

A PRACTICAL APPROACH TO IMPROVING AND COMMUNICATING ASSURANCE

Jeffrey R. Williams
Arca Systems, Inc.
williams@arca.com

George F. Jelen
G-J Consulting
gjelen@acm.org

Abstract

Assurance in the security of rapidly evolving enterprises depends on a complex set of evidence. This paper describes a method for structuring this body of evidence into a manageable framework called an “assurance argument.” The method is extremely flexible, and is capable of including all relevant claims and evidence. The structure allows the practitioner to compare the costs and benefits of different assurance approaches, to keep track of the rationale for each piece of evidence, and to identify areas where additional evidence is indicated. Further, the approach is modular and allows assurance to be communicated and reused efficiently. Ultimately, assurance arguments enable a better understanding of all the factors that go into creating assurance. This understanding, in turn, plays a major role in the intelligent management of risk.

A PRACTICAL APPROACH TO IMPROVING AND COMMUNICATING ASSURANCE

1 Introduction¹

Assurance is part of every engineering effort. People want confidence in various properties of what has been built. Nowhere is this truer than in the security discipline, where properties like correctness, completeness, and robustness are goals of considerable engineering effort. Some have defined assurance as “the degree of confidence that security needs are met.” [2]. The authors of this paper have previously offered a more specific definition, assurance is “a measure of confidence in the accuracy of a risk or security measurement.” [3].

While the exact definition may be debated, the basic notion of confidence in security properties remains the key to assurance. This paper addresses ways in which this confidence can be generated and how it can be communicated among relevant parties.

Historically, practitioners have relied on security criteria to define the “assurance requirements.” These standards defined the set of evidence that would be sufficient for different ratings under the criteria. Unfortunately, the justifications for these particular evidence requirements have never been clear. This limited practitioners’ ability to make informed decisions about possible alternatives.

It is becoming increasingly difficult to gain and communicate assurance. As the level of interconnection and complexity in products and systems increases, the amount of information required to justify even the simplest claims grows quickly. Simple lists of evidence requirements are not adequate to represent the information required to represent assurance. A more sophisticated structure for communicating assurance related information is indicated.

To help deal with this problem, producers are looking for alternative methods of establishing an increasing range of security relevant properties. Consumers have a related problem, namely, determining the relative importance of the vast amount of information available at all levels of abstraction. Security decision makers try to make sense of all the information, to ensure that the risks associated with the enterprise are tolerable, and to take reasonable precautions where warranted.

1.1 *Purpose of Paper*

The security decision maker depends upon having the right information in an understandable form. This paper suggests a way to prioritize, generate, assemble, and communicate the information needed to make an intelligent decision. This paper is intended to help all types of security practitioners by providing a simple method for structuring security relevant information into a manageable form that allows communication and reasoned judgments.

Frequently, the concepts of risk and security are confused with the notion of assurance. Keeping them separate allows consideration of the interesting cases where the two dimensions lead to seemingly conflicting conclusions. In situations where there is a large amount of uncertainty about the security of a system, simply adding more security mechanisms is not likely to help.

¹ This research was partially supported by the National Security Agency under Contract Number MDA904-97-C-0223. It borrows heavily upon an earlier study by the authors and published as an ARCA report [1].

The goals for the method are to:

- ◆ Apply to the broadest range of security related properties
- ◆ Accommodate the widest range of possible evidence
- ◆ Allow the rationale for each piece of evidence to be maintained
- ◆ Permit cost-benefit comparisons of different assurance alternatives
- ◆ Provide insight into which properties are well established and which need additional support
- ◆ Provide some ability to reuse assurance in components

1.2 Overview

The approach advocated in this paper is that an “argument” is the most efficient way to organize and present assurance information. The first step is to identify the overall goal of the effort by creating a top level claim. This top level claim is then supported by several sub-claims which, taken together, are sufficient to justify the top level claim. Each of these sub-claims is then supported in the same manner. The intent of each successive claim is to support the one preceding. This process is continued and sub-claims are advanced until all claims are sufficiently supported by evidence. The series of claims and sub-claims comprising the assurance argument is intended to form a chain of evidence sufficient to justify the top level claim.

