

Is this the right room for an argument - Improving arguments for safety and security

Dr T.J. Cockram PhD, CEng, MIEE, MSaRS
Praxis High Integrity Systems Limited

ABSTRACT: An argument is a set of connected statements intended to establish a proposition. In the context of safety, a safety argument is an essential part of a safety case. This argument needs to be as clear and concise as possible to provide a logical case that the equipment is safe. This idea is being extended into aspects of security where it is intended that a case be established that the security of a system can be shown by means of a similar logical argument. Safety arguments have been developed using a number of different graphical techniques including claim structures, claims argument evidence, goal structured (structuring) notation (GSN) and Toulmin logic. In this paper I also extend the notation to address the important issues of counter evidence and design options.

1 INTRODUCTION

One famous definition of an argument is “A set of connected statements intended to establish a proposition” (Python M 1989). An alternative definition is “A well organized and reasoned justification based on object evidence.” (MoD 1997)

A safety argument is an essential part of a safety case. The argument needs to be as clear and concise as possible to provide a logical case that the equipment is safe. Safety arguments have been developed using a number of different graphical techniques including claim structures (MoD 1997), goal structured notation (GSN) (YSE 1997), and Toulmin (Toulmin S 1969). Current practice has centered on the use of GSN for producing safety arguments (Kelly 1998). GSN provide a means of modeling the success case for the safety argument in a clear, structured and well defined manner.

1.1 *Safety Cases*

The provision of a safety case is a requirement of many standards (HMSO 1992, MoD 1996, 1997, HSE 2000). A Safety Case presents the argument for the safety of a system and summarises and justifies the supporting evidence and is a key input to the safety approval process for a system. The body responsible for safety approval will consider all relevant safety submissions, primarily the Safety Case and supporting documentation such as reports of Safety Audits and Assessments, in order to satisfy themselves that the system is adequately safe and

conforms to the relevant international, national and industry safety standards. The essential elements of the argument for the safety of the equipment are often spread over different parts of the safety case, (for example the argument that risks have been reduced to an acceptable level may appear in a hazard log report, and the justification for sound engineering appearing in another part of the safety case). The overall logical reasoning is also often lost within text. This makes a strong case for presenting the main safety argument in graphical form.

2 PROBLEMS WITH CURRENT ARGUMENT STRUCTURES AND SEMANTICS

Experience in the use of Goal Structured Notation for the production of safety cases has shown that: the approach although relatively successful has limitations. These limitations are principally:

- There is no easy semantics to model logical operands. The GSN is read as a series of ‘ANDed’ operations, so for the argument to be true all elements of the case must be true. The argument must contain all the necessary and sufficient elements to establish the argument. For a single instance of the safety case it is possible to provide an argument in this fashion only describing those elements, which are true and taking no account of any reservations or counter evidence.
- In addition it is not possible to adequately represent alternative design options.

- We have no means of measuring the completeness of an argument. If there is evidence to support a goal then the implication is that it is sufficient to satisfy the goal.
- There are also issues relating to determining the strength of evidence to support the argument. The GSN argument implies that each leg of the argument has an equal weight.

The suggested GSN extensions are:

- 1 The incomplete/counter-evidence notation (3.1)
- 2 The “OR” notation for option capture (3.2)
- 3 Indicating Primary/Secondary evidence status (3.3)
- 4 Indicating the argument completeness (3.4)
- 5 Argument Justification (3.5)
- 6 Argument semantics and connectivity rules (3.6)

The GSN extensions have been applied in several industrial case studies and have been to subject a debate within the safety case community, and further debate is welcome.

3 EXTENDING GSN

3.1 Managing Counter evidence

The requirement to address counter evidence within a safety case has been recently introduced (MoD 2004). There is currently no means of modeling challenges, questions or reservations to the argument with the current GSN structure. In other words we need a “No it isn’t!” structure. Counter evidence is any evidence that indicates that the proposition is not completely true. As in the example below, some of the verification test evidence show that not all the safety requirements tests have been completed satisfactorily. A simple negation symbol (small circle) on the solution node indicates this.

