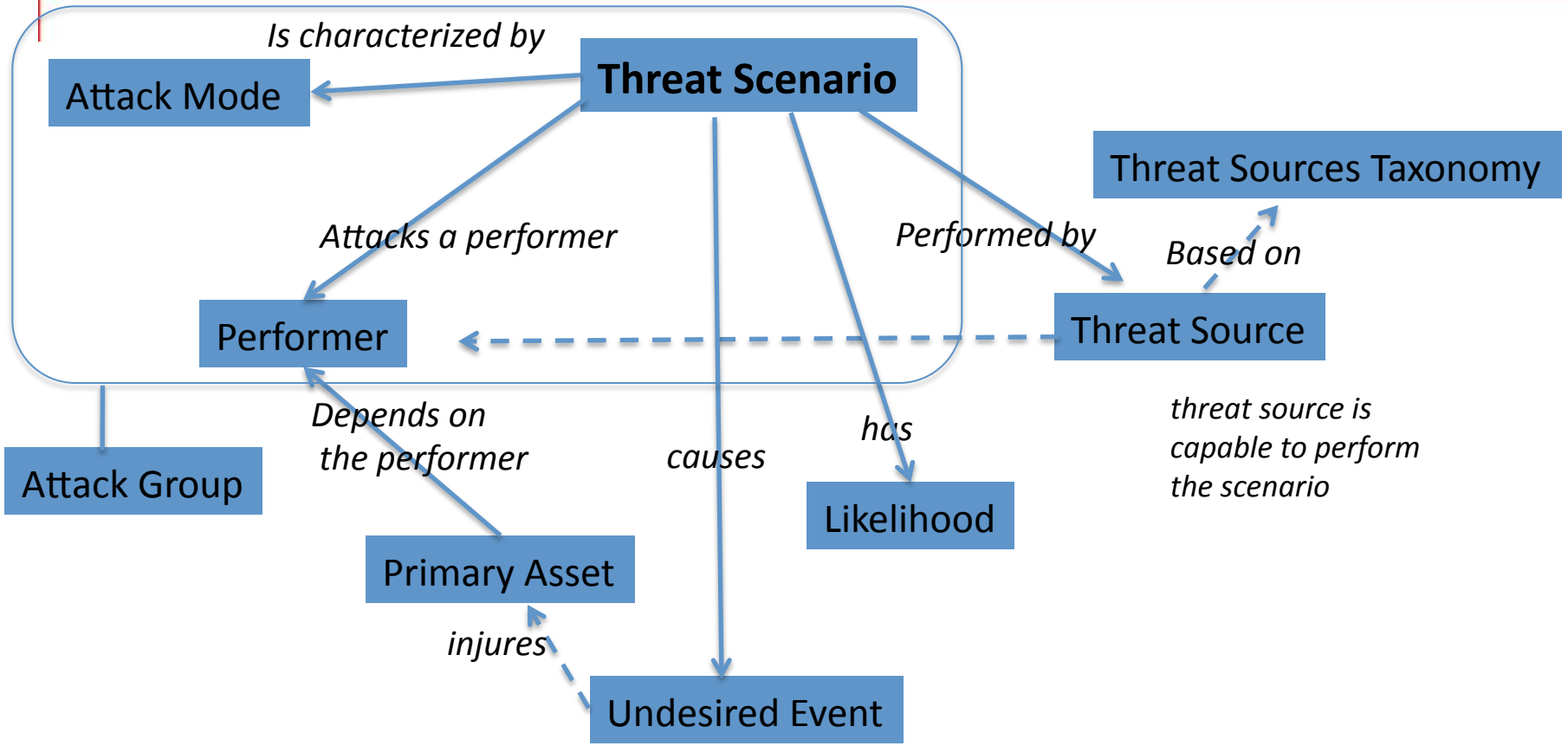
*Analysis of targets, Entries and attacks*
*=>* **likelihood**

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



# *Analysis of Attack Groups is based on the following model*




Attack Group considers a single entry point; This is a class of attacks
Threat Scenarios within the Attack Group describe various multi-stage attacks;
In a simple case Attack Group involves a single Threat Scenario





**Attack modes**

- Abuse
- Exceeding capacity
- Damage
- Loss
- Modification
- Information Gathering

**Attack Groups**

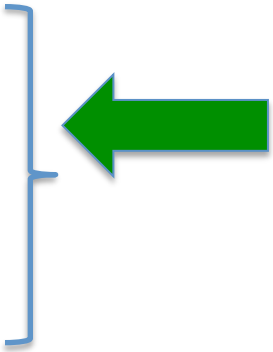
- Abuse of Wireshark node
- Damage to the Wireshark node
- Exceeding capacity of the Wireshark Node
- Modification of the Wireshark Code
- Information Gathering on Wireshark Code
- Information Gathering on Stored Files
- …

Note: Attack Groups can be systematically enumerated as a product of attack modes and systems (performers, entry points);
this is a controlled brainstorming approach

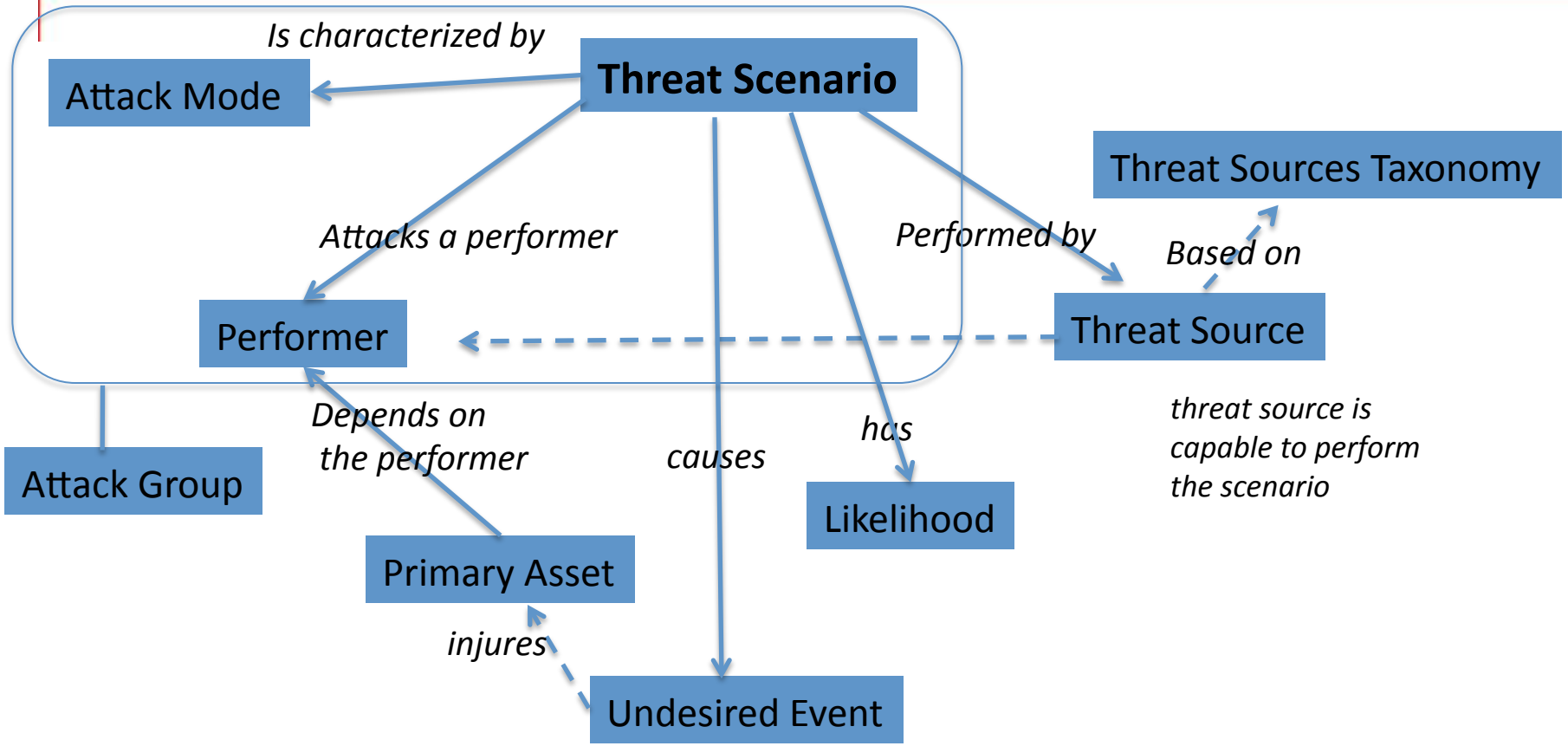


## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



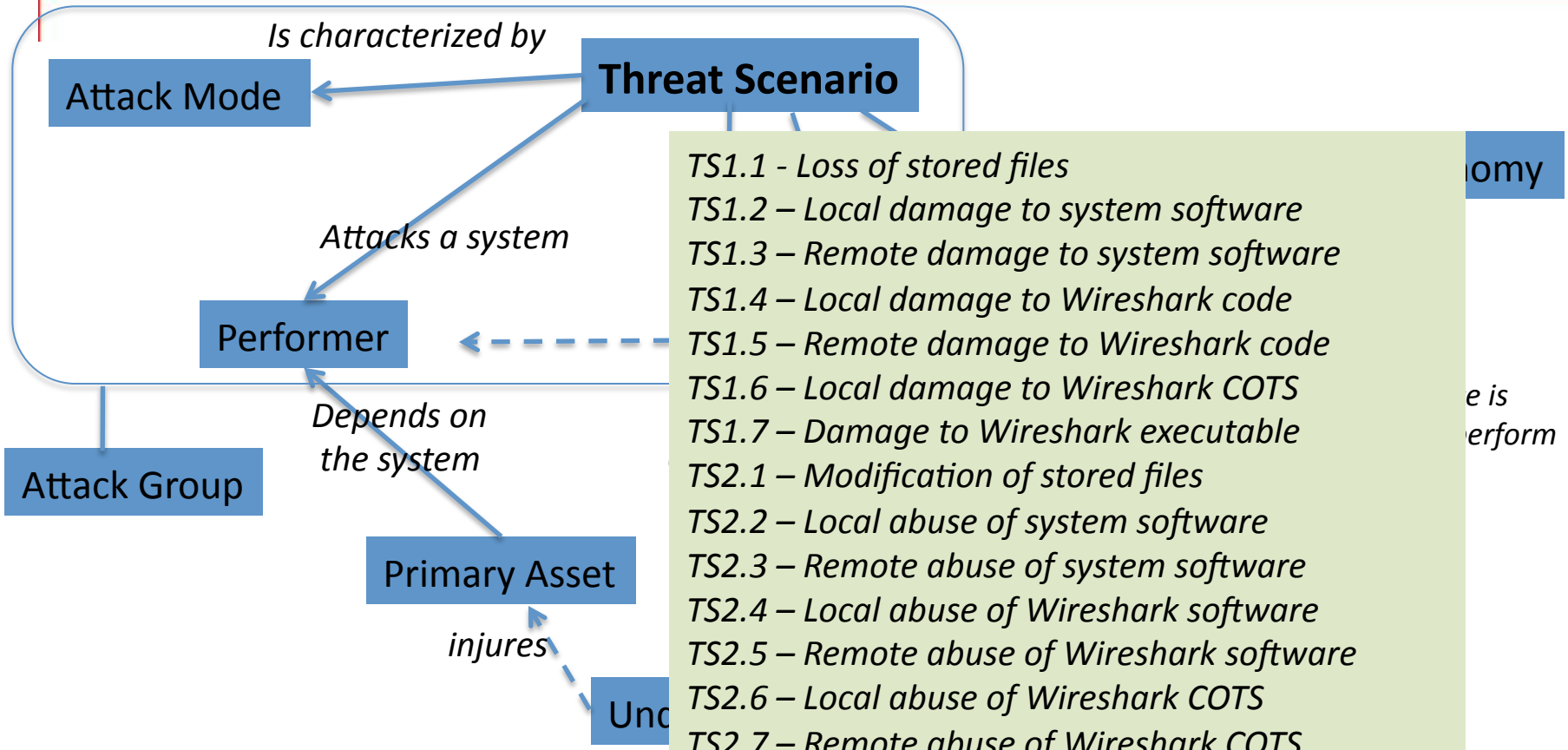
# *Analysis of Threat Scenarios is based on the following model*