Obviously, there are a huge number of ways to build such an argument and some will be more effective than others. This paper discusses various techniques for making the construction of assurance arguments tractable. In particular, several different ways of structuring arguments are presented and heuristics for building arguments are proposed. Several benefits of using assurance arguments, and ways in which several other assurance related efforts can be reconciled with the concepts, are also presented.

2. Assurance Arguments

Basically, assurance arguments consist of a set of claims based on a logical framework, supported by evidence, and bounded by a set of assumptions. These arguments are nested in the sense that each argument is composed of lower-level supporting arguments and evidence. The cycle of generating lower-level claims and supporting evidence continues until it is reasonable to assume the claim without further evidence.

An assurance argument is simply a structured way of presenting evidence and supporting arguments to decision makers. An assurance argument might be presented in the form of a brochure, fact-sheet, white-paper, security documentation, or certification package. Most products and systems should be accompanied by some sort of assurance argument, although the arguments may vary considerably in their level of detail.

2.1 Assurance argument ingredients

An assurance argument consists of a claim, supporting evidence and subordinate arguments, and some reasoning that establishes the link between the claim and the support [4].

$$\textit{Argument} = \textit{Claim} + \textit{Evidence} + \textit{Supporting Arguments} + \textit{Reasoning}$$

2.2 Claims

Claims are statements that something has a particular attribute or property. These statements are called “claims” because they may or may not be substantiated by evidence. There are a wide variety of ways to phrase claims, but all of them involve a subject and a predicate. The subject is the target of the claim, the thing about which a claim is being made. The predicate is whatever one chooses to say about a subject.

$$\textit{Claim} = \textit{Subject} + \textit{Predicate}$$

2.3 Subjects

Subjects are the things about which one wishes to make a claim. A subject could be an entire enterprise, a network card, an engineering team, a battleship, or an operating system. There are a variety of ways to decompose subjects, the choice among them being situation dependent. However, we have found that a good starting approach for a wide range of applications is to view subjects in terms of their supporting processes, people, technology and environment. Each of these four areas is discussed below in order to show how a subject might be decomposed.

People – Claims about people involve users, administrators, maintenance personnel, security officers, operators, organizations, and anyone else who could affect the security of the enterprise. People who are capable, experienced, knowledgeable, reputable and trustworthy are considered much more likely to perform without introducing security vulnerabilities. Thus, evidence about the education, training, past performance, experience, and background can be very useful in supporting arguments about people.

Process – Claims about processes involve activities that establish, affect, or maintain the security of an enterprise. Examples of processes include clearing users for access to the system, writing software, escorting maintenance personnel, reviewing audit logs, releasing magnetic media, scanning the system for viruses, using the system, administering the system, handling written logs, monitoring the system, system administration, and assessing risk. Processes that are controlled, defined, mature, optimized, and repeatable, are more likely to contribute to better security and lower risk. Evidence that can support process arguments includes process documentation, process metrics, and past performance information.

Environment – Claims about environment involve geographical location (e.g., country, terrain), structural considerations (e.g., doors, windows), the physical setting (e.g., locks, protected network hardware, and locked computer rooms), and the organizational culture. An environment that is controlled, quality-centered, and stable is more likely to reinforce a security focus. Evidence that can establish these properties includes site security plans, physical architectures, blueprints, structural design analysis, and results of physical penetration tests.

Technology – Claims about technology involve the combination of hardware, software, and communications that automates enterprise processes. Examples of technology might include point of sale terminals, ATMs, access control software, encryption devices, networks, file servers, trusted workstations, or word processors. Some useful properties for technology include that it be documented, correct, well designed, fault tolerant, and tested. Relevant evidence might include schematic diagrams, certification reports, architectures, test results, problem reports, and testimonials.

Enterprise – Supporting arguments can also help establish claims for the entire enterprise. The term, “enterprise,” is intended to represent the composite of the people, process, environment, and technology. Many different claims may be made about an enterprise. Some of the most security relevant of these claims are analyzability, correctness, completeness, and strength. Analyzability implies that the enterprise is not

overly complex and is structured such that it can be understood. A correct enterprise indicates that the enterprise accurately performs as specified. Enterprise completeness indicates that all threats to the enterprise are addressed and all the security policies are implemented. A strong enterprise is capable of withstanding attack. Supporting evidence could include corporate security policies; organizational charts; and historical data, including results of previous enterprise-wide risk assessments.