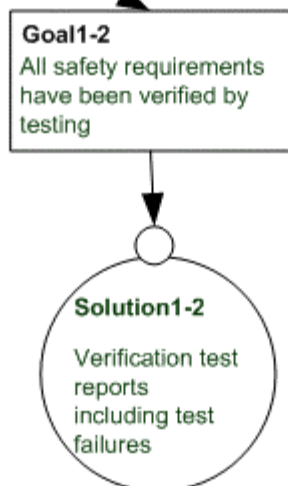


Figure 1 Counter evidence

For the safety argument to hold we need to add a rebuttal to the test failures by providing additional

argument addressing the test failures. The approach presented below in figure 2 is to provide a strategy that justifies the cases where the test schedule was incorrect but the functional behaviour was correct, and additional evidence to cover the equipment modification and subsequent re-testing.

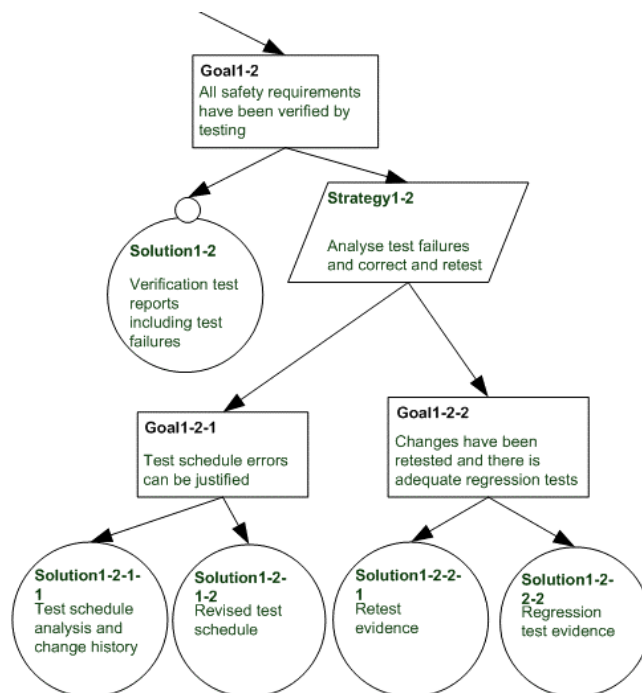


Figure 2 Rebutting the counter evidence

3.2 Arguing alternative approaches

We need to find a means of arguing alternative means by which we can satisfy the (functional, safety, security or environmental) objectives for the equipment. We can have a number of options. In the nuclear industry this approach is known as option-eering. As a simple example an analysis approach may provide a means to show that a safety requirement has been met, alternatively testing may be used to provide the required level of assurance. Each approach may be equally valid, but currently only one would be used in the safety case. If there was an ‘OR’ combiner model within GSN alternative strategies could be made explicit within the safety case with the obligation to show that one strategy/solution arm had been satisfied. The evidence answering the alternative goals will need to indicate the strengths and weakness of alternative approaches and to justify the actual approach taken see figure 3. Two alternative methods are represented as sub-goals under the OR symbol with both positive and counter evidence for each option recorded. Sub-goal 1-3 covers the selection of the better option and the criteria and results of the selection process.

