Security Assurance Cases

Mazen Mohamad

Chalmers | University of Gothenburg, Sweden

mazenm@chalmers.se

Master Course “Secure Software Engineering”
Summer Semester 2022

Learning objectives

- The emerging importance of security assurance and the driving forces
- Structure of Security Assurance Cases (SAC)

Reading material about Security Assurance Cases
R. Alexander, R. Hawkins, T. Kelly, "*Security Assurance Cases: Motivation and the State of the Art*", The University of York, 2011

Reading material about how to build Security Assurance Cases
M.Mohamad, Ö.Askerdal ,R.Jolak, J,Steghöfer, R.Scandariato, "*Asset-driven Security Assurance Cases with Built-in Quality Assurance*", IEEE/ACM 2nd International Workshop on Engineering and Cybersecurity of Critical Systems 2021
, 2021

- Usage of SAC
- SOTA and SOP

Security Assurance - What?

https://csrc.nist.gov/glossary/term/security_assurance

- Measure of confidence that the security features, practices, procedures, and architecture of an information system accurately mediates and enforces the security policy. NIST SP 800-39

- The grounds for confidence that the set of intended security controls in an information system are effective in their application. NISTIR 7298

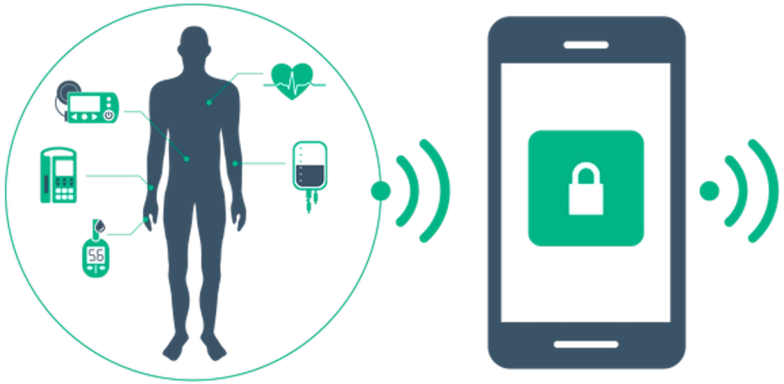
Security Assurance - What?

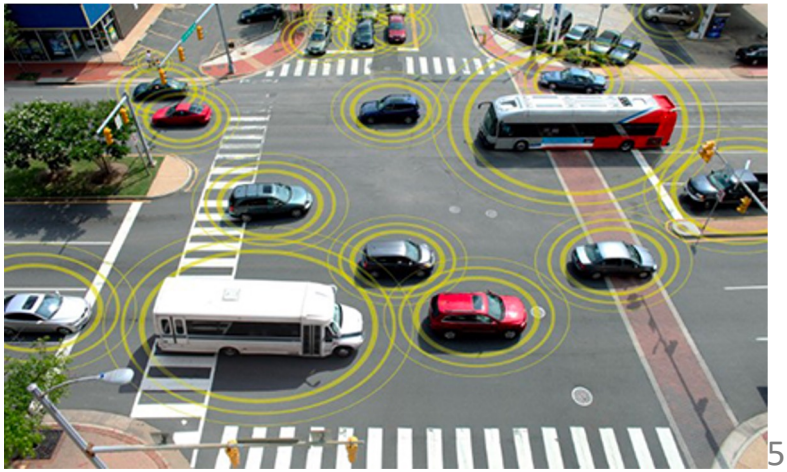
https://csrc.nist.gov/glossary/term/security_assurance

- Measure of confidence that the security features, practices, procedures, and architecture of an information system accurately mediates and enforces the security policy. NIST SP 800-39

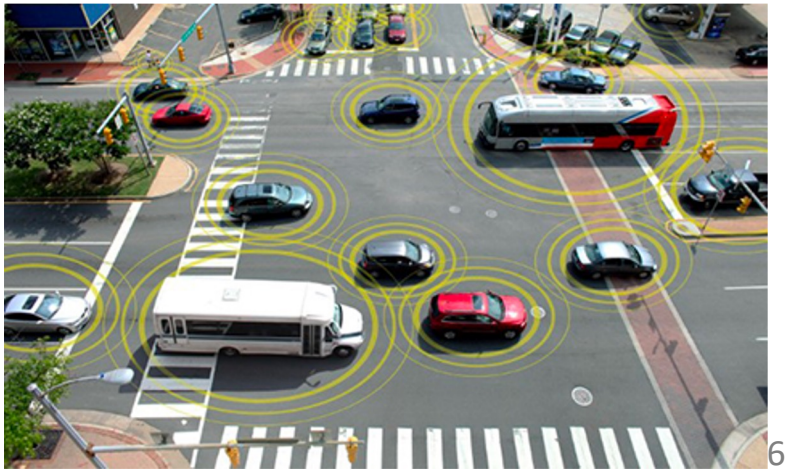
- The grounds for confidence that the set of intended security controls in an information system are effective in their application. NISTIR 7298

Security Assurance - Why?





Security Assurance - Why?



Security Assurance - How?

- There are multiple frameworks and approaches for security assurance, e.g., Common Criteria.

- What we are going to focus on is called Security Assurance Cases (SAC).