2.4 Predicates

In addition to subjects, claims also contain predicates. Predicates are the things one chooses to say about the subjects. A predicate can contain an attribute or property assigned to the subject by the statement or claim, or it can address some specific threat or vulnerability, such as indicated by the statement, *The environment has been rendered completely safe from fire.*

At the highest level, the claim that one would wish to make about any subject should probably address risk directly, for example, *The risk associated with the subject is less than some value* (basically the amount one is willing to lose). In those cases for which the application is unknown, the claim statement is not able to address risk, but would address likelihood instead. For example, *The likelihood of a loss from this subject is less than some amount.* The difference between these two statements is that the first considers consequence and the second one does not. Since, in the second case, the application is not known, the consequence associated with various events can not be assessed.

In order to determine a value for risk, one must determine consequence, and in order to place a value on consequence, one has to know the mission of the enterprise. Another way to state the difference between the two above statements is that the first is application dependent, whereas the second is not. When the specific application is known, and the mission of the enterprise understood, the first statement would be the more appropriate. In those cases where the subject about which the claim is being made could have many different and unknown applications, such as a security product under development, the second statement is the more appropriate.

One form of lower-level predicates involves specific properties that are combined with a given subject to form a claim. We call these “property predicates.” A property is a characteristic trait relevant to establishing assurance. Table 2 lists some examples of security relevant properties but is not intended to be exhaustive. Not all properties apply to all components of an enterprise. In addition, these properties may be interpreted differently when they are used to describe different things. For example, *strength* takes on a slightly different meaning when it refers to technology than when it refers to an environment.

Claim predicates can also involve the negation of a risk event (a threat-vulnerability pairing). We call these “risk event predicates.” An example might be, *The software has not been subverted.* As with claims involving properties, those involving risk events can be substantiated with positive or negative evidence, or with supporting arguments.

Table 2: Examples of Assurance-Relevant Properties

Properties	Description
Analyzable	Capable of being checked
Attentive	Alert, vigilant, observant, watchful
Capable	Having required or wanted skills or faculty
Complete	Having all the necessary parts or providing a total solution
Consistent	Uniform and steady
Controlled	Kept within defined limits
Correct	Free from error, defect, or fault with respect to a higher level specification
Defined	Described by a fixed set of parameters
Documented	Committed to writing
Easy-to-Use	Capable of being put in service, performed, or maintained without difficulty
Effective	Produces the desired result
Efficient	Performs with a minimum of waste, expense, or unnecessary effort
Evaluated	Tested against a standard
Experienced	Having performed similar events or activities
Fault Tolerant	Tolerant of mistakes or errors
Knowledgeable	Possessing requisite information
Learning	Capable of acquiring knowledge as result of experience
Managed	Operates according to a plan; an organized effort
Mature	Well seasoned; time-tested
Measurable	Capable of having dimensions, quantity, or capacity ascertained
Optimized	Designed and built with efficiency in mind; fine tuned for performance
Predictable	Anticipated, expected, foreseen, prepared for
Profiled	Described in a way prescribed by a standard criteria
Quality Focused	Very concerned about quality issues
Rated	Scored against a standard
Recoverable	Able to be repaired or brought back from harm
Repeatable	Capable of being performed, experienced, or produced again
Reputable	The estimation in which a person or organization is held by the public
Robust	Able to continue; resistant to undesirable change
Scaleable	Scope or granularity can be adjusted to meet changing circumstances
Stable	Unwavering; not subject to excessive variation
Strong	Capable of enduring or being defended
Successful	Possessing a high rate of past success
Tested	Subjected to a regimen of testing
Trustworthy	Deserving of confidence that a responsibility will be fulfilled
Well Understood	Universally comprehended across the entire enterprise

2.6 Evidence

In order to substantiate claims, information or “evidence” that helps to demonstrate their truth, is assembled. Evidence is empirical data on which a judgment or conclusion can be based. Anything that contributes to the believability of a claim can be considered as evidence. Good evidence tends to be measurable, repeatable, and testable. Design analysis results is an example of evidence that helps to support a correctness claim. Other examples of evidence include analysis results, design documentation, or background investigations. Even circumstantial evidence can contribute to the believability of a claim, even though it may not be directly related. For example, the qualifications of the designers would constitute circumstantial evidence of the quality of the design.