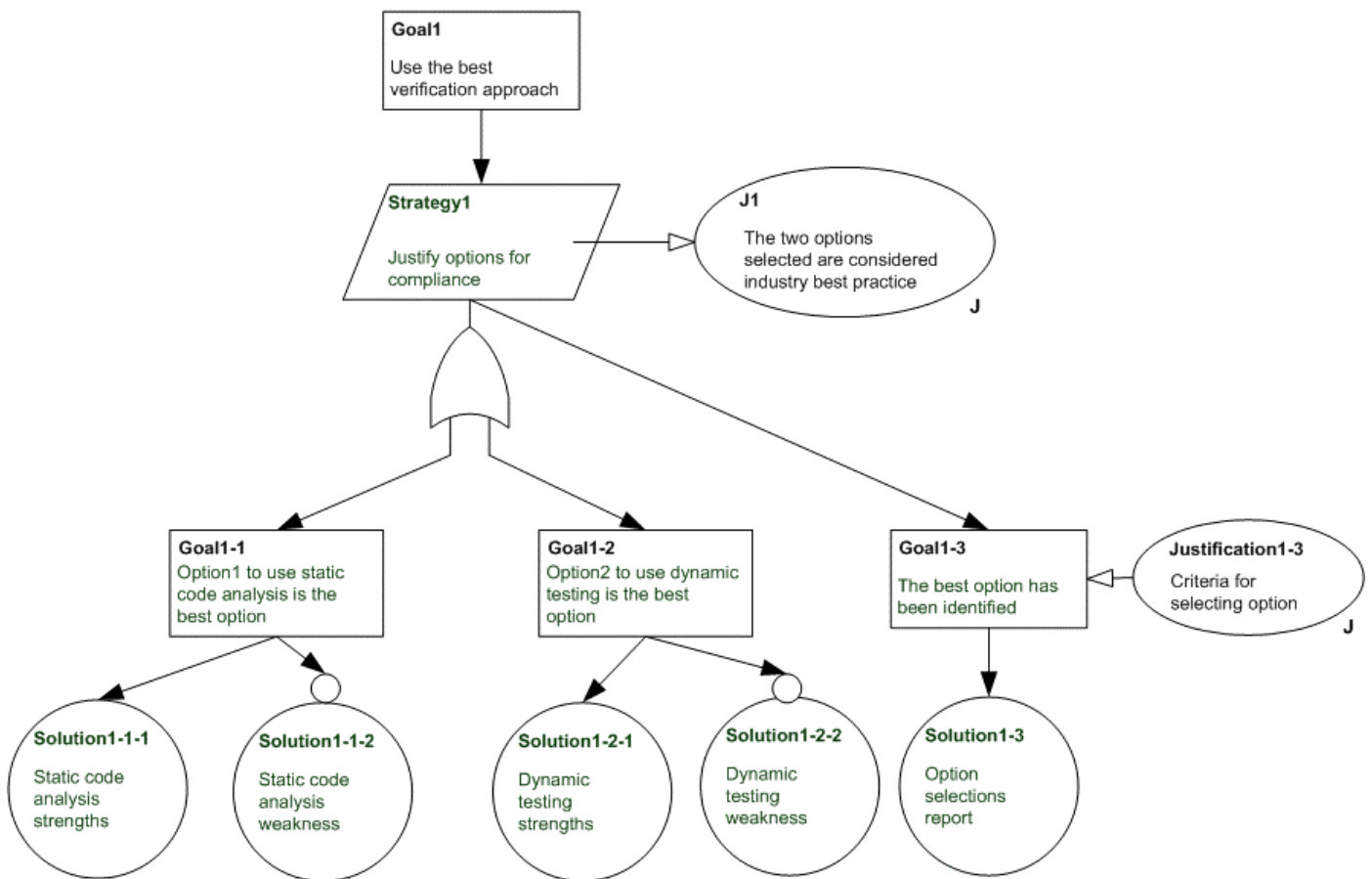


Figure 3 Representing options

3.3 Weight of evidence

The basic GSN is an implied AND structure where logically each solution has equal weight. However this does not address the strength of evidence within solution nodes. Some argument tools have introduced the concept of strength of solutions by making the width of the connecting line increase with the strength of evidence.

As an example of the issue: one sub-goal of a goal that “The design meets its safety requirement” is that “There are satisfactory test results from requirements verification testing”. This is primary evidence as it directly relates to the product and to demonstrating the product is fit for purpose.

Another sub-goal is “The equipment has been developed in accordance with the relevant standards”. this sub-goal provides a supporting argument. Many arguments have been generated using the compliance to standards as a demonstration of adequacy, however “A statement claiming compliance with any standard is unlikely to provide adequate support for a Safety Case.” [Rodger K 1989]. By this Keith Rodger is stating that compliance to a standard is not primary evidence, hence it carries less weight than the product-focused evidence, therefore safety arguments should be based on primary evidence and supported by supporting evidence.