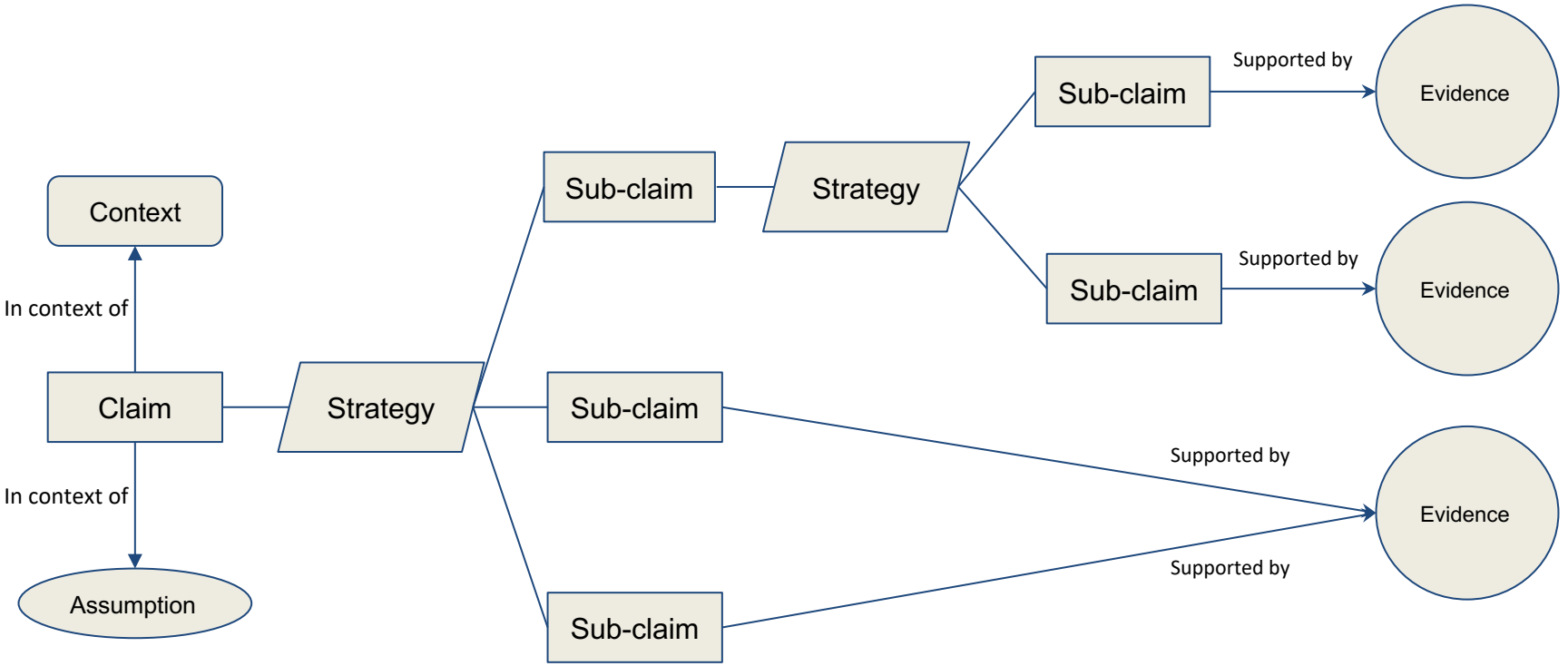
Security Assurance Cases

- “An assurance case is a structured argument, supported by evidence, intended to justify that a system is acceptably assured relative to a concern in the intended operating environment.”

  [Handbook of System Safety and Security, 2017]

In our context the concern is cybersecurity.

Security Assurance Cases



Goal Structuring Notation [Kelly 2004]

SAC - Claims' types

Claim

Sub-claim

- Claiming confidence in the achieved level of security in a specific context. Takes the form: X is acceptably / adequately secure. Where X is an asset / function / sub-system… etc

  Example: The auto parking function is acceptably secure

- Negating the possibility of realizing a harm or threat on a certain asset.

  Example: It is not possible to tamper with the data sent to the steering wheel module.

SAC - Evidence types

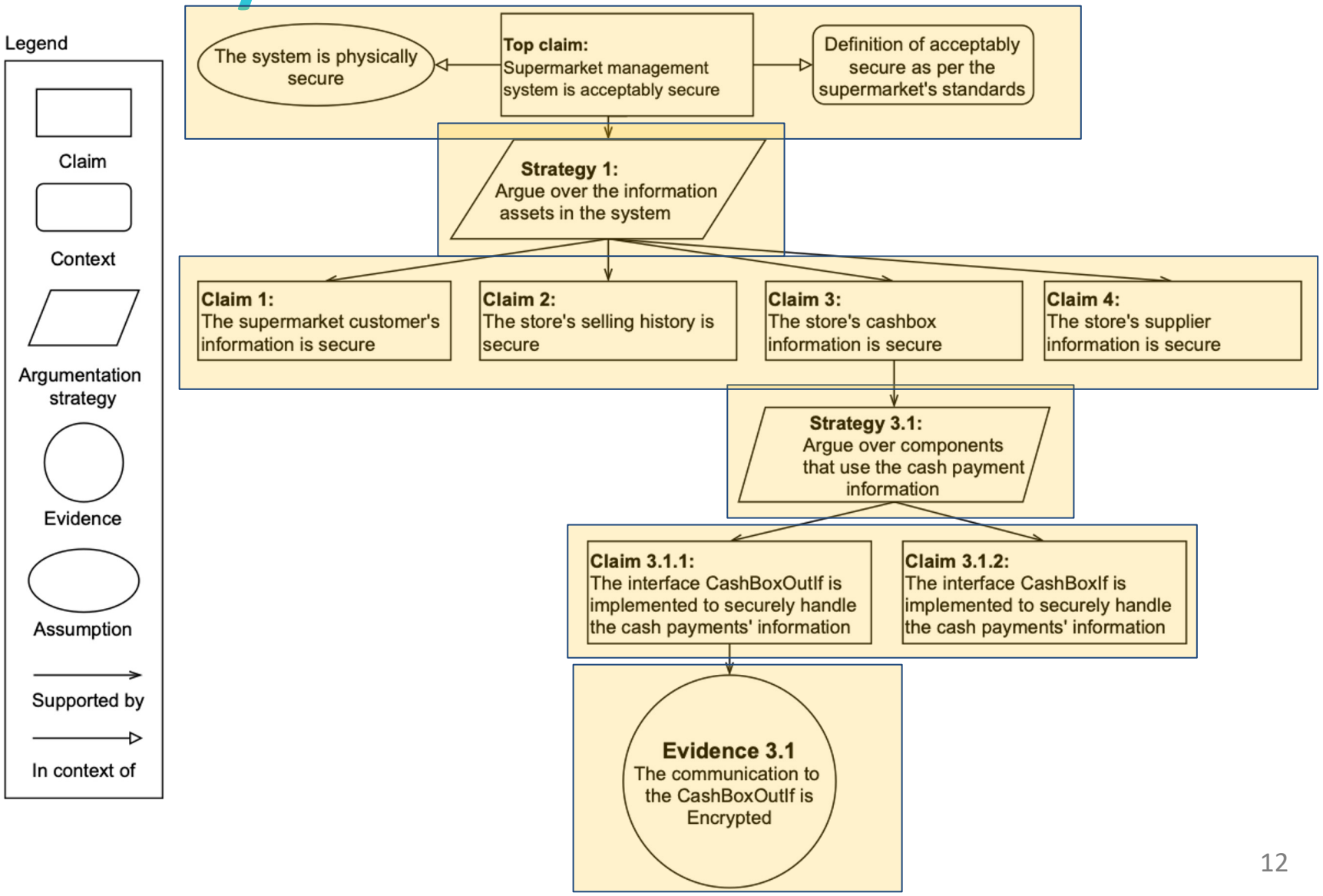
- Reports from test cases
- Code reviews
- Peer review reports
- SME reviews

.

.

.

Security Assurance Cases



SAC - Driving forces

External and Internal forces.

- Current and upcoming standards and regulations in industries. Examples:
  - ISO/SAE 21434 - Road vehicles cybersecurity
  - UNECE 115 - Cyber security and cyber security management system

- Potential for many usage scenarios.

Proven approach from safety.

How do you think SAC can be used in practice?

Go to menti.com – xxxx yyyy

Usage Scenarios

Many usage scenarios identified in industry.