Attack Group considers a single entry point; This is a class of attacks.
Threat Scenarios within the Attack Group describe various multi-stage attacks;
Analysis of Threat Scenarios is about connecting to impacts ( the “How” to the “What”)



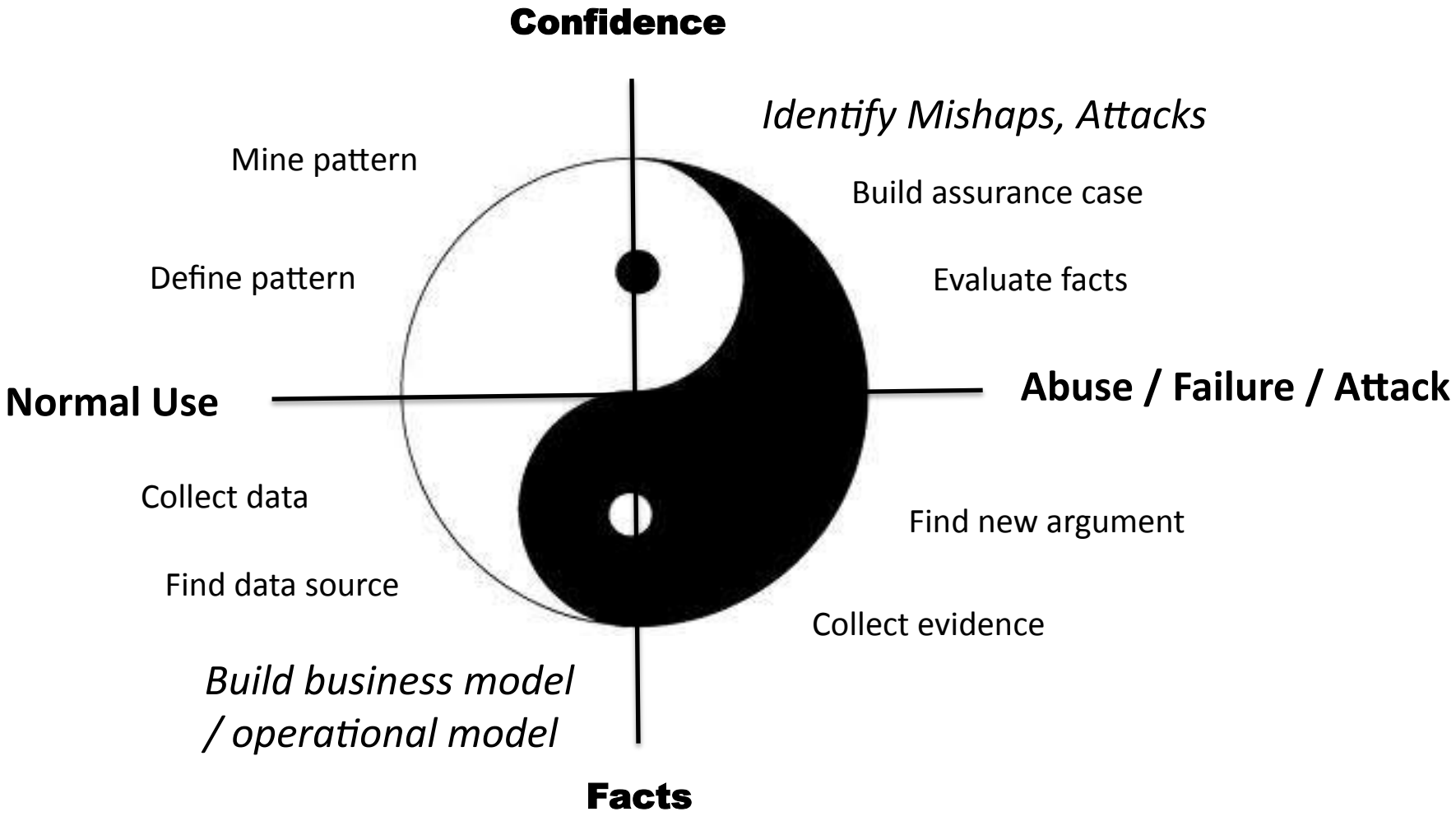
# *Analysis of Threat Scenarios is based on the following model*




*TS1.1 - Loss of stored files*
*TS1.2 – Local damage to system software*
*TS1.3 – Remote damage to system software*
*TS1.4 – Local damage to Wireshark code*
*TS1.5 – Remote damage to Wireshark code*
*TS1.6 – Local damage to Wireshark COTS*
*TS1.7 – Damage to Wireshark executable*
*TS2.1 – Modification of stored files*
*TS2.2 – Local abuse of system software*
*TS2.3 – Remote abuse of system software*
*TS2.4 – Local abuse of Wireshark software*
*TS2.5 – Remote abuse of Wireshark software*
*TS2.6 – Local abuse of Wireshark COTS*
*TS2.7 – Remote abuse of Wireshark COTS*
*TS2.8 – Modification of Wireshark executable*
*TS3.1 – Information gathering on Wireshark code*
*TS3.2 – Information gathering on stored files*



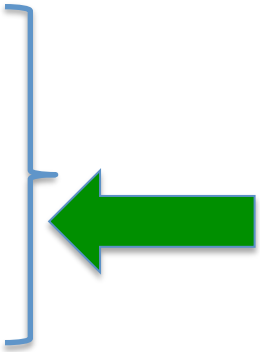
# *The Yin and Yang of Security Assurance*



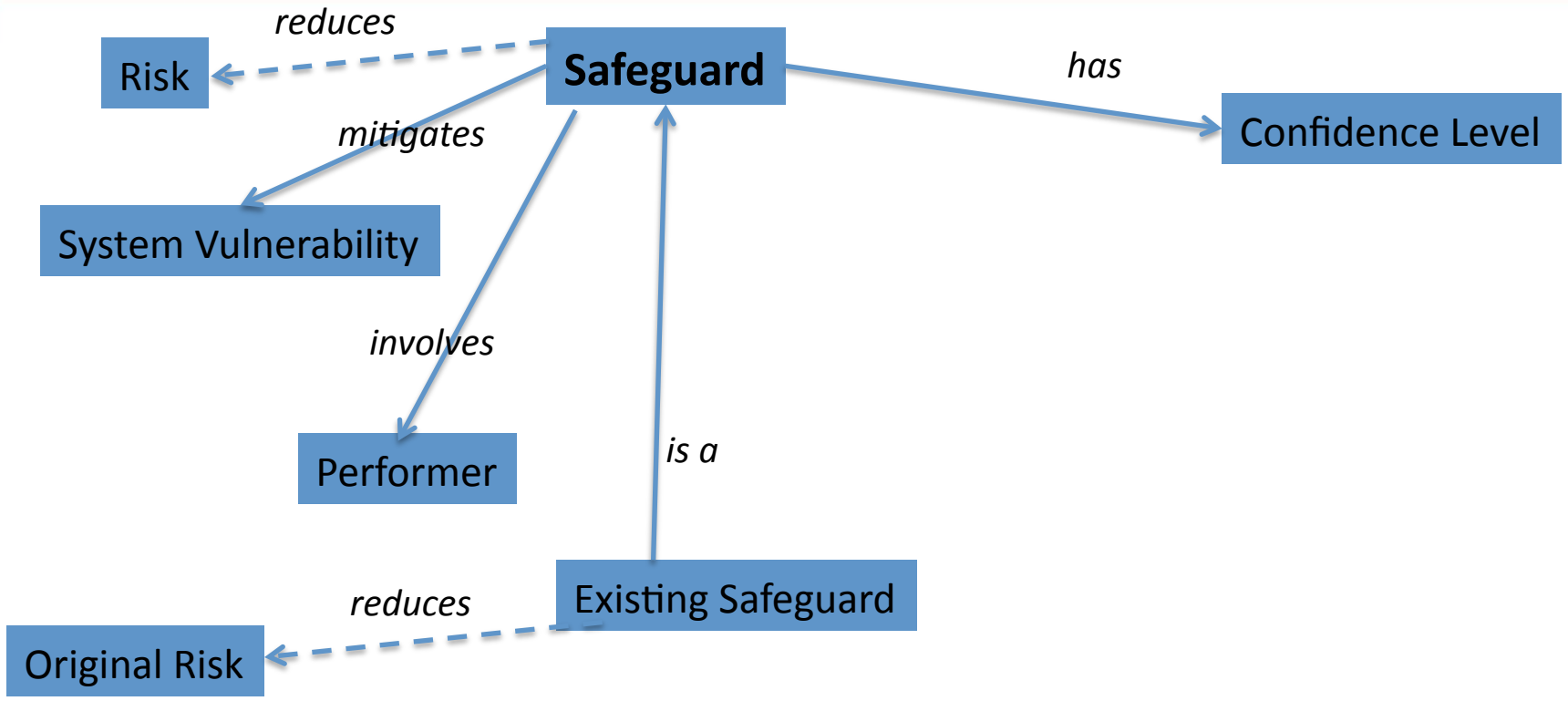



## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



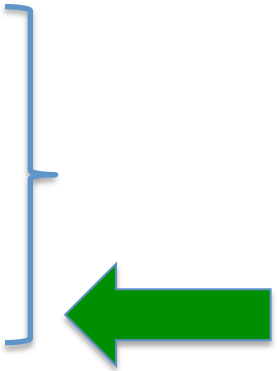
# *Analysis of Safeguards is based on the following model*