To indicate the type of evidence we have proposed the use of a filled arrowhead to indicate the primary and an open arrowhead for supporting evidence as illustrated in figure 4 below.

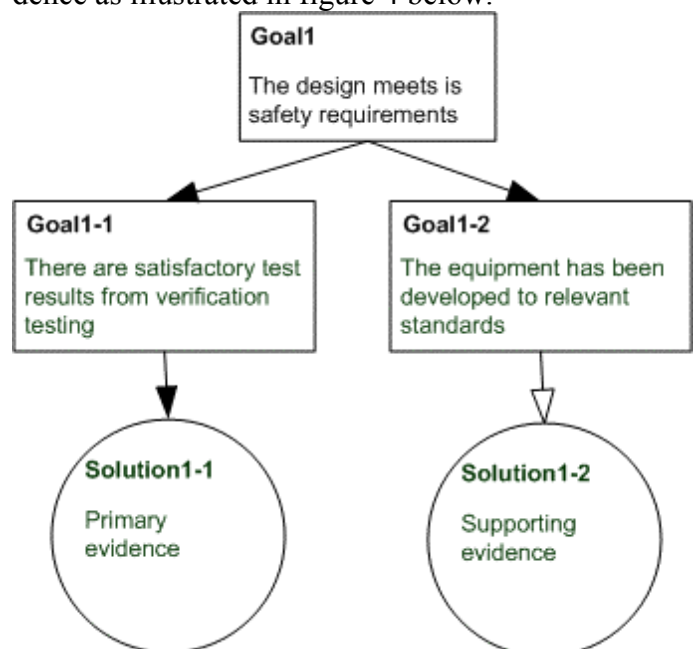


Figure 4 Indicating primary and supporting evidence

3.4 Completeness

An important issue with a safety argument is its completeness. During the evolution of a safety case the safety argument will evolve and evidence supporting the argument will be at various stages of completion. Ideally completion is a self-evident logical conclusion, i.e. a proof of completion, however, in many cases the evidence provided needs to

be judged to determine if it does provide the complete solution to the goal. This is often a subjective judgement and may be of a qualified form of confidence; i.e. the evidence appears to satisfy 50% of the goal's objectives. There are more objective and statically means of measuring confidence in a safety argument, which is a subject for a further paper.

3.5 Argument Justification

During the first phase of the ASAM project (Wilson et al 1996) attempts were made to use a Toulmin style of argument for safety cases. These had the disadvantage of producing cluttered argument structures with a greater number of semantic elements than the equivalent GSN structure. The basic form of the argument is for a single proposition. Its 'T' form of structure is also a little cumbersome when a number of arguments are combined. However in the view of the author the logical structure of a Toulmin form argument can be used as a basis of developing the GSN argument structure.

Context statements have been used to provide limitations on the validity of the safety argument, but there is no semantic equivalent to 'UNLESS' within the current structure.

The approach suggested is to determine the positive conditions for which the argument holds. Providing evidence for a justification statement as illustrated in figure 5.



Figure 5 Evidence for justification

3.6 Connection rules and semantics

A basic GSN argument should be of a form as illustrate in figure 6. This can be representing as:
 [Goal] X is satisfied by sub-goal Y given [Solution] Q, supported by [Solution] P because of approach [Strategy] S [justified] by J because of evidence from [Solution] A