The top ones are:

- Prove conformance / compliance with security standards and regulations by the compliance team
- Assess the security quality of a product by product owners
- Use as evidence in court by lawyers
- Use to communicate with suppliers by the purchasing team
- Support security informed go / no go decisions by project managers

SAC - Knowledge Transfer

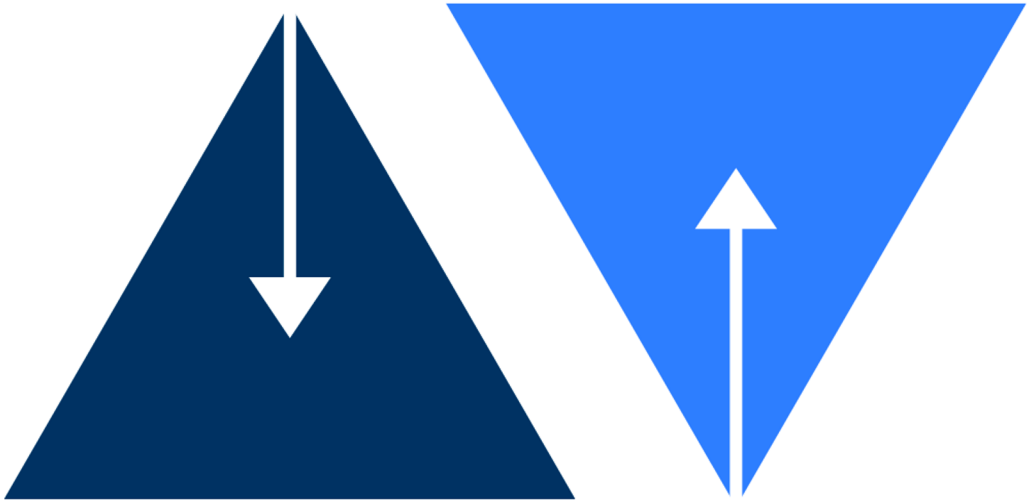
Differences between the domains of safety and security.

- Theoretical differences:
  - Presence of intelligent adversary
  - High level of uncertainty about attackers' behaviour - hence taking measures that are not responses to specific threats
  - Security-critical software often has to adapt quickly as attack patterns change
- Practical differences:
  - Process maturity of security critical practices
  - Safety has more problems with requirements, whereas security with low-level defects in implementation
  - Safety standards are way more elaborated than security ones in terms of development practices

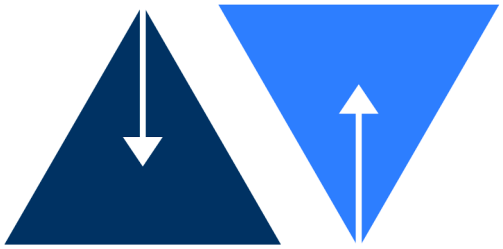
Security Assurance Cases: Motivation and the State of the Art - Rob Alexander et al. 2011

How to build SAC

SACs can be build in different ways. There are two main strategies



How to build SAC



Top-down strategy:

- Starting from the top claim and working our ways down to the evidence
- This is the most common approach

Bottom-up strategy:

- Works by looking at the artifacts and evidence we have, and build the arguments based on them.
- More common for systems that are already built

How to build SAC

Literature includes many approaches, e.g.,

- Argumentation strategies
  - Standard based
  - Security requirements
  - Software components … etc
- Structures:
  - Layered-based
  - Document retrieval

How to build SAC

Limitations:

- Wide variety of approaches.. But

Cover both process and product

- Lack of quality assurance

Actively assessing the quality of SAC

- Imbalance in coverage

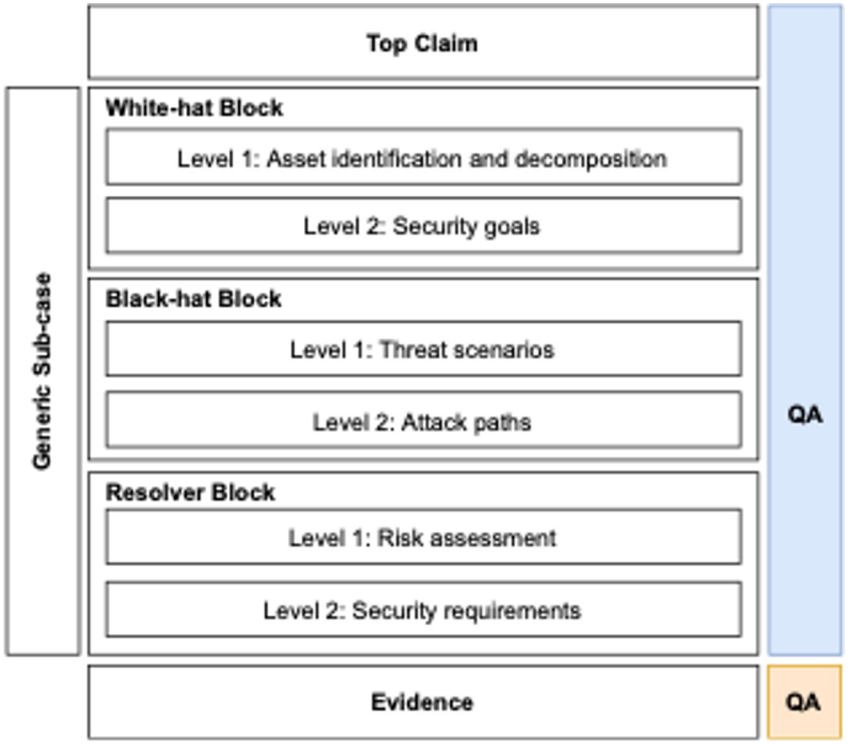
The challenging nature of working with SAC

CASCADE

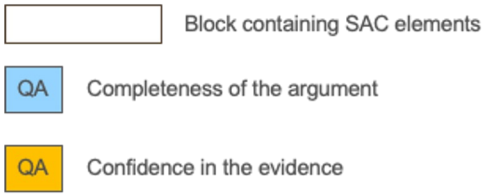
Block containing SAC elements

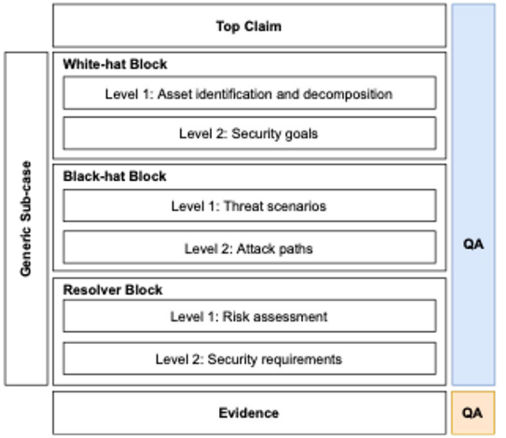
QA Completeness of the argument

QA Confidence in the evidence



CASCADE



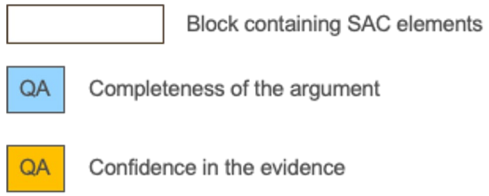


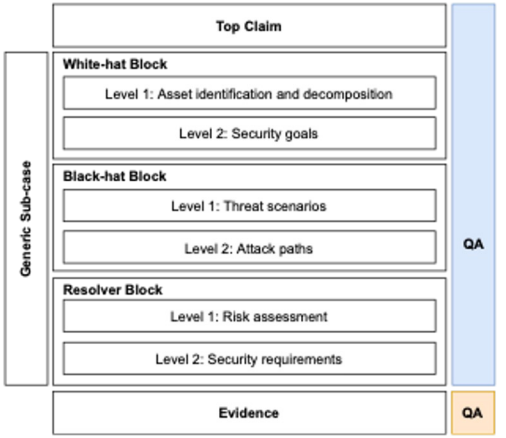
CASCADE is an asset-driven approach which provides a block-based structure for creating the arguments of a security assurance case.