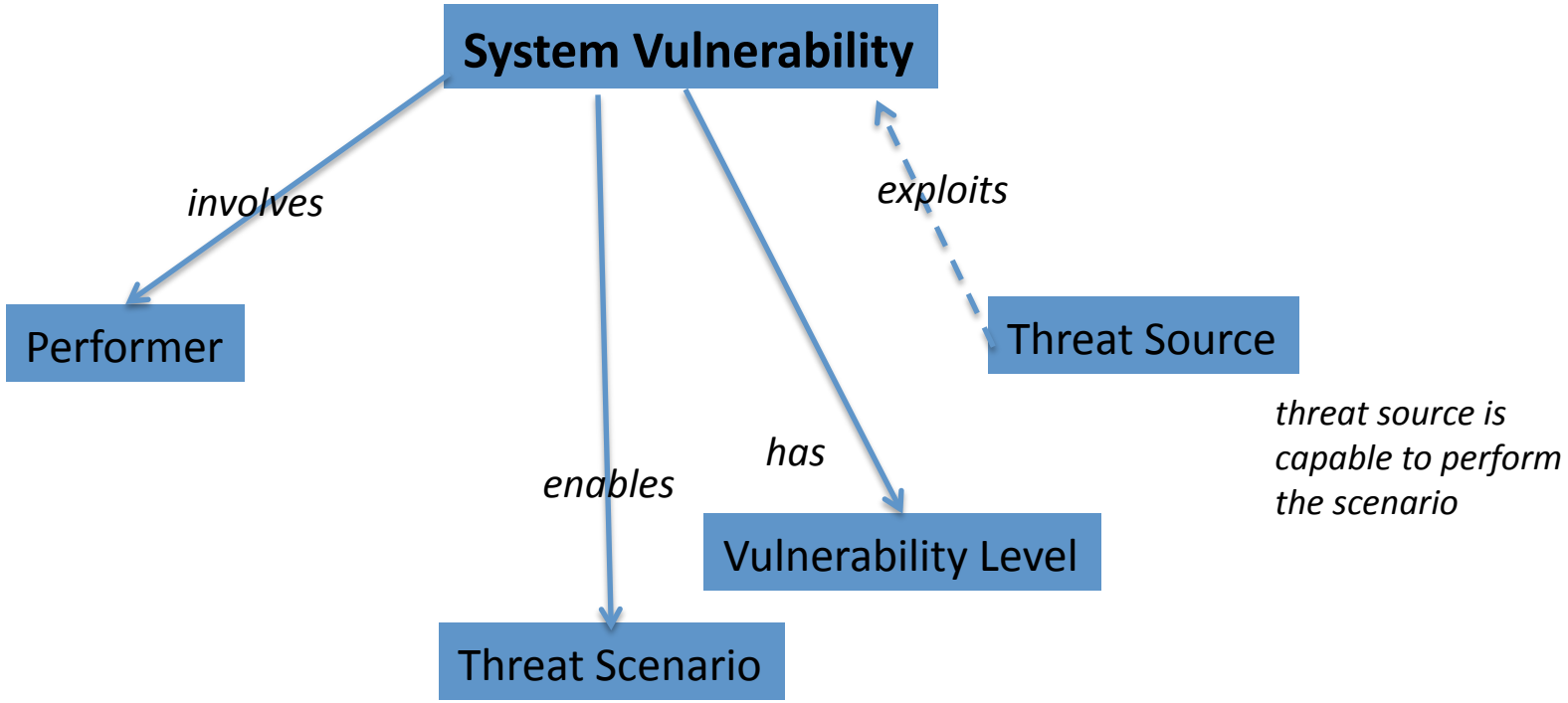

Threat Scenario within an Attack Group describes a multi-stage attack that involves multiple Performers;
A Safeguard prevents an attack from succeeding;
A System Vulnerability is a condition involving a certain Performer that
enables the attack and permit it to propagate and eventually cause an undesired event

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



# *Analysis of System Vulnerabilities is based on the following model*




Threat Scenario within an Attack Group describes a multi-stage attack that involves multiple Performers;
A System Vulnerability is a condition involving a certain Performer that enables the attack and permit it to propagate and eventually cause an undesired event



# *FORSA: The Cause analysis*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis

At this point we can justify that we understand ALL classes of attacks on the system and justify their likelihood; and justify the level of vulnerability