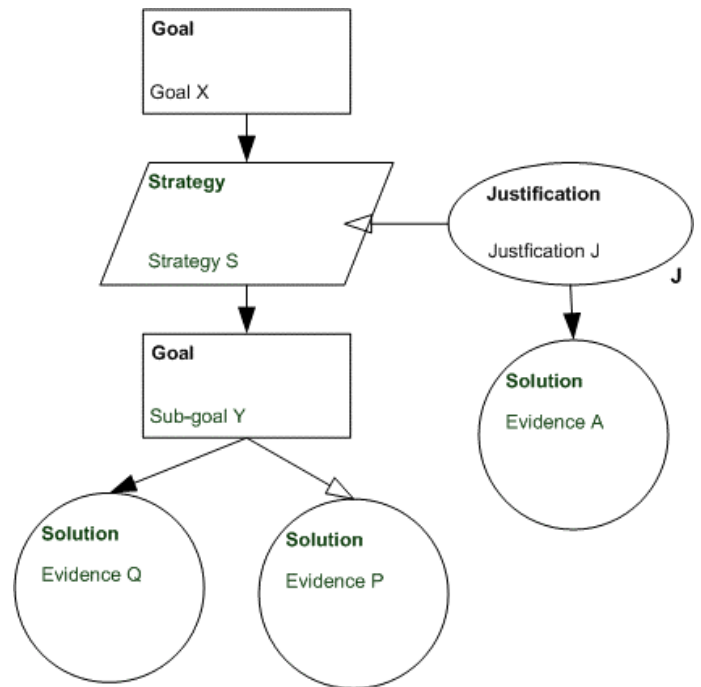


Figure 6 Basic GSN argument form

The sub-goals may be further refined or decomposition as above, or may be satisfied by means of evidence in solutions. The depth of decomposition is self-contained. When the evidence is unlikely to be challenged or is self-evident then the argument is complete. To clarify the weight of the evidence for sub-goals of this type should be classified as either direct evidence or supporting evidence.

We can use this basic form of argument structure to determine if the connections within the argument are correct.

The modular GSN approach also fits this model where the module identifier acts as a wrapper to the semantic element.

4 ARGUMENTS FOR SECURITY

Security is becoming more important to all of us and there is a growing need to know that a system is sufficient resilient to security threats. We can apply a similar argument technique to security assurance (Lautieri et al 2004), but the actual argument must reflect the category of the system being considered. Taking the common criteria (ISO 1999) for information security and other security considerations it is possible to build an argument for the resilience of a system. Further information can be found on common approaches to safety and security in the results of the Safsec project. (Safsec 2005). An example of the form of a security argument is shown in figure 7 below. In this case the small diamond symbol under a goal indicates that it is an undeveloped goal.