It is asset-driven, as the arguments start from the identification of assets which exist in the system.

The blocks in CASCADE include elements of SACs.

CASCADE





Quality assurance:

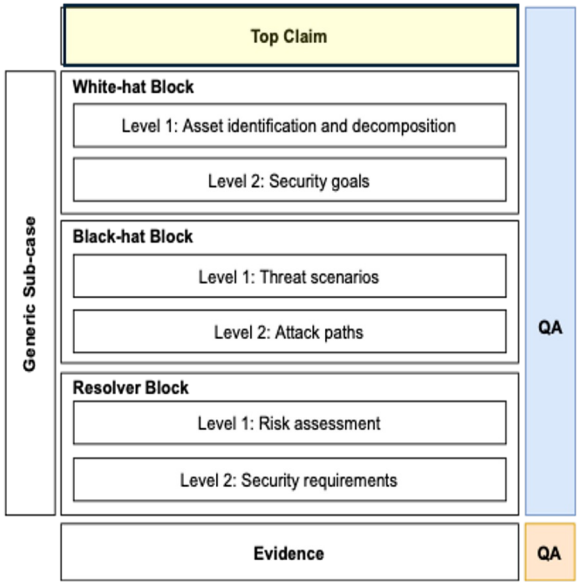
- Completeness of the argument : To make sure that the arguments are complete within the given scope documented in the context and assumption nodes.
- Confidence in the evidence: indicates the level of certainty that a claim is fulfilled based on the provided evidence

CASCADE - Top Claim

The top claim block includes the top claim, its context, and the assumptions we make on the highest level of the argument.

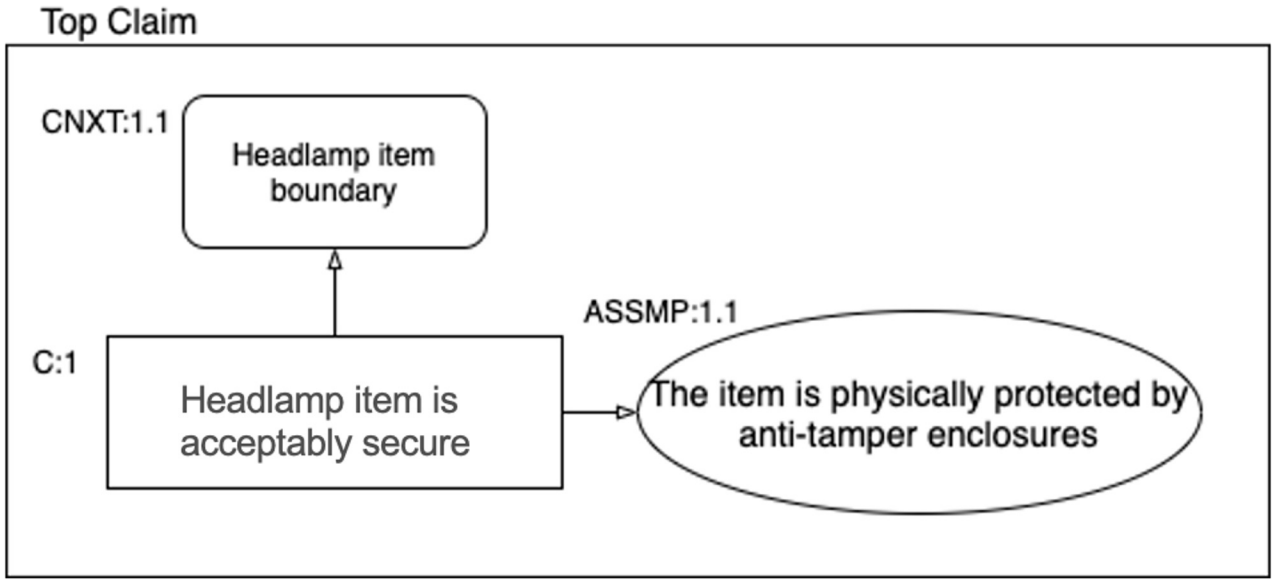
The top claim decides the abstraction level of the SAC, e.g., whole product, end-user function, component… etc.

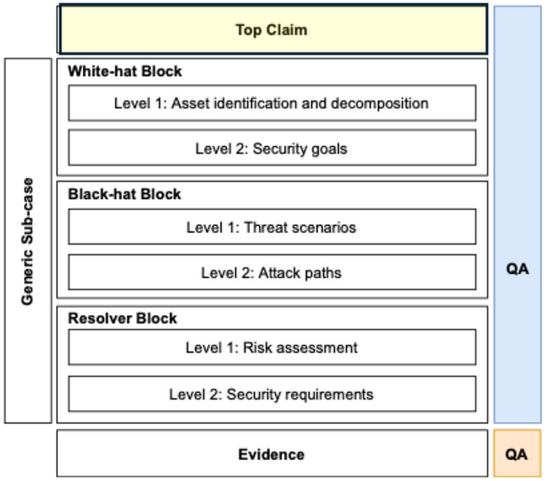
The context in the top claim block decides the scope of the whole argument with the support of assumptions.