+validation & verification

+validation & verification

+validation & verification

+validation & verification



# *PART 4: RISK ANALYSIS & MITIGATION*

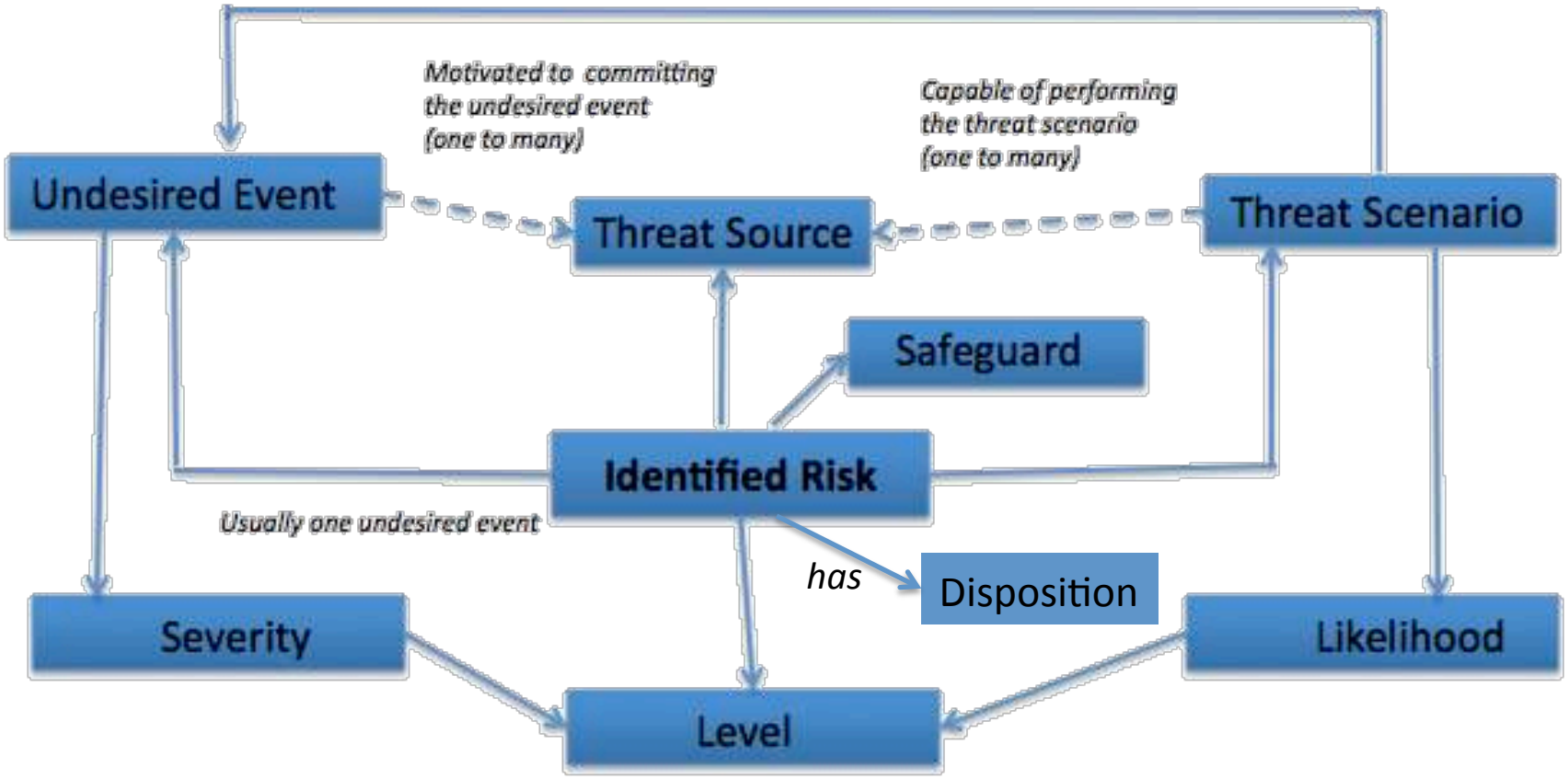


## FORSA STEPS

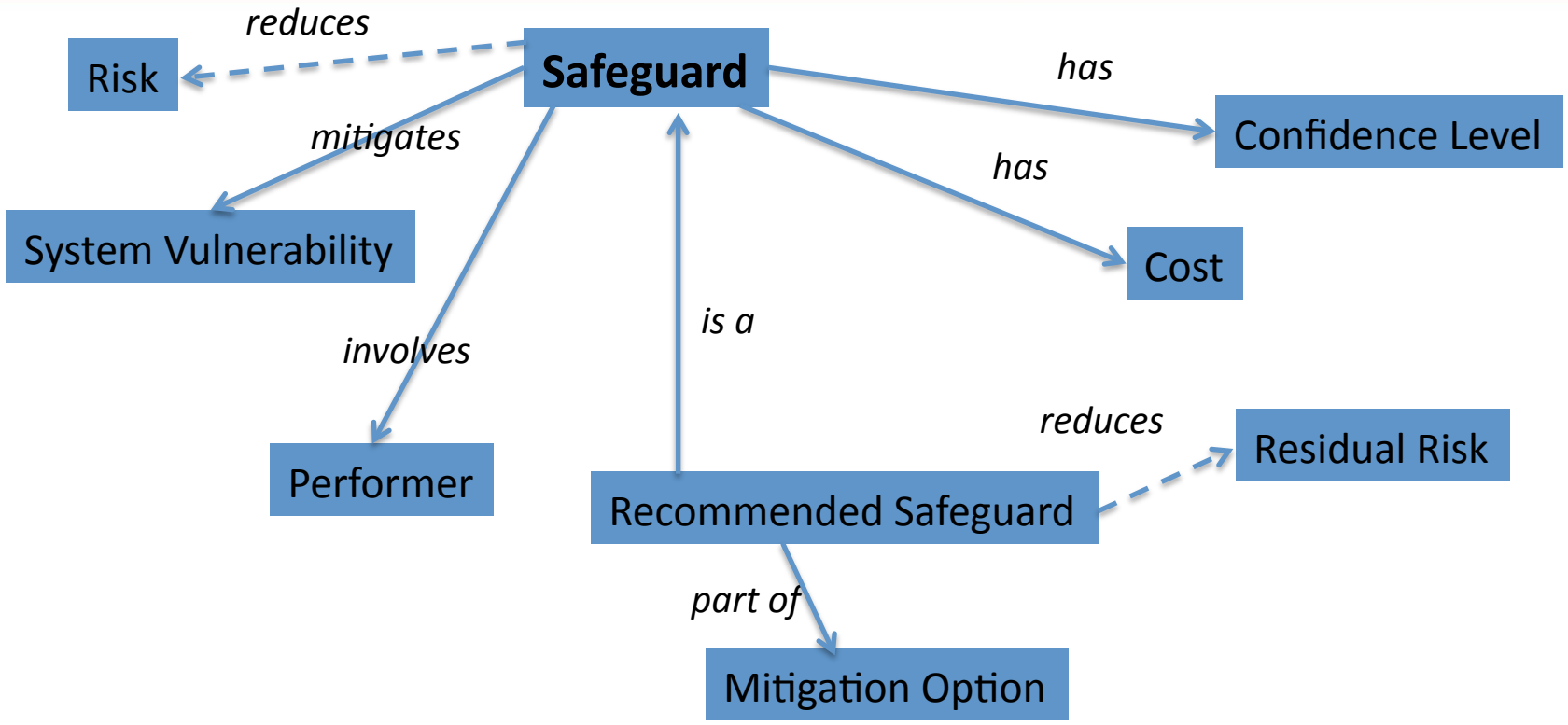
1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis

Analytics, recommendations, Mitigation option analysis

# *Analysis of Risks is based on the following model*



# *Risk Mitigation Analysis is based on the following model*




Threat Scenario within an Attack Group describes a multi-stage attack that involves multiple Performers;
A Safeguard prevents an attack from succeeding;
A System Vulnerability is a condition involving a certain Performer that
enables the attack and permit it to propagate and eventually cause an undesired event