It is important to recognize that evidence can be positive or negative, in the sense that it can help negate or confirm a risk. For example, the results of failed tests are important evidence that a risk exists. Either type

of evidence reduces the amount of uncertainty surrounding a risk estimate and thus contributes to added assurance.

Evidence can also be consolidated. For example, the Trusted Product Evaluation Process (TPEP) considers a great deal of evidence and produces a product evaluation rating that summarizes it in a standardized way. This sort of packaging can be extremely helpful in reducing the quantity of evidence presented to consumers.

As with subjects, evidence has properties. Some examples of evidence properties are correctness, completeness, and analyzability. Perhaps the most important property of evidence is its relevance to whatever argument one is attempting to make. It may be 100% correct, complete, totally analyzable, and there may be large amounts of it, but if it has little or no bearing upon the claim that one is trying to make, it is of little use.

Evidence is generally of one of three types: descriptive documentation, analytic results, or historical data. The first type consists of documentation regarding the way a subject is intended to work. Examples of this type of evidence include business process models, architectures, plans, and designs. A second type of evidence is produced by analysis. This type of evidence can be extremely strong; depending upon the degree of objectivity, the method used to evaluate, and skill of the analysts. Some examples of analysis evidence include certification reports and security audits. Finally, the third type of evidence demonstrates how the subject has performed in the past. Examples of this type of evidence include security metrics, financial information, and customer satisfaction indices.

In many cases, a single piece of evidence can support multiple claims. Examples might be a system profile under the Common Criteria, or an assessment and rating of an organization's processes under the System Security Engineering Capability Maturity Model [7]. Both of these constitute a collection of strong statements about their respective subjects.

Evidence can take the form of a proof but generally does not. More often, it requires a judgment on the part of the receiver as to whether it is relevant, credible and sufficient to substantiate the claim. Unfortunately, these qualities are not measurable, and what one person would accept as relevant, credible and sufficient, another might not.

2.7 Reasoning

Simply presenting evidence may not be enough to convince a decision maker that a claim has been established. This is particularly true if the evidence and supporting arguments do not go directly to the claim being made, but only circumstantially support the claim.

Reasoning is a set of statements that ties together the evidence and supporting arguments to establish a claim. An assurance argument is not a mere collection of the evidence to support a claim. For example, imagine having three pieces of evidence related to a claim about a system. The evidence alone may not be convincing to the decision maker. What is missing is some reasoning that shows how the package of evidence supports the claim.

Assurance derives from the reduction of uncertainty surrounding claims. As uncertainty is reduced, assurance increases. A preponderance of evidence does not necessarily establish assurance claims. The evidence must be shown to be relevant, compelling, and cohesive. To create an assurance package, it is necessary to assemble pieces of evidence. This process of matching evidence to claims can be very complicated. It may take many different types of evidence to sufficiently establish a claim.

3. Structuring an Assurance Argument

Assembling all the pieces of evidence at all different levels of abstraction into a logical argument can be quite daunting. The approach advocated here is to structure an assurance argument in a hierarchical manner. We have seen that claim statements form the building blocks of an assurance argument. This section examines the specific ways in which these claim statements can be composed and decomposed.

In general, the composition and decomposition of claims can be achieved in two ways – by subject or by predicate. Subjects can be decomposed relatively simply into people, process, environment, and technology. This decomposition of subjects, we call the subject tree. Predicates, on the other hand, can be decomposed by properties or by risk events to form the predicate tree. Building an assurance argument involves working down through both the subject and predicate trees to construct a hierarchy of claims in which each successive claim supports and adds detail to the one above it.

3.1 *Choosing a top level claim*

The top level claim is the root of an assurance argument. All of the evidence, subclaims, and reasoning go towards establishing confidence in this claim. The subject of this claim should, therefore, be appropriately broad and be as inclusive as possible. The security of this top level subject is the overall goal of the assurance argument. Examples include an enterprise, a product, or a work force. Anything could be the subject of a top-level claim, but the artificial narrowing of this top-level subject, such as addressing only the automated system, should be avoided.