CASCADE

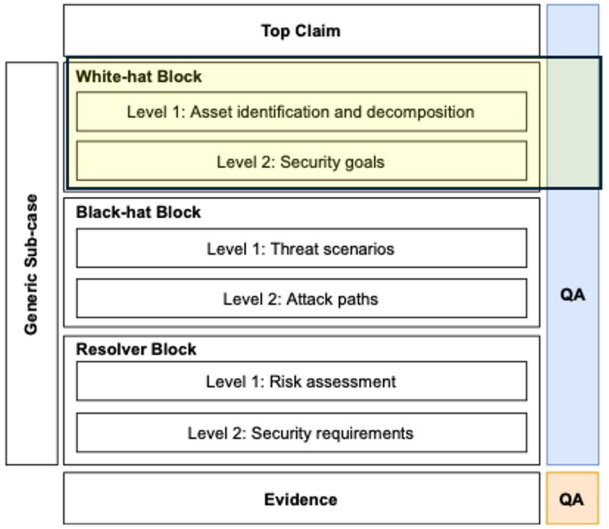
TOP CLAIM BLOCK





Headlamp example
ISO/SAE DIS 21434
Appendix G

CASCADE - White-hat

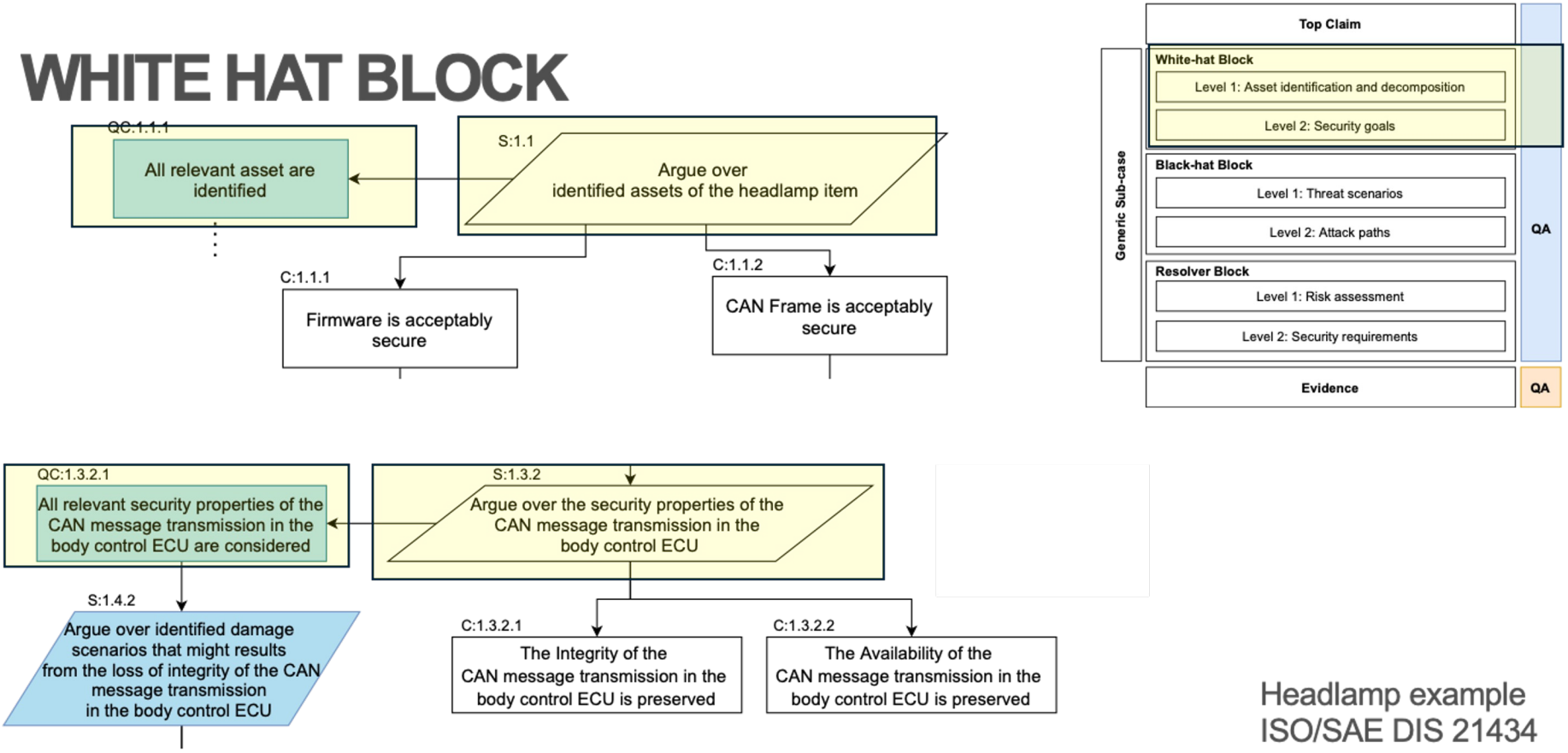


This block has two levels:

- Identification of assets: this is done by conducting an analysis to find the artefacts of the system that are likely to be subject to an attack, then creating claims about the security of these assets

- Security goals: done by identifying relevant security properties for each asset, and then creating claims about preserving these properties for each asset.

CASCADE



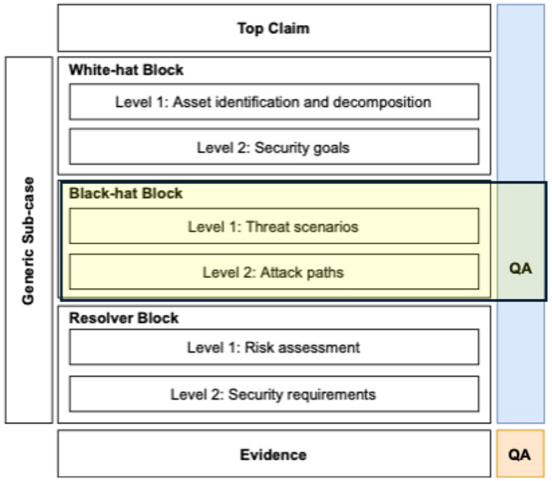
Headlamp example
ISO/SAE DIS 21434
Appendix G

CASCADE - Black-hat

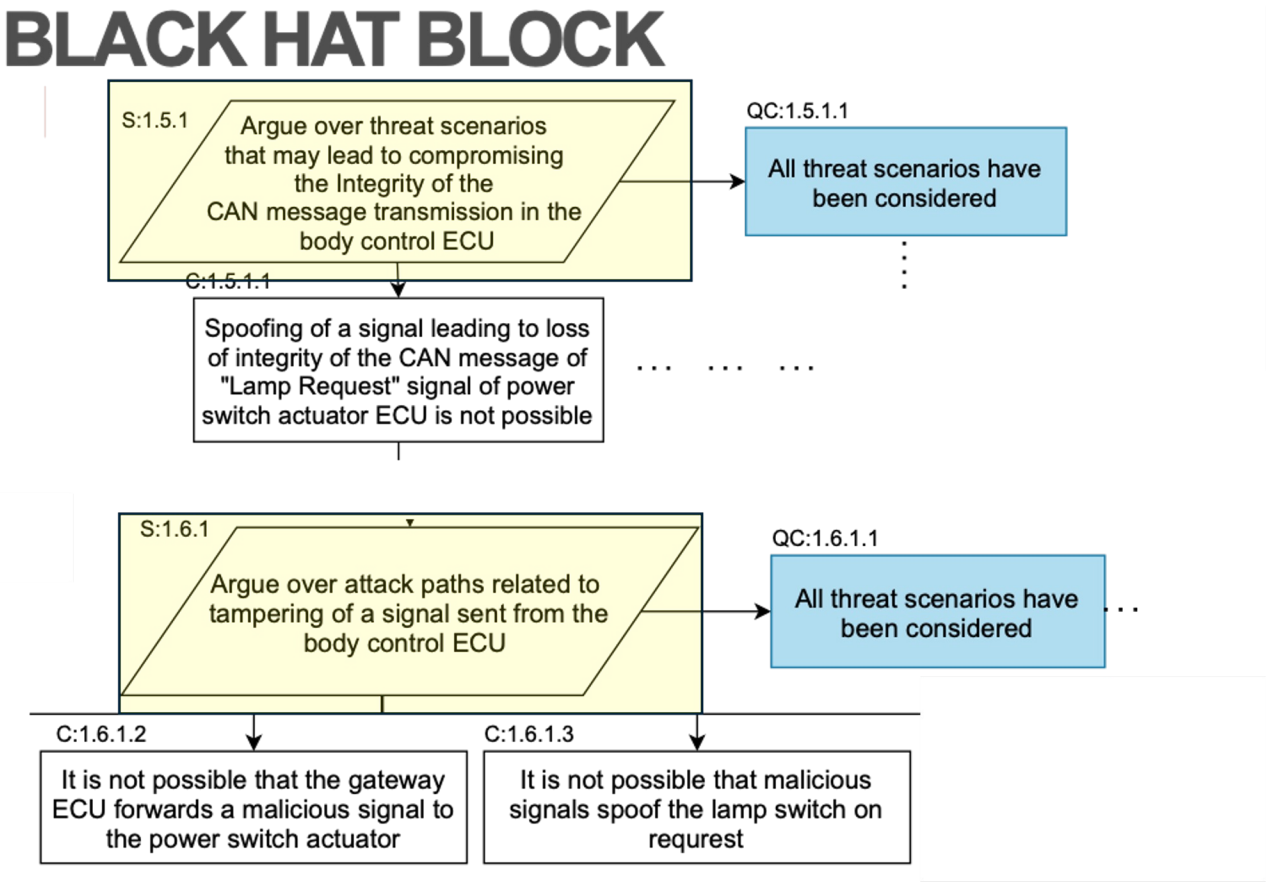
This block has two levels:

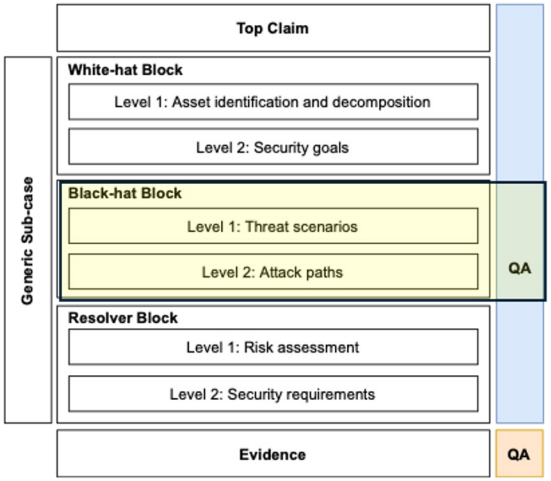
- Threat scenarios: this is done by identifying the threats that might compromise the security goals identified in the white-hat block - security goals level. Then we create claims negating the possibility of these threats.

- Attack paths: done by identifying ways in which an attacker can realize the threats we identified in the earlier level. We then create claims negating the possibility of these attack paths taking place.



CASCADE





Headlamp example
ISO/SAE DIS 21434
Appendix G

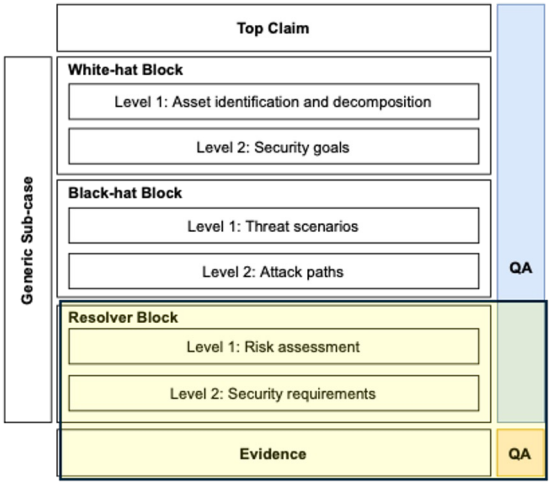
CASCADE - Resolver and Evidence

This block has two levels:

- Risk assessment: In this level, we assess the risk of the identified attack paths. Based on the risk level, the creators of the SAC create claims to treat the risk by, e.g., accepting, mitigating, or transferring it.

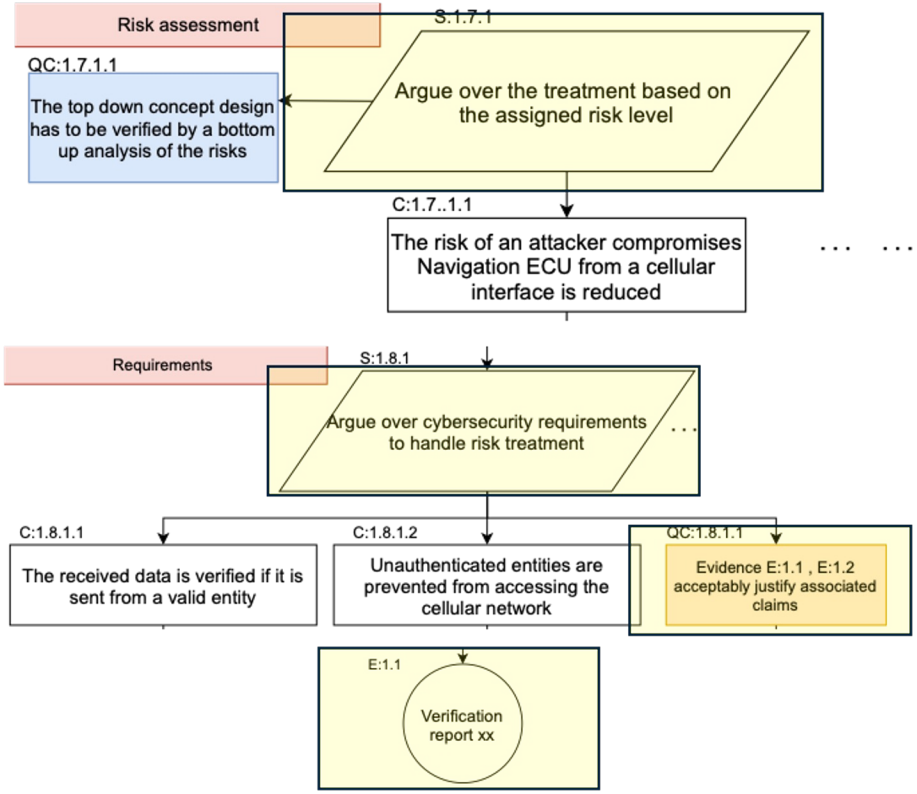
- Requirements: At this point, requirements of risk treatments identified in the previous level are to be expressed as claims.