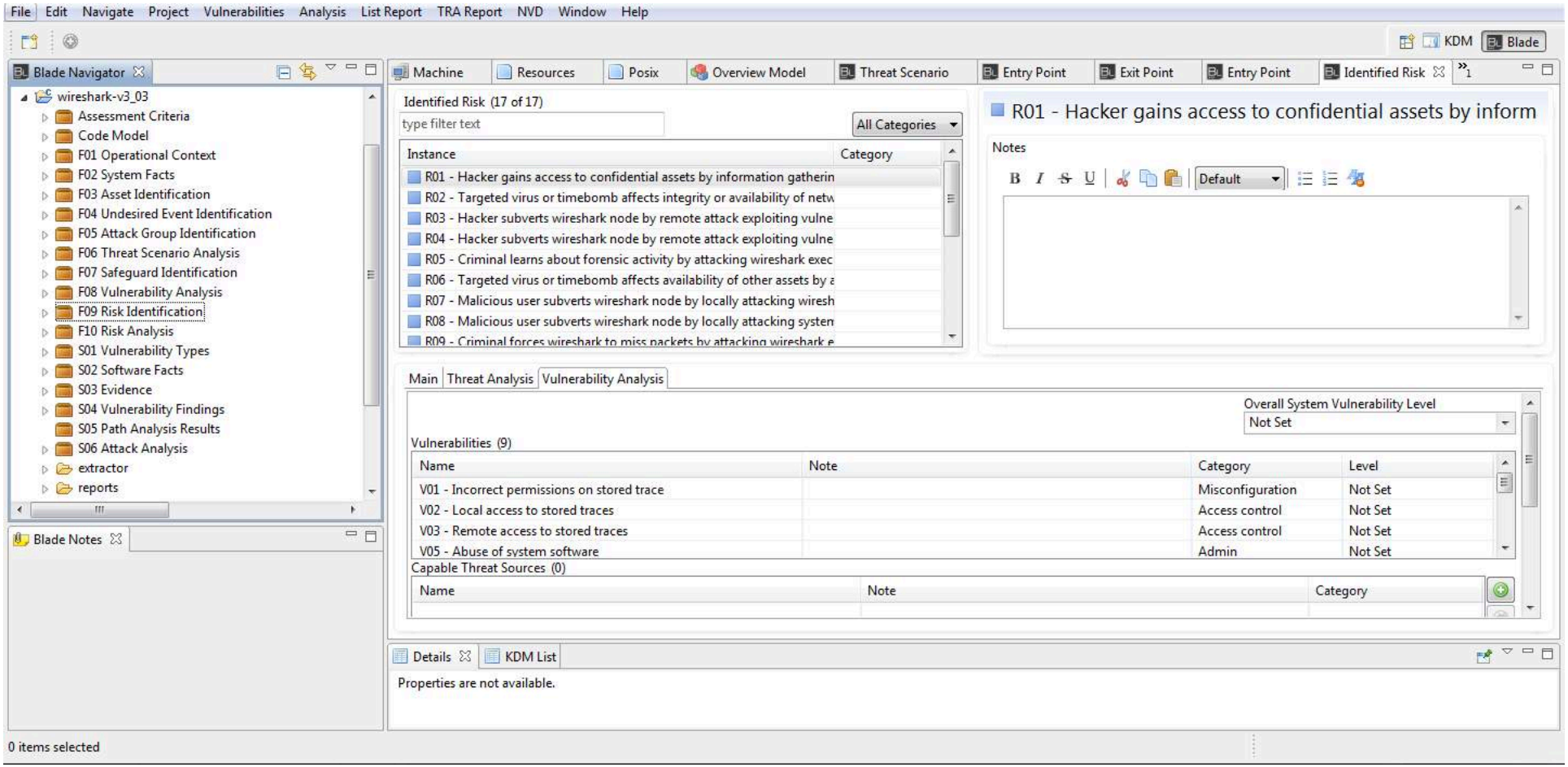



# *KDM BLADE TOOL*

*Now integrated with NoMagic tools as Cameo Risk Analyzer*



# KDM Blade GUI

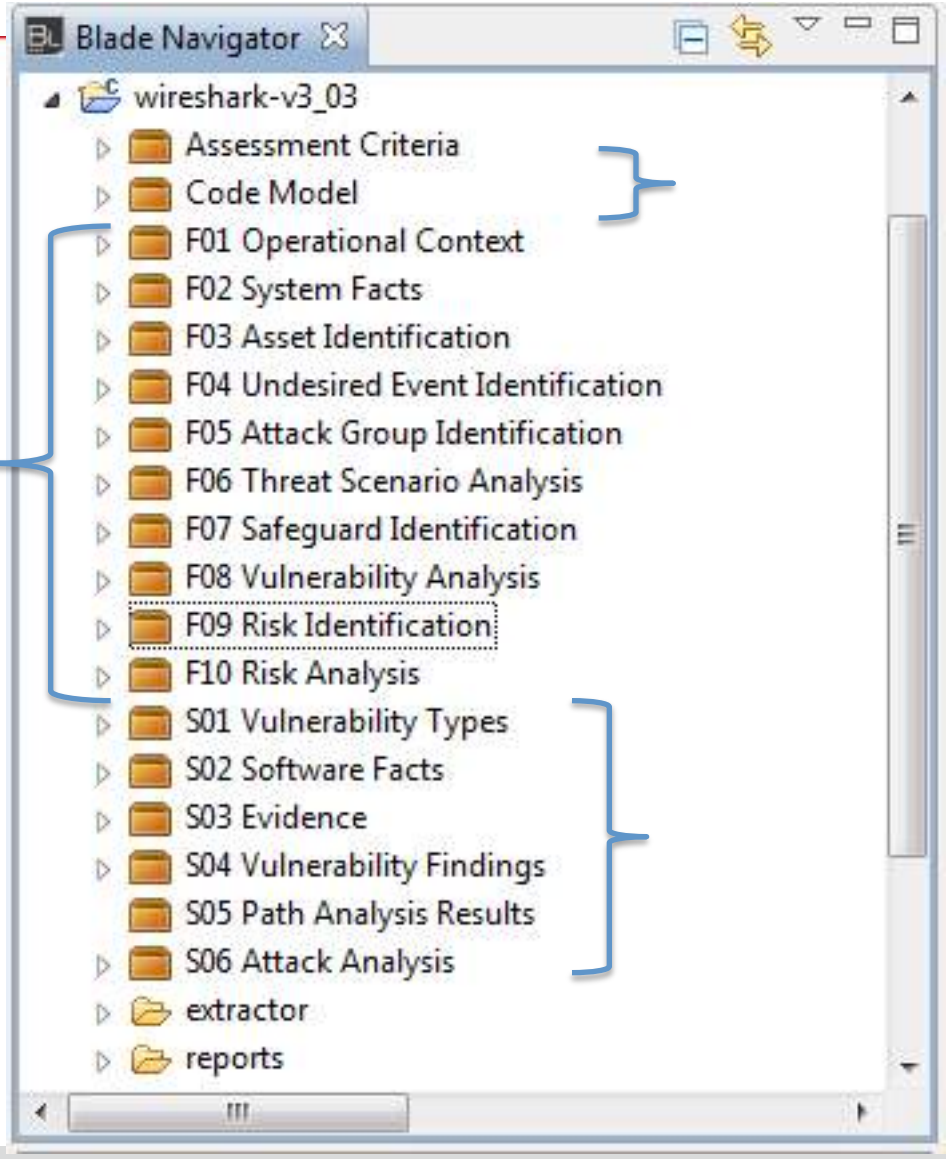





# *Blade Navigator*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis
11. Evidence Analysis

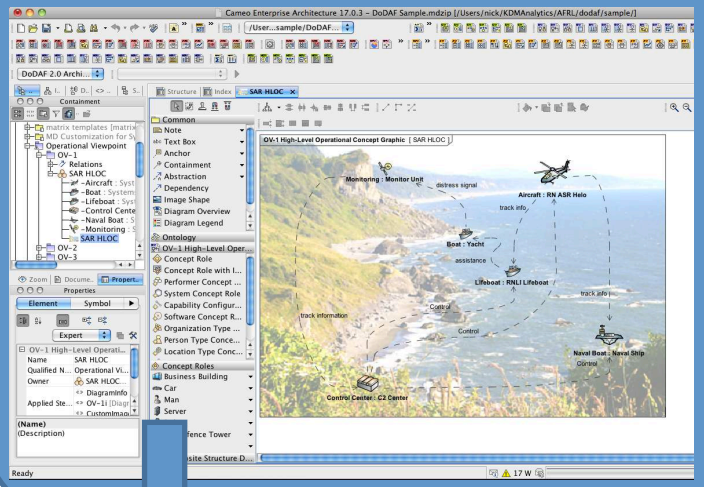





# *Sources of Facts in KDM Blade*

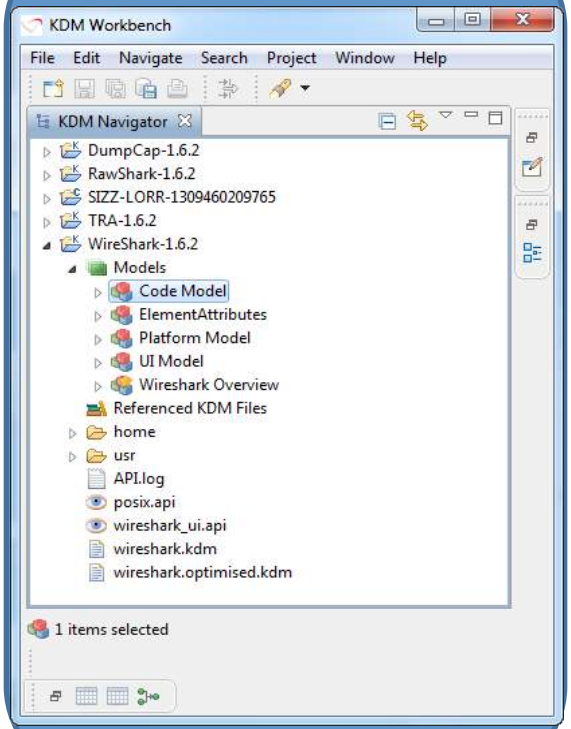
1

DoDAF tool



4

KDM Blade GUI




Manually asserted facts

2

DoDAF Analytics

3

5

Workspace A

Projects
Project A
Project B
Project C

RDF Repository

6

Automatically extracted facts



# *DoDAF Analytics: "Automated Everything"*

Full automation

## DoDAF Analytics

- Import DoDAF XML
- Collect Initial Evidence and generate overview report
- Perform DoDAF Analytics and calculate the score for the input model
- Generate Blade Facts for Operational Context
- Generate Blade System Facts
- Identify Internal Threat Sources
- Perform Asset Identification
- Perform Undesired Event Identification
- Perform Threat Scenario Identification
- Perform Threat Scenario Analysis
- Perform Risk Identification
- Perform Vulnerability Identification

Assessment of the input model

Skeleton of the full report



***QUESTIONS ?***



***JUSTIFIABLE RISK ASSURANCE = RISK MANAGEMENT + ASSURANCE CASE***



# *Justifiable Risk Assurance*

Assurance Case
is related to the level of risk

Assurance Case
is structured according
to Risk Metamodel

Assurance Case

Risk Metamodel

Assurance Process delivers
assurance case
(argument + evidence)

Assurance Case
provides guidance
on how to collect evidence

Assurance Process

Risk Metamodel provides guidance
on how to collect evidence

Assurance Process — Generic guidance & tool

Report — Data specific to system



# *Justifiable Risk Assurance*

Assurance Case
is related to the level of risk

**System Model**

e.g. DoDAF views

Assurance Case
is structured according to
System Model

Assurance Case
is structured according
to Risk Metamodel

System Model
constrains evidence

Risk Metamodel
describes evidence

**Report / Risk Manage-ment Plan**

Assurance Case

**Evidence**

Risk Metamodel

Assurance Case
structures
report

Evidence justifies claims

Assurance Process delivers
assurance case
(argument + evidence)

Assurance Process defines
the steps to collect
evidence

Assurance Case
provides guidance
on how to collect evidence

Assurance Process

Risk Metamodel provides guidance
on how to collect evidence

Assurance Process — Generic guidance & tool

Report — Data specific to system



# *The key outcome is a justifiable list of risks*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis

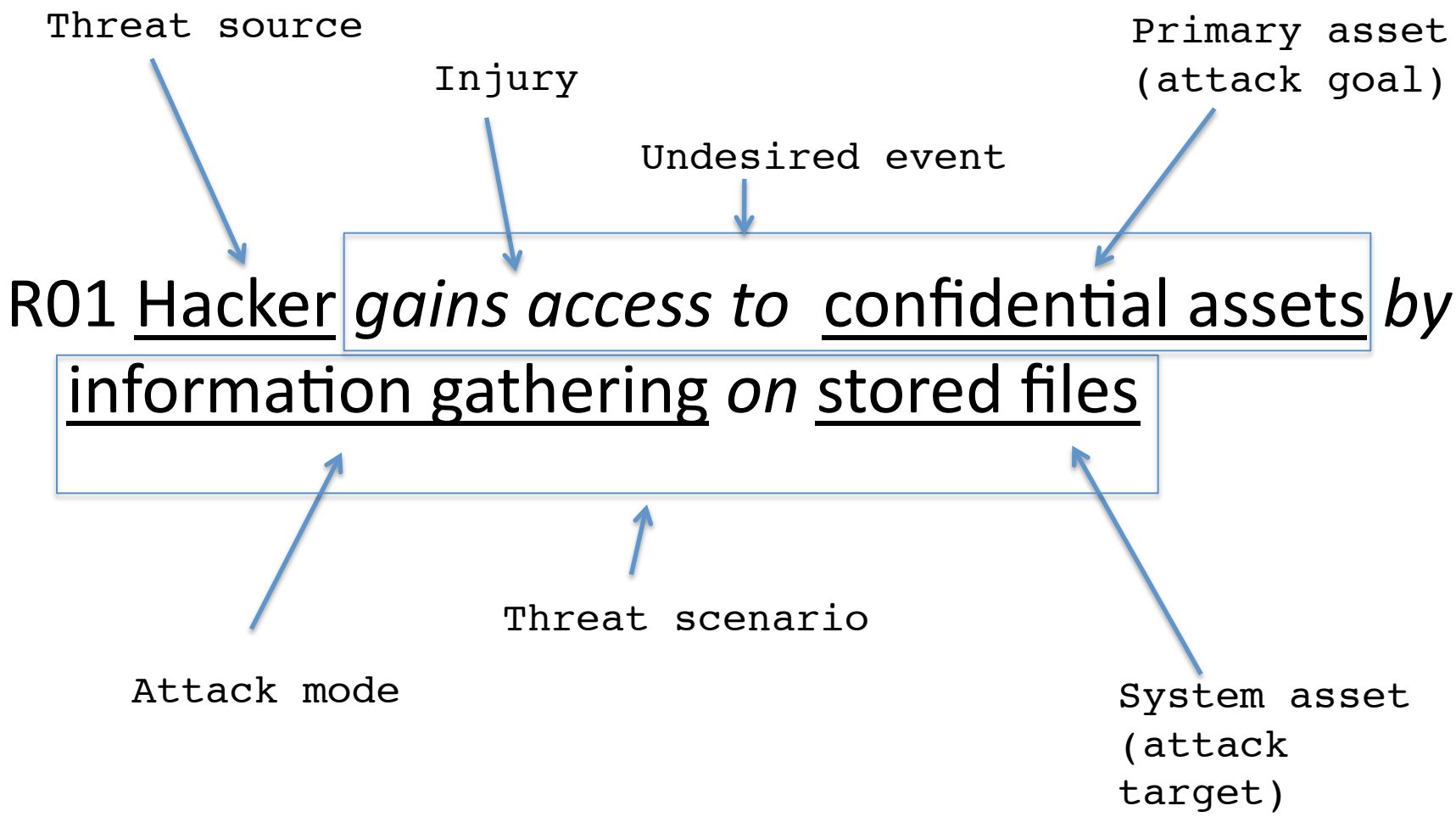
Steps leading to systematic identification of risks

# *Results of justifiable risk analysis (1 of 2)*

| ID | Description | Severity | Likeli-hood | Level | Resi-dual | Confi-dence |
|---|---|---|---|---|---|---|
| R01 | Hacker gains access to confidential assets by information gathering on stored files | high | high | high | low | 80% |
| R02 | Targeted virus or timebomb affects integrity or availability of network by attacking wireshark executable | high | high | high | low | 80% |
| R03 | Hacker subverts wireshark node by remote attack exploiting vulnerabilities in Wireshark code | high | medium | high | low | 80% |
| R04 | Hacker subverts wireshark node by remote attack exploiting vulnerabilities in system software on wireshark node | high | medium | medium | low | 80% |
| R05 | Criminal leans about forensic activity by attacking wireshark executable | medium | high | medium | low | 80% |
| R06 | Targeted virus or timebomb affects availability of other assets by attacking wireshark executable | medium | high | medium | low | 80% |
| R07 | Malicious user subverts wireshark node by locally attacking wireshark code | medium | low | medium | low | 70% |
| R08 | Malicious user subverts wireshark node by locally attacking system software on wireshark node | medium | low | medium | low | 70% |
| R09 | Criminal forces wireshark to miss packets by attacking wireshark executable | medium | high | medium | low | 70% |



# *Anatomy of a risk statement*



## *Cause and effect in a risk statement*

**What?** Loss of confidentiality *of* confidential assets

Security criterion

**Undesired event**

Primary asset: sensitivity subject & attack goal

Threat source

Injury

R01 Hacker *gains access to* confidential assets *by*

**How?** information gathering *on* stored files

Attack mode

Threat scenario

System asset (attack target)