The predicate for a top-level claim should also be extremely broad, to be defined by lower level claims. A broad property, such as “security” or “minimal risk” should be claimed. The subclaims, if complete, will define the scope of this predicate.

3.2 *The subject tree*

Frequently, an enterprise is far too large and complex to deal with directly. Risk analyses, therefore, will usually break the enterprise up into smaller enterprises, e.g., divisions of a company, agencies of a governmental department, colleges or campuses of a university, etc. Each of these component enterprises consists of people, processes, environment, and technology. In addition to enterprises consisting of sub-enterprises, subjects at the same level within an enterprise are made up of component subjects. Technology is made up of products; processes are made up of subprocesses, etc. This decomposition into component enterprises and component subjects can continue for several levels, but at each level, just as at the top enterprise level, we can talk about the components of risk and of assurance at that level. This decomposition through different levels of the subject constitutes the subject tree.

Figure 1 shows how a top-level subject can be decomposed into people, process, environment, and technology for lower level claims. Each of these second-tier subjects could then be supported by evidence or could be viewed as another top-level subject, which can be further decomposed.

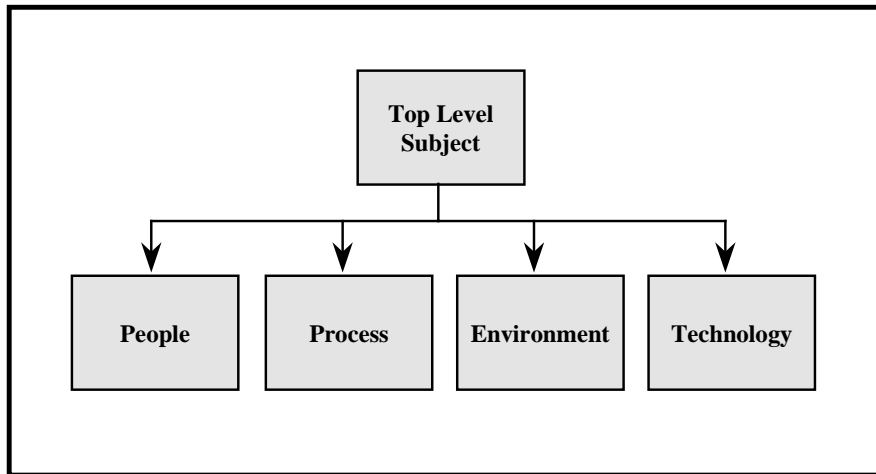


Figure 1: Subject Tree

3.3 Evidence claims

Evidence can also be viewed as a subject of a claim, and can be supported by claims about the people, process, environment, and technology used to develop or analyze the evidence. This is depicted in Figure 2. Evidence claims can be made about the properties of the evidence, which can then be supported by additional argument. For example, evidence of a particular rating from an evaluation process could be supported with some additional claims about the process used, the expertise of the people performing the assessment, and the technology used to support the assessment.

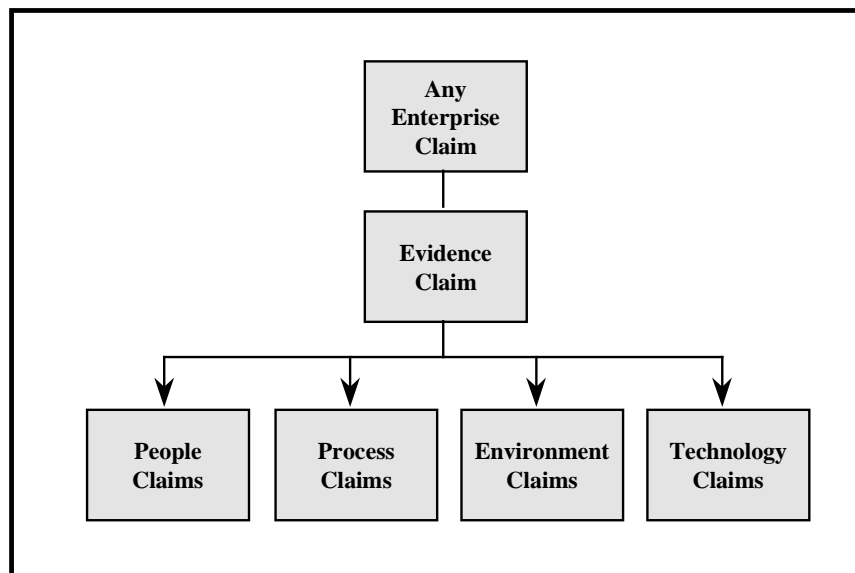


Figure 2: Evidence Claims

Evidence claims are, by their nature, a step removed from the main argument. Nevertheless, if an evidence claim at a high level in an argument can be supported by strong evidence, it may be more important than a lower level claim about the enterprise itself.