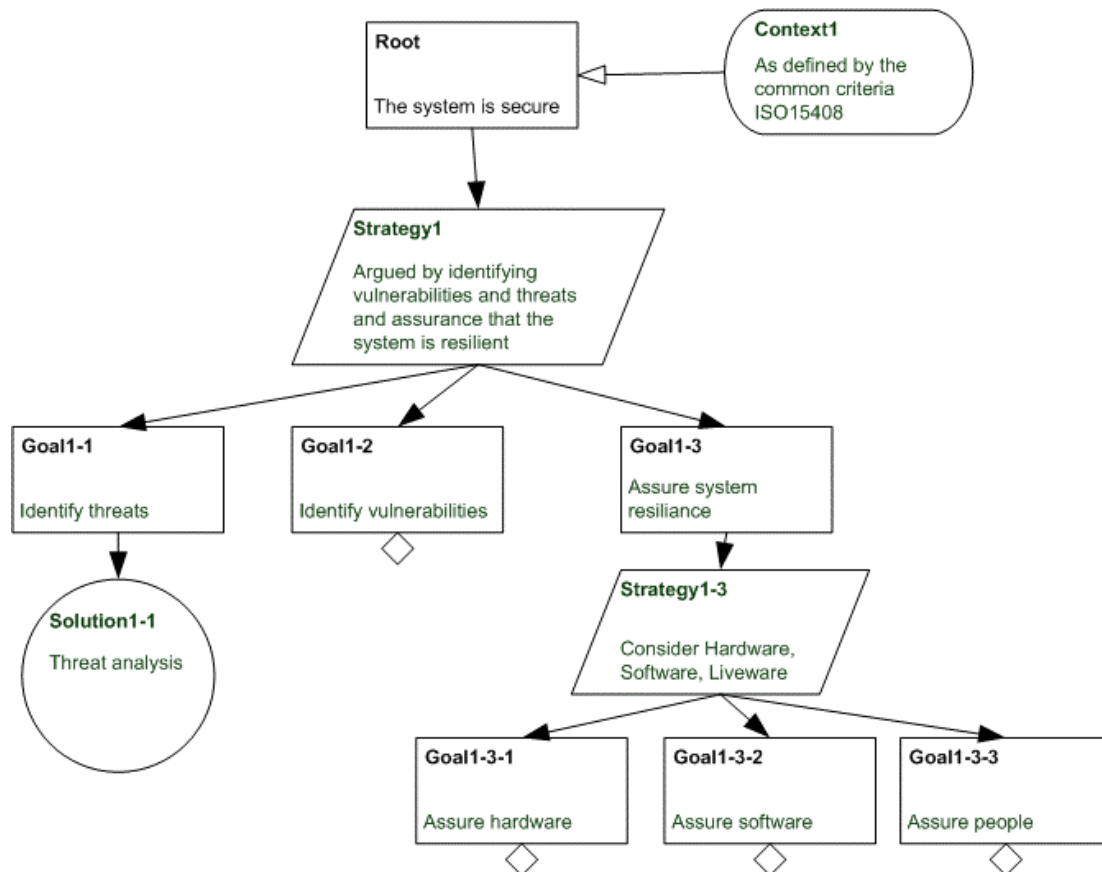


Figure 7 Outline Security argument

Where we make a security argument that can be modularized, in the example above Goal1-3-1 and Goal1-3-2 can be turned in modules within the GSN. We use the modular structure to characterize and specify the interface between the hardware and software.

The approach we have successfully applied is to define module contracts that specify security behaviour across the module boundary. We apply the principle of a verifiable module contract. This contains a guarantee clause specifying the security properties provided by that module and a rely clause which specifies the required properties from the interfacing modules to ensure the provision of the guarantees (including the level of assurance). From these we can define the assurance level, which is the level of confidence we have in the guarantees provided by the module. We also need to supply a context defining any operational assumptions and counter evidence defining any limitations (including defects and residual risks) that exist in the module.

REFERENCES:

- HMSO 1992 *Offshore Installations (Safety Case) Regulations 1992* UK HMSO
HSE 2000 *Railways (Safety Case) Regulations 2000* UK Health and Safety Executive
ISO 1999 ISO/IEC 1508: 1999 *Information technology Security techniques Evaluation criteria for IT security* IEC publications
Kelly 1998 *Arguing Safety – A systematic approach to Safety Case Management* DPhil Thesis York University

- Lautieri et al 2004 *Assurance Cases: how assured are you? in proceedings of the 2004 International Conference on Dependable Systems and Networks 2004*
MoD 1996 *Defence Standard 00-56 Issue 2: Safety Management Requirements for Defence Systems*. Ministry of Defence Directorate of Standardization
MoD 1997 *Defence Standard 00-55 Issue 2: The procurement of safety critical software in defence systems*. Ministry of Defence Directorate of Standardization.
MoD 2004 *Interim Defence Standard 00-56 Issue 3: Safety Management Requirements for Defence Systems*. Ministry of Defence Directorate of Standardization
Python M 1989 *Monty Python's Flying Circus: Just the Words*, Volume 2, episode 29. Methuen, ISBN 0-413-62550-8
Rodger K 1989 *Guidance for the provision of an aircraft Safety Case* DERA Report: DERA/AT&E/MC/TROO05/1.0 March 1998
Safsec 2005 www.safsec.com
Toulmin S 1969 *The uses of argument* Cambridge University Press
Wilson et al 1996 *The Safety Argument Manager: An Integrated Approach to the Engineering and Safety Assessment of Computer Based Systems in IEEE Symposium and Workshop on Engineering of Computer Based Systems (ECBS'96)*
YSE 1997 *Goal Structured Notation Handbook*, York Software Engineering Ltd., 1997