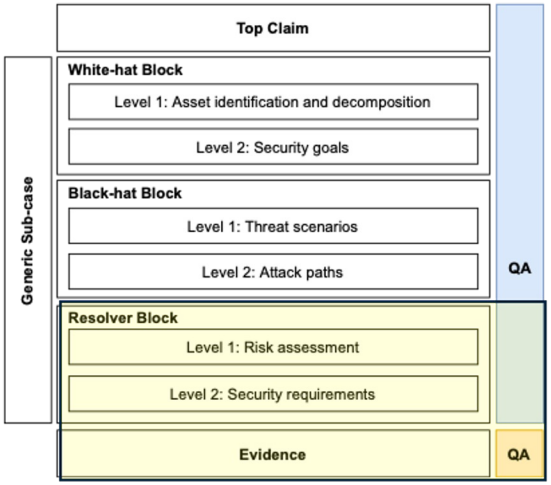
- Evidence: When claims about the security requirements are identified, we assign evidence to justify / solve these claims



CASCADE

RESOLVER BLOCK





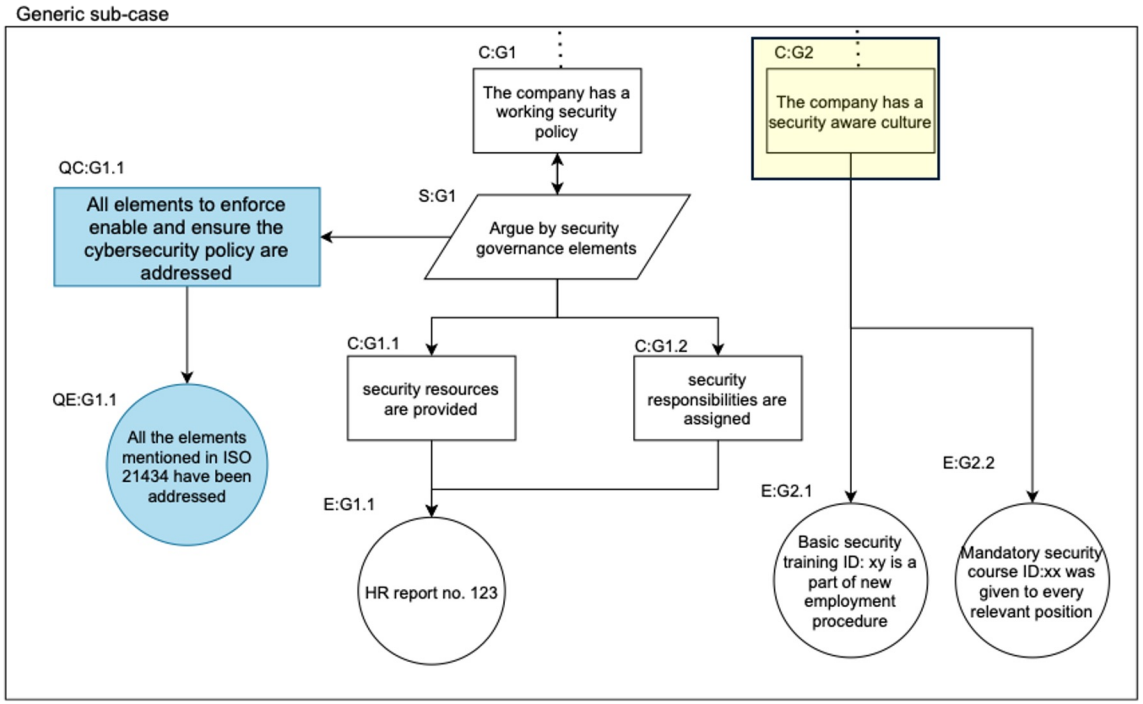
Headlamp example
ISO/SAE DIS 21434
Appendix G

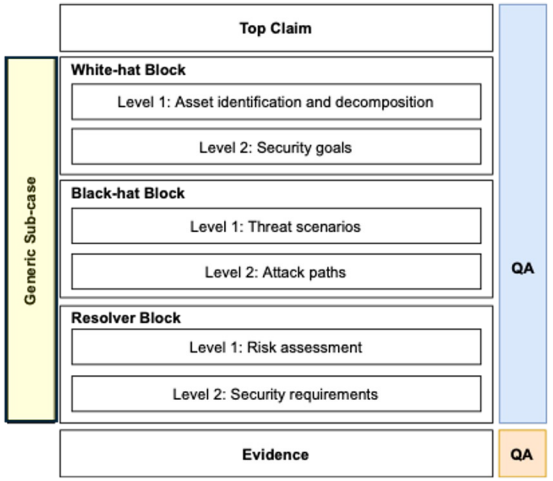
CASCADE - Generic Sub-case

This block contains a sub-case that is applicable not only to the artefact for which the SAC is being created but instead to a larger context. For example, if a company defines a cybersecurity policy, enforced by cybersecurity rules and processes, then the policy can be used in security claims for all its products. These claims can be re-used when creating SAC for individual artefacts.