3.4 *Subordinate claims*

A claim about a particular subject can be backed by supporting claims about the same subject or by similar claims about the various components that comprise the subject. For example, the claim, *The enterprise has a mature process for verifying and validating security*, could be backed up by the subordinate claims that decompose the predicate:

- ◆ the process employed is defined
- ◆ the process is analyzable
- ◆ the process is measurable
- ◆ the process is complete
- ◆ the process is tested

Alternatively, the same claim could be supported by subordinate claims that decompose the subject:

- ◆ the process for identifying verification and validation targets is mature
- ◆ the process for defining a verification and validation approach is mature
- ◆ the process for performing verification is mature
- ◆ the process for performing validation is mature
- ◆ the process for providing verification and validation results is mature

The first method holds the subject constant and decomposes the predicate; the second holds the predicate constant and decomposes the subject. Although it is possible to decompose both the subject and the predicate in the same decomposition step, this is not recommended.

3.5 *The predicate tree*

In Section 2.2, claims are defined as statements linking subjects with properties or attributes of those subjects. At the highest level, the property might be the association of a value of risk or likelihood of loss to those subjects. At lower levels, the property is usually some attribute like complete, measurable, mature, or robust (see Table 2).

As we have seen, the subject, enterprise, can be decomposed into people, process, environment and technology. Each of these can also be further decomposed into subunits of people, subprocesses, portions of the environment, or components, forming what has been called the subject tree. Similarly, properties can be decomposed to form a predicate tree of subordinate properties. This decomposition of subjects and predicates can be repeated until a level is reached such that the claims can either be adequately supported with evidence or assumed without evidence.

Some claims, although they are backed by quantities of unassailable evidence and sound very positive and of value, can be irrelevant to the risks confronting an enterprise. For example, the claim, A security process is mature, would be irrelevant if the enterprise were staffed with persons paid by the company's competition.

There are two fundamental challenges that confront any security product or system. The first is demonstrating that it does all that is supposed to do, i.e., it performs “at least” the actions stated in claims or specifications. The second is showing that it does not do anything that it is not supposed to do, i.e., it does “no more than” those actions; that is, it must not contain extra functionality or side effects that could be exploited by an attacker [5]. Whereas claims with property predicates can be used to support the “at least” type of requirement or higher level claim; claims with risk event predicates are required to support the “no more than” type. This is because property predicates, are, by their nature, positive statements. “No more than” statements, are, by their nature, negative.

Since the second type is far more difficult to support than the first, it is important that, among the set of nested claims, at least one of them specifically addresses the reduction of the likelihood of some event, which, if not dealt with, would threaten the enterprise. Examples of such claims might be *A system within the enterprise is not affected by power outages*, or *There are no industrial spies on the payroll*. These claims might prove a challenge for the evidence producer, but clearly address the likelihood of unwanted events – the first by negating a vulnerability, the second by denying a threat.

Such claims could be substantiated with sub-claims and might themselves be sub-claims to higher level or more sweeping claims. For example, the claim, *There are no industrial spies on the payroll*, might be supported by sub-claims, backed by evidence, that *The backgrounds and financial interests of all employees are carefully checked every six to nine months*, that *All workplaces are under constant video surveillance*, and that *A cash award of \$100,000 awaits any employee who is able to supply evidence of another employee’s industrial espionage*. At the same time, the same claim, *There are no industrial spies on the payroll*, helps to support the higher claim that *Key industrial secrets are adequately protected from disclosure to competing companies*.