# *The corresponding FORSA steps*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis

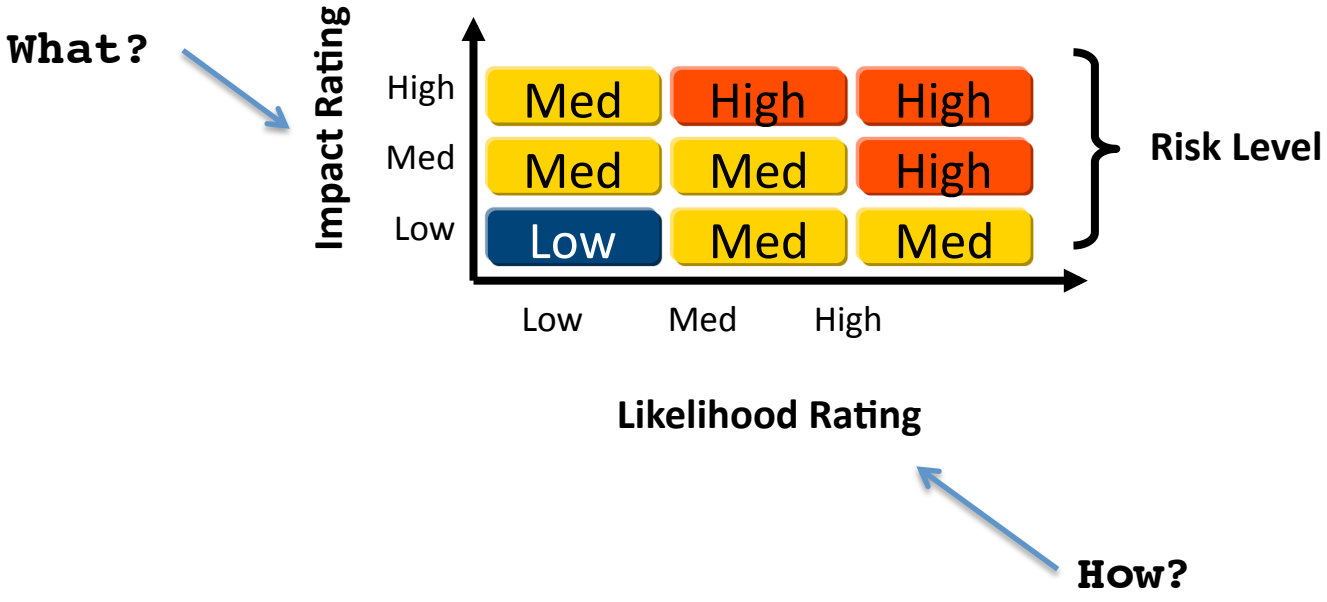
What?

*Analysis of valuables, sensitivities and Impacts*
*=>* **severity**

How?

*Analysis of targets, Entries and attacks*
*=>* **likelihood**

# *Risk analysis*




Risk= ∫(Severity, Likelihood)

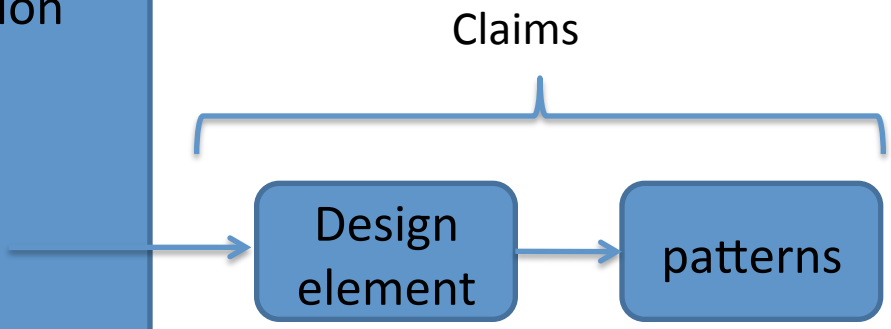
# *FROM RISK TO CLAIMS & EVIDENCE THROUGH PATTERNS*



# *Evidence collection planning*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



# *Business Models -> SysML -> Code*

Capabilities

Performers

Operational Activities

Exchange Elements

Systems

Functions

Entity Items

Subsystems

Entity Items

Modules

Modules

Modules

Files

Files

Files

Procedures

Procedures

Data Elements

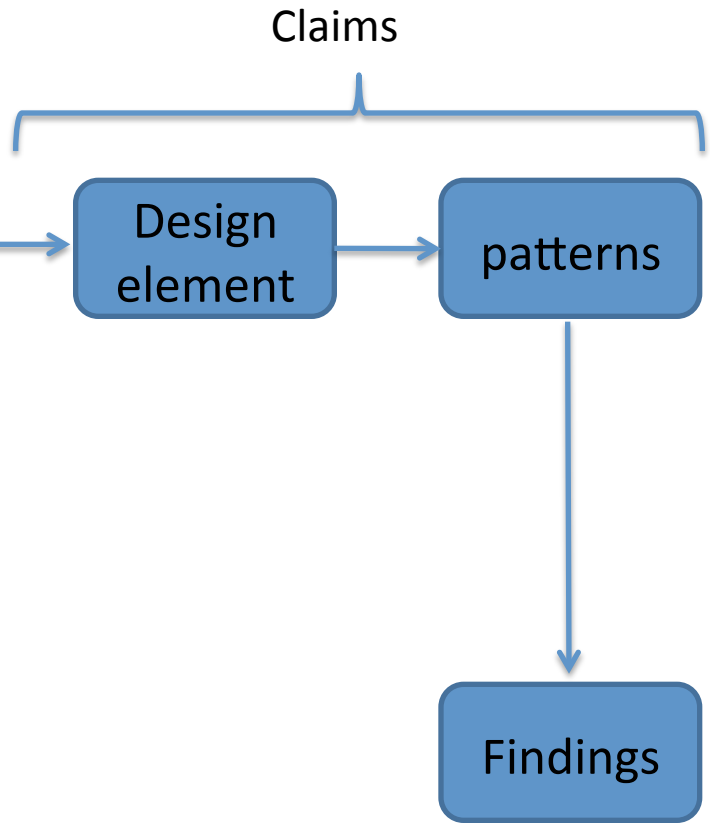
Blocks



# *Evidence collection*

## FORSA STEPS

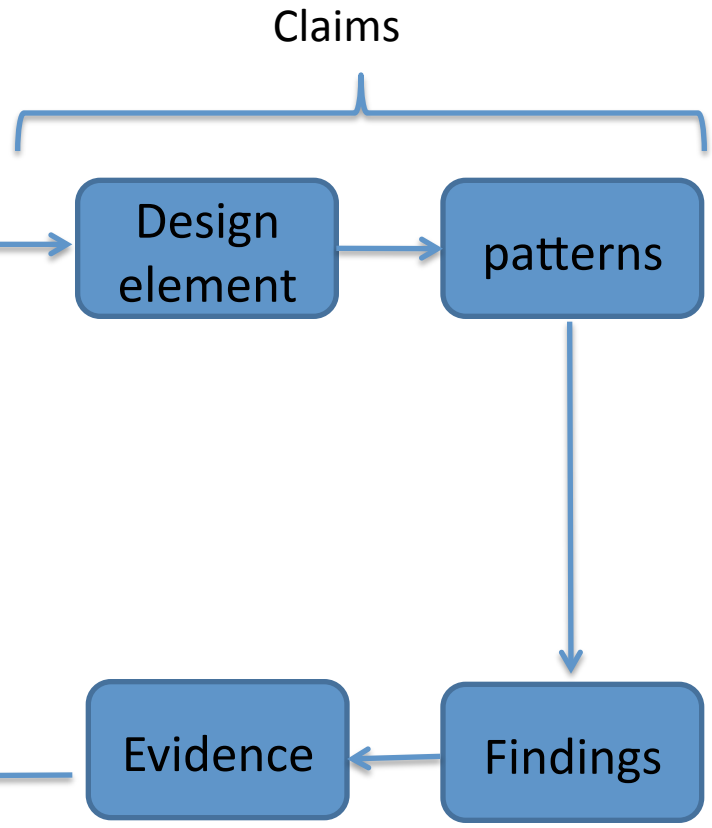
1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis



# *Evidence analysis*

## FORSA STEPS

1. Operational Context Identification
2. System Facts
3. Asset Identification
4. Undesired Event Identification
5. Attack Group Identification
6. Threat Scenario Analysis
7. Safeguard Identification
8. Vulnerability Analysis
9. Risk Identification
10. Risk Analysis
11. Evidence Analysis



## *Undesired Events -> Claims -> Patterns*

- We start with an enumeration of undesired events (something happens; an *injury*)
  - Usually the injury happens *to an asset*
  - There is a *design element* (*piece of code)* that causes the injury when executed
  - Can we identify all necessary pieces of code for the specific injury ?
- We make *claims* regarding all possible necessary conditions of an injury
- As the result we get the *patterns* to identify the corresponding design elements
- Some patterns are *common* vulnerability patterns
- Others are application-specific patterns

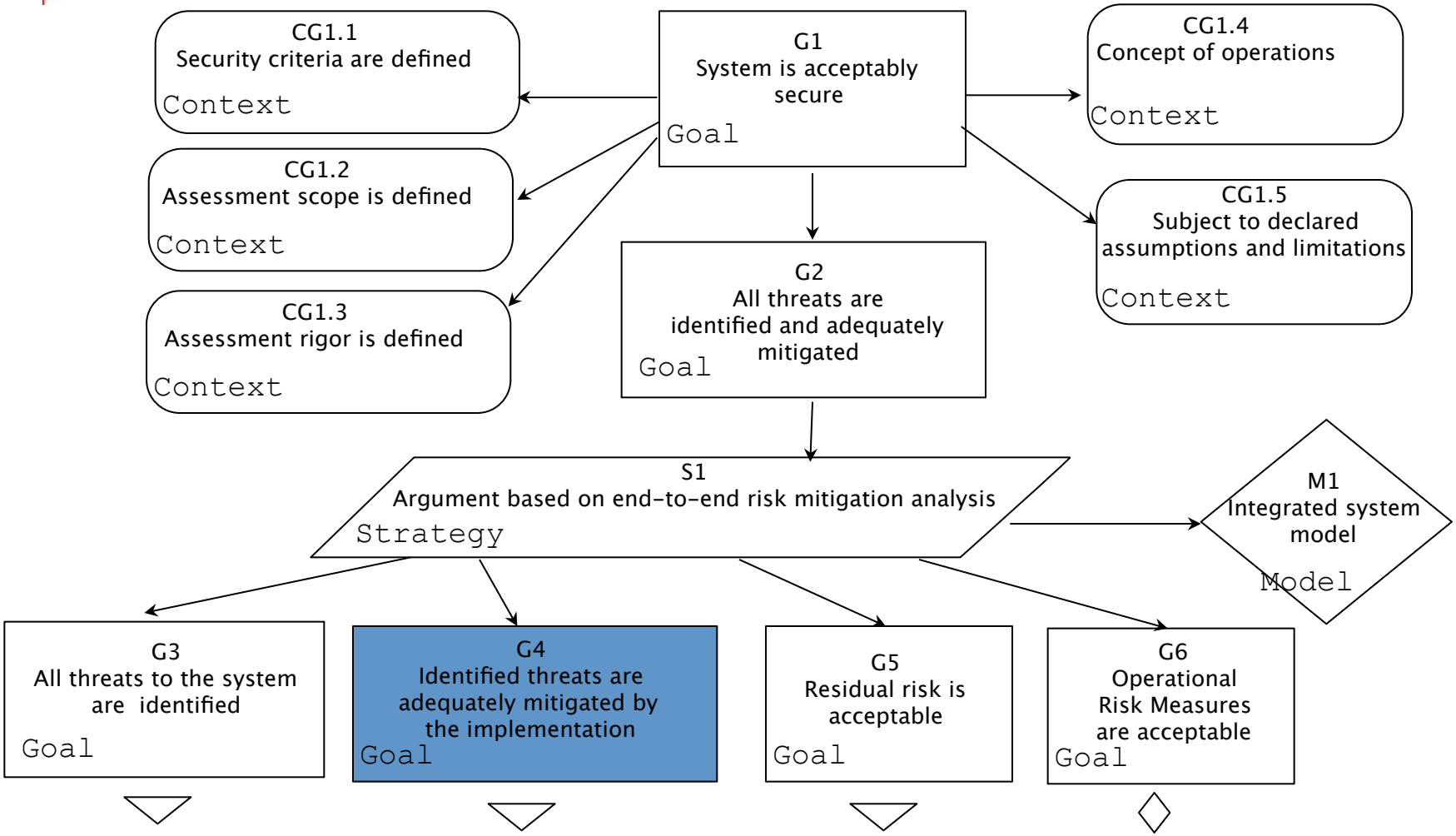


# *ASSURANCE CASE FOR JUSTIFIABLE RISK MANAGEMENT*

uses OMG Structured Assurance Case Metamodel (SACM)