CASCADE

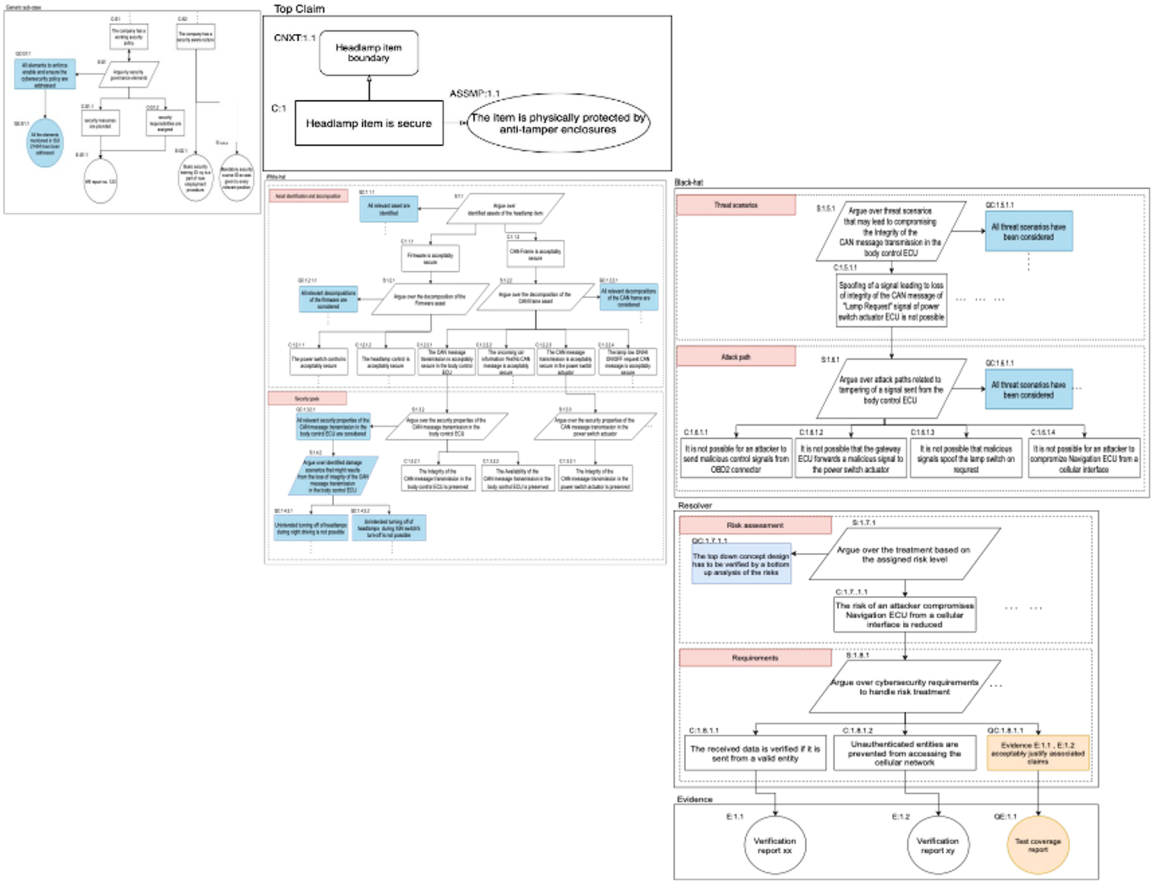
GENERIC SUB-CASE

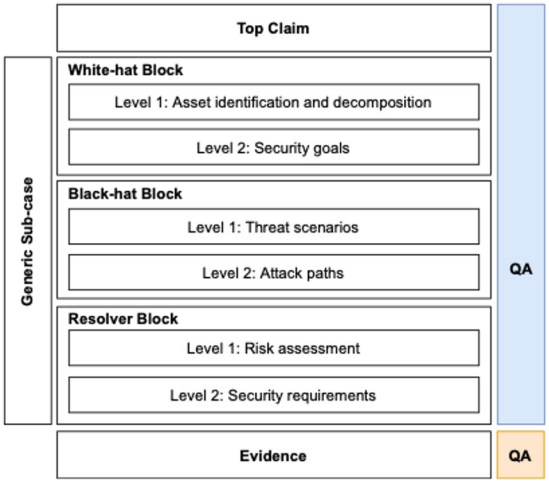




Headlamp example
ISO/SAE DIS 21434
Appendix G

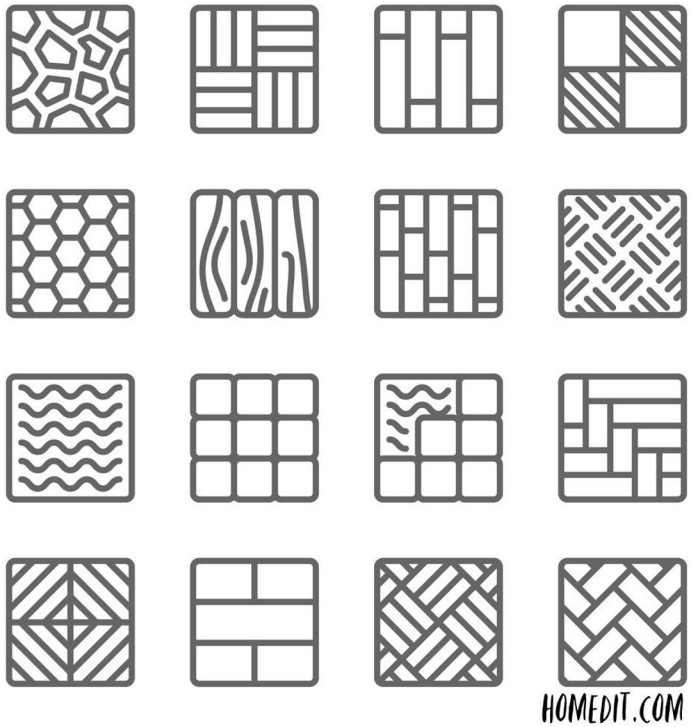
CASCADE


Top Claim
CNXT:1.1
Headlamp item boundary
C:1
Headlamp item is secure
ASSMP:1.1
The item is physically protected by anti-tamper enclosures
White-hat
Black-hat
Resolver
Evidence
Verification report xx
Verification report xy
Test coverage report


Top Claim
Generic Sub-case
White-hat Block
Level 1: Asset identification and decomposition
Level 2: Security goals
Black-hat Block
Level 1: Threat scenarios
Level 2: Attack paths
Resolver Block
Level 1: Risk assessment
Level 2: Security requirements
Evidence
QA
QA

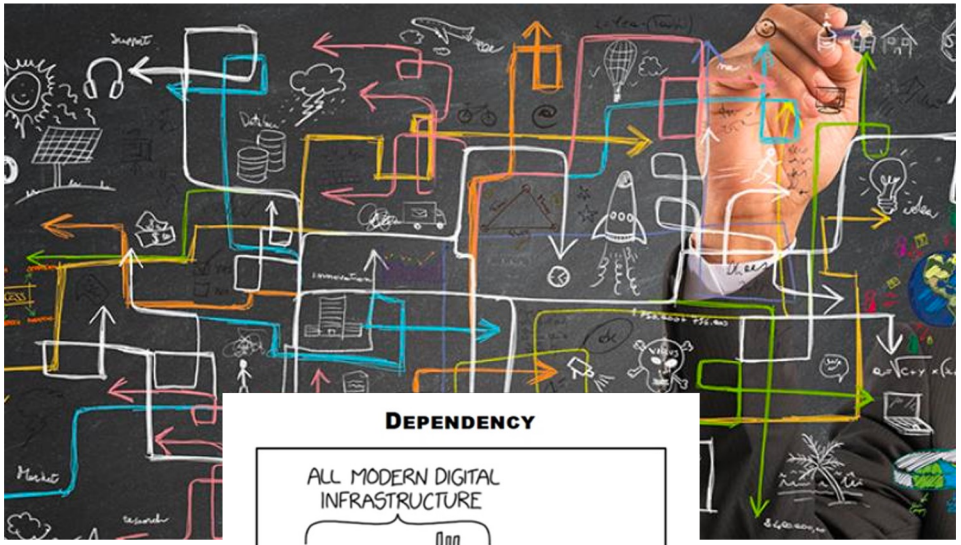
State of Practice - Automotive

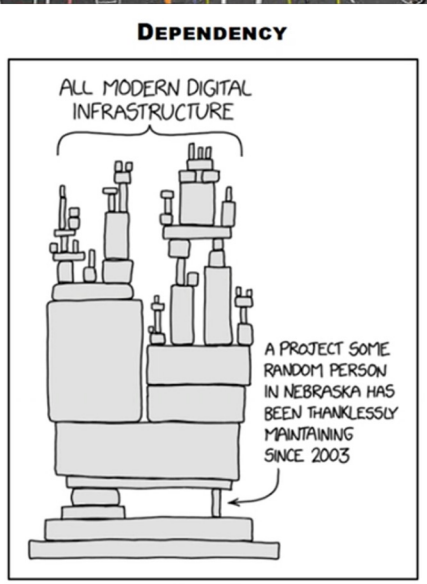
Creating patterns of arguments to be reused



State of Practice - Automotive

Handling the complexity of automotive products and processes.
For example the level of dependency among the systems





https://tomorrowtodayglobal.com/2017/09/05/tuesday-tip-simplify-2-ways-deal-complexity/

Research areas

- Compositionality
- Automation
- End user assurance
- And many more
- Applying to other domains (health care)

Is it an impediment?

Go to menti.com – xxxx yyyy

Questions ?

More questions at a later time?

mazenm@chalmers.se