Predicates can take several forms. In addition to previously mentioned “property predicates” and “risk event predicates” (see Section 2.4), predicates can address event components, i.e., threats or vulnerabilities, or they can address consequences that result from events. Consider the event: *a collapse of the roof of the main enterprise building due to an accumulation of snow*. The risk event claim statement might be: *The average risk (or expected loss) to the enterprise from a snow-caused roof collapse is less than \$100 per year*. This claim statement could be supported by the following set of sub-claims:

- ◆ The roof is capable of withstanding a load of 250 pounds per square foot without collapse (vulnerability predicate).
- ◆ The probability that the accumulation of snow on the roof at any time during a given year exceeds 250 pounds per square foot is 0.001 (threat predicate).
- ◆ The total loss to the enterprise that could result from a roof collapse is less than \$100,000 (consequence predicate).

Claims statements can be either quantitative, as in the examples above, or qualitative, e.g., *It is highly unlikely that the snow accumulation will ever exceed the load bearing capacity of the roof*.

Figure 3 is a state transition diagram depicting the process for decomposing a predicate for use in lower level claims. Starting with an event or property predicate, one can follow the arrows to find ways of structuring supporting claims. Each level of the argument should use only one type of predicate at a time. Continuing this process, each level of the argument can be structured according to a particular type of predicate. Each argument should, at some point, include an event predicate, to ensure that the argument is actually relevant to security and not merely an interesting but unrelated assertion.

Threat predicates are predicates concerning the likelihood of a particular threat occurring. As shown in Figure 3, once an argument relies on a set of claims which use threat as a predicate, there is no way to then

support those claims by making claims with property, vulnerability, or risk event predicates. The only way to break down threat predicates further is by subject.

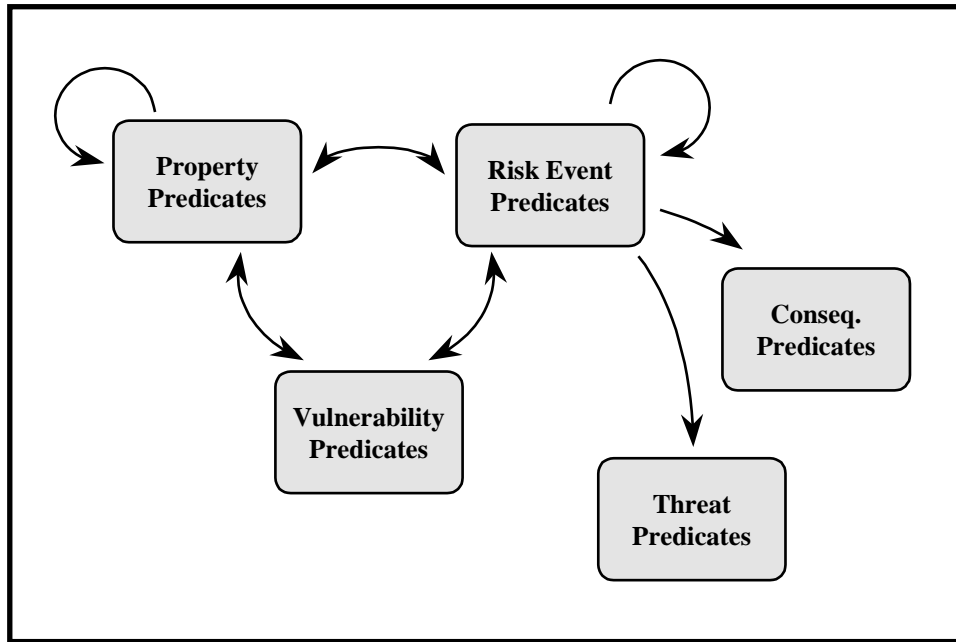


Figure 3: Generalized Predicate Decomposition

Similarly, consequence predicates are predicates that target the severity or impact of a particular event. Consequence statements take the form, *The impact (or consequence) to a given subject resulting from a certain event would be 'X'*. Although, theoretically, such a statement could be developed for any subject within an enterprise, it normally only makes practical sense to make such a statement for the entire enterprise. Thus, we will assume that consequence statements will be of the form, *The impact to the enterprise resulting from a certain event would be 'X'*. The value to be assigned to “X” would be developed directly, based upon available data or evidence. It is generally not helpful to decompose the statement either through a decomposition of enterprise or through a decomposition of the event.