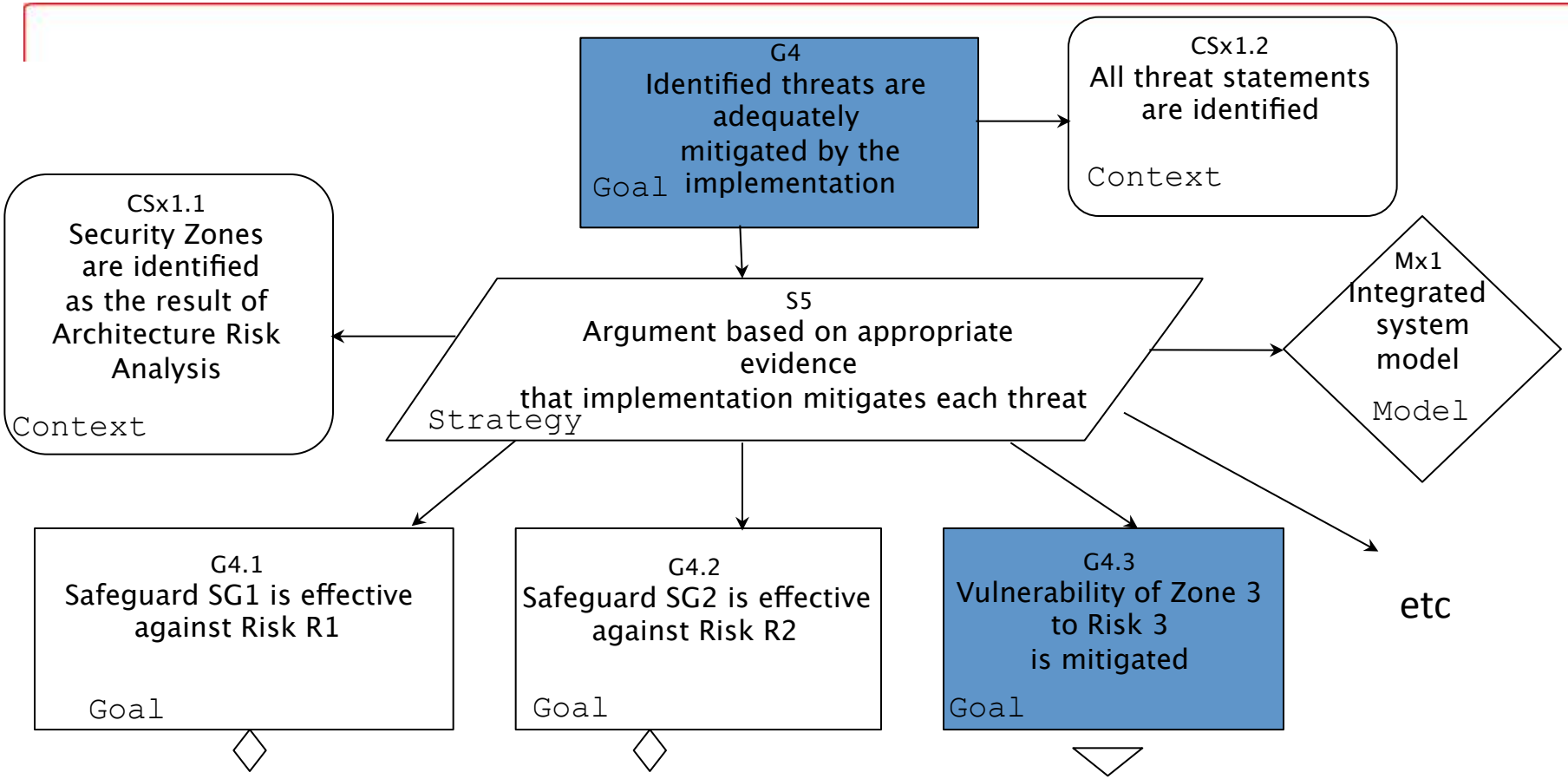


# *Security Assurance Case (top)*






# *Mitigation argument*

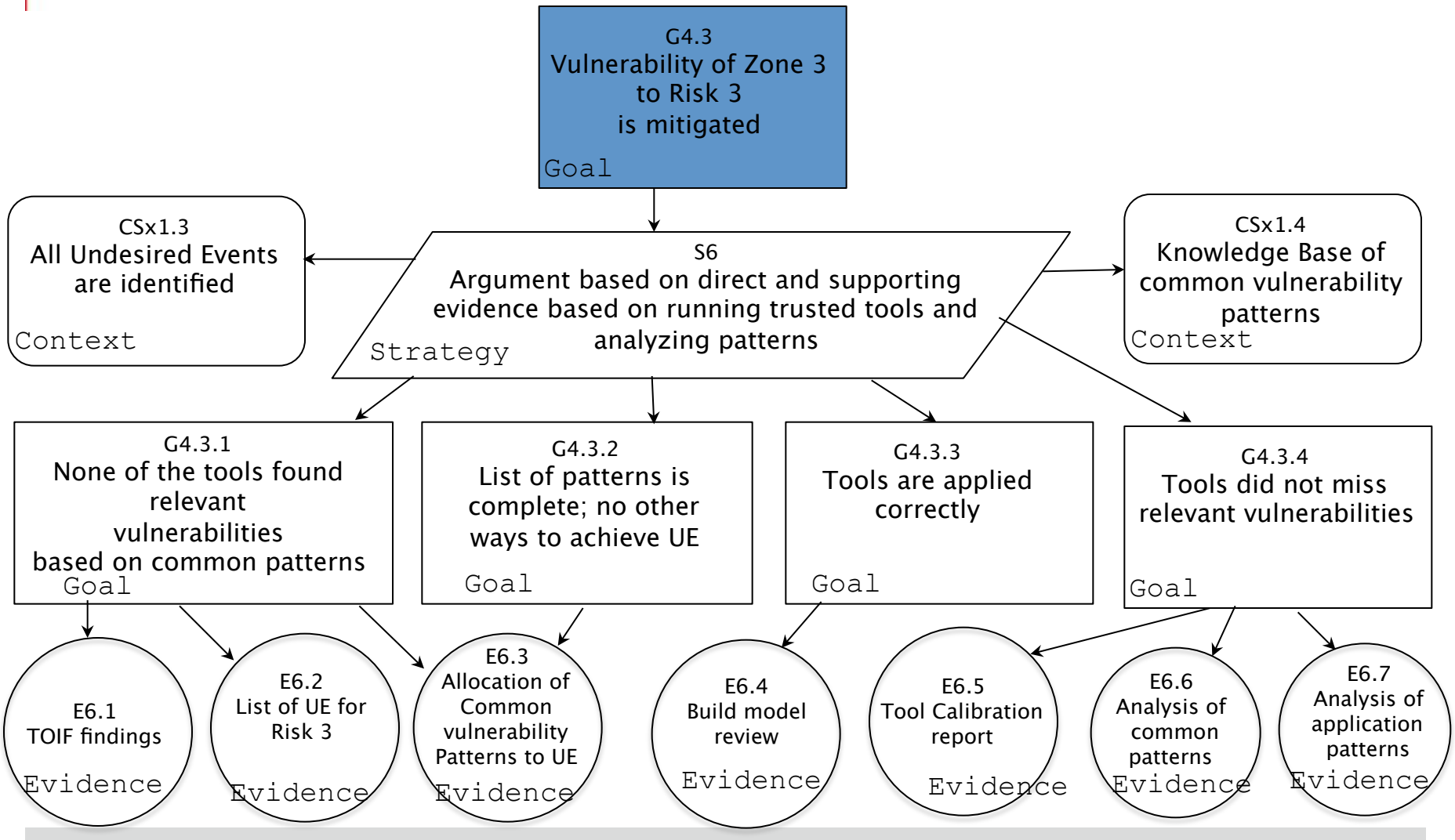



Specific argument:
Claims and evidence

Specific argument:
Claims and evidence

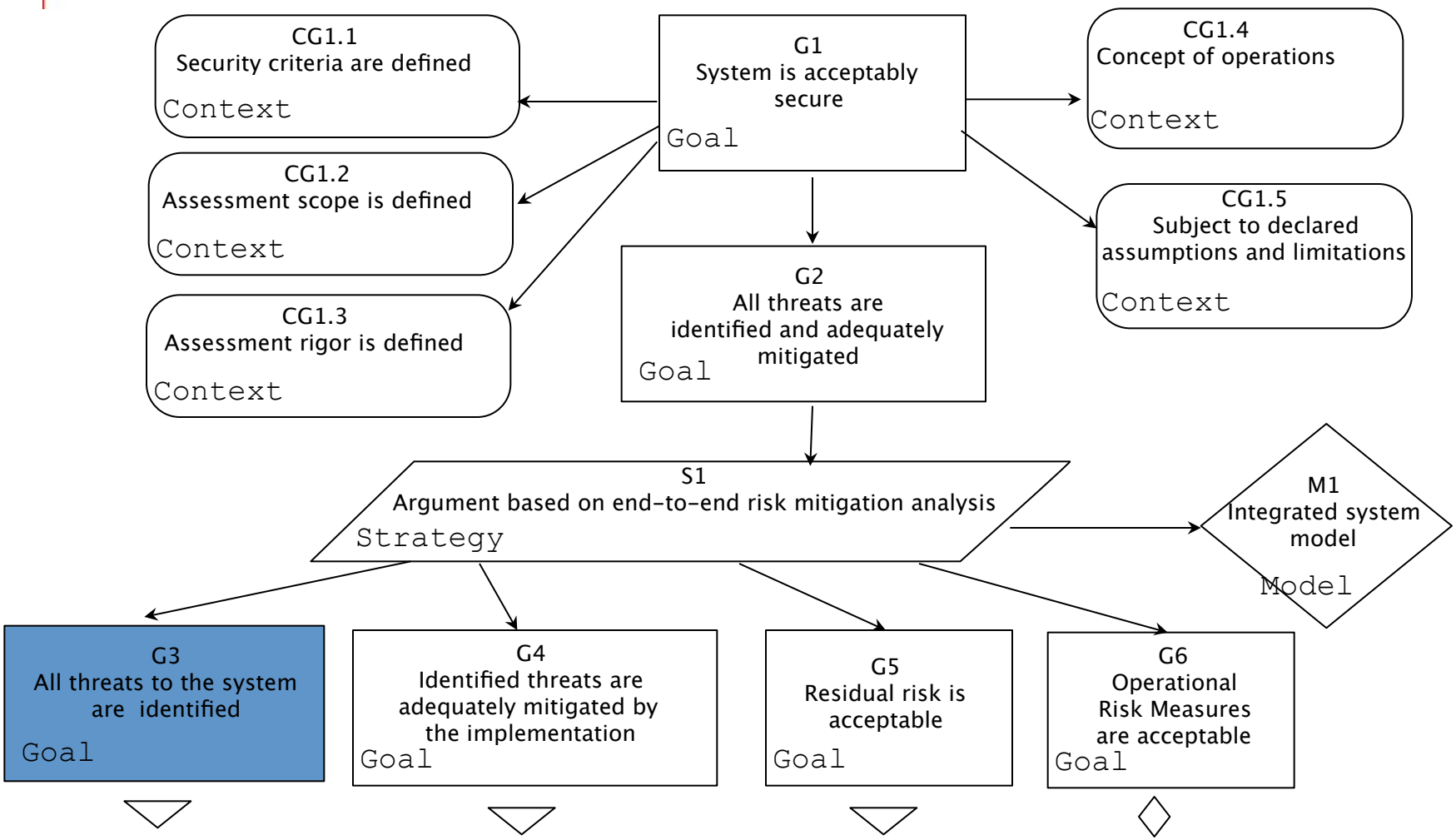


# *Vulnerability Mitigation Argument*



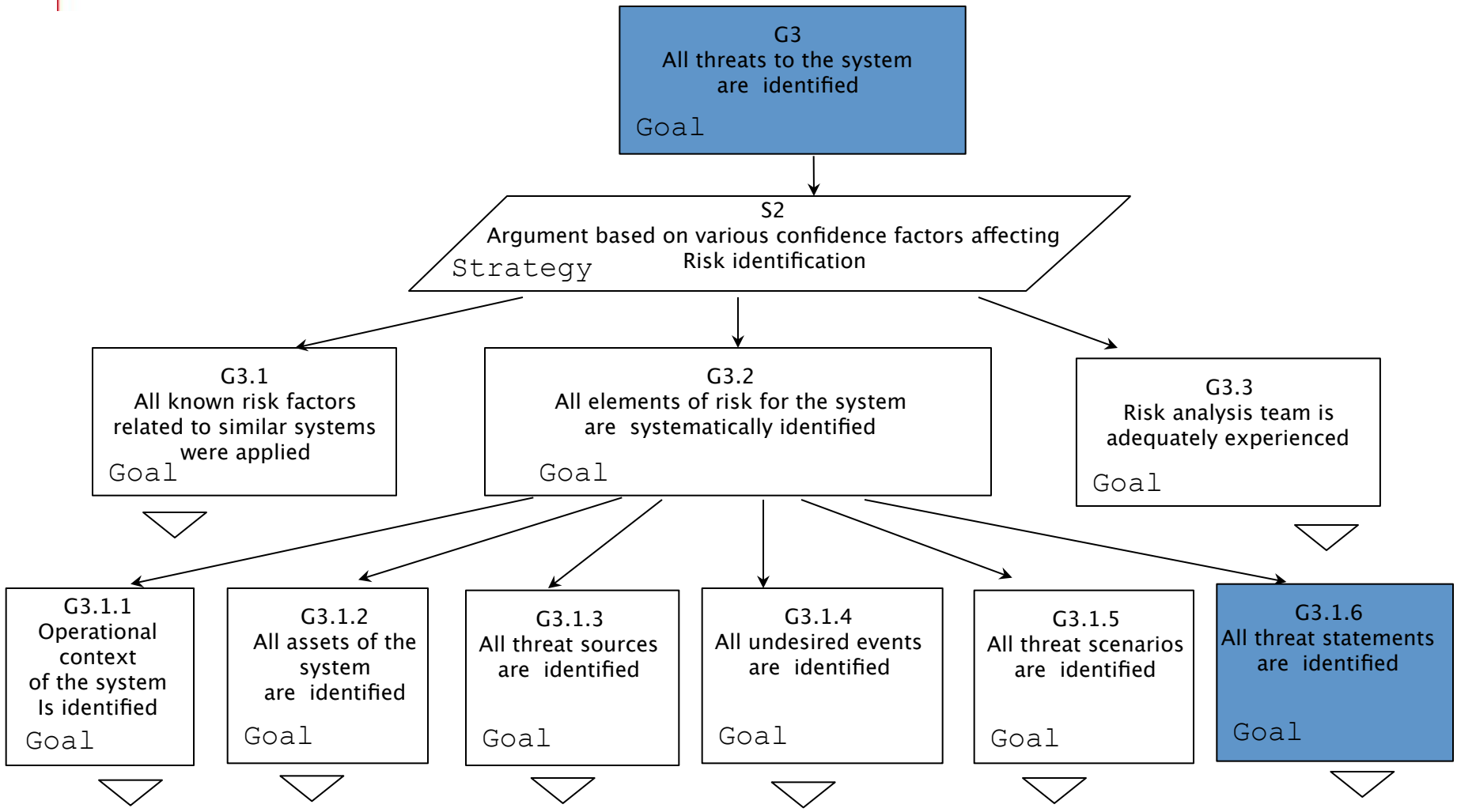



# *Security Assurance Case (back to the top)*



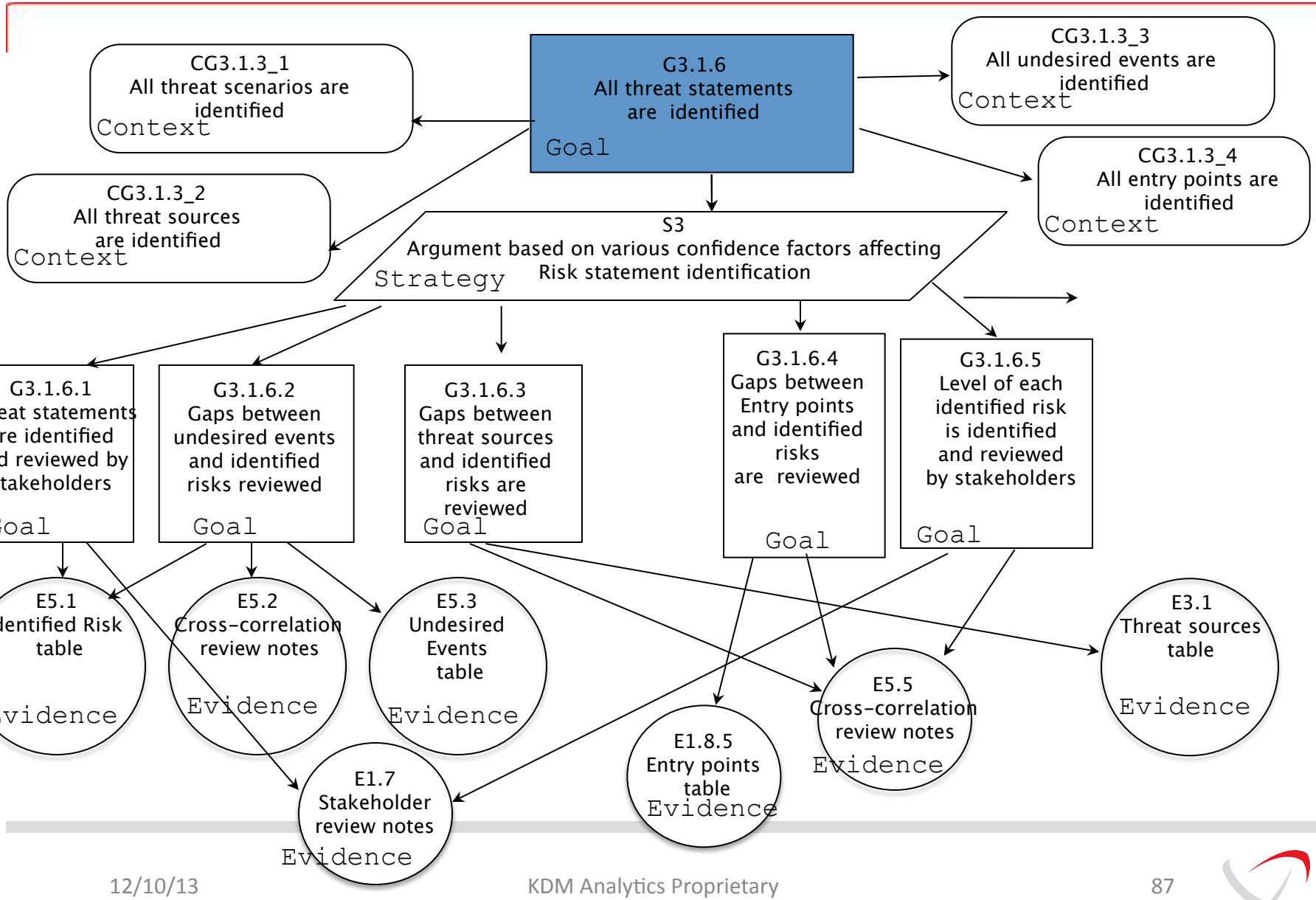



# *Threat identification argument*






# *Threat identification argument (cont'd)*



# *Evaluation of confidence*


Confidence = ∫(Importance of Claim, Strength of Evidence)



***QUESTIONS ?***