So, how does an enterprise achieve a better estimate of the impact of some untoward event on itself? Certainly, much of an organization’s ability to estimate this derives from its own experiences. Additional relevant “evidence” can be provided from the testimony of other similar enterprises that have suffered a similar event. And finally, consequence analysis is usually very amenable to modeling and simulation methods. Many “what if” scenarios, i.e., “events” in the parlance of this paper, can be modeled and their effects observed.

3.6 Structuring principles

The process of composing and decomposing subjects and predicates will be difficult in practice, since there are many possible ways to create a successful argument. For example, one could decide to structure an argument around the components of a system or, alternatively, to structure it around the high level risks to that system. While the structure of any particular argument will be heavily dependent on the details of the enterprise, several principles may guide the argument building process.

First, to avoid unnecessary redundancy, each of the subclaims should be as independent as possible. One way to achieve this is to hold either the subject or the predicate constant as one proceeds from one level in the argument to the one below.

Second, the number of subclaims should be neither too many nor too few. Given two otherwise equally attractive ways of decomposing a claim, the alternative with a manageable number of subclaims will make the argument easier to comprehend. In general, three to five subclaims make for a nicely structured argument. Fewer subclaims will result in an extremely “deep” argument which is difficult to follow. Similarly, more subclaims will make the argument so broad that it becomes impossible to keep track of all the relevant factors.

Another design principle for building good assurance arguments is that, in so far as possible, subjects and predicates should be at a similar level of abstraction. If high-level subjects are assigned low-level predicates, the resulting claim is difficult to understand. For example, the claim, *The enterprise is not susceptible to short circuits*, could be improved by narrowing the breadth of the subject to only the automated system. Similarly, low-level subjects with high-level predicates are also difficult to handle. For instance, the claim, *The hiring process is secure*, could be improved by narrowing the predicate to specify a particular attribute of the hiring process, such as “repeatable.”

Each level of the argument must be relevant to the claims above. This principle helps to ensure that effort is not wasted on sophisticated arguments that do not ultimately relate to the overall risk. Checking the relevance of each subclaim is a useful exercise as it helps to increase the understanding of the importance of the argument to overall assurance.

Finally, if the structure of the available information suggests a particular decomposition, following that structure in the argument is likely to be easier than attempting to restructure it. This is simply a pragmatic suggestion that takes advantage of the existing information and its structure. For example, if an organization has achieved SSE-CMM Level Three in some process area, implying well-defined security practices, it may have assembled a considerable amount of information organized according to the process area that produced it. In this case, it may make sense to structure the claims by process area. The evidence will then naturally support the claims, and the argument will be easier to understand, which is the goal.

3.7 Where to stop

As claims are successively backed by evidence and subordinate claims, eventually it is possible to reach a point where the evidence is unassailable or the claims are simply accepted without further justification. However, it is not always efficient to push the argument to this point.

It may be more efficient instead to focus effort on the parts of the argument that are critical to establishing the important properties. The assurance argument structure facilitates this by allowing the practitioner to identify the evidence that is critical to the claims being made and which is helpful but non-critical.

4. Relevance and Utility of Arguments

This section addresses a natural relationship between the approach described in this paper and three different assurance related security efforts—the Network Rating Methodology, the System Security Engineering Capability Maturity Model, and the Common Criteria. By describing the types of claims that each of these efforts can support, we hope to start the discussion of how these different approaches to gaining assurance might be compared and combined.

4.1 Network Rating Methodology

The “matrix” approach adopted by the Network Rating Methodology (NRM) is, although rigidly defined, quite consistent with the assurance argument structure presented here [6]. Expressing the NRM in terms used in this paper, the top level claim might be, *The security provided by the network is “good enough.”*

This claim is broken down into sixteen matrix cells representing predefined subclaims. These subclaims result from breaking the top-level subject (the network) into four areas and the top-level predicate, “good enough,” into four areas. The specific subject areas of the NRM are personnel, operational procedures, physical environment, and technology, which equate rather directly to the people, process, environment, and technology subjects described above. The predicates of concern within the NRM paradigm are confidentiality, integrity, availability, and authenticity. The approach outlined in this paper, therefore, can easily accommodate the full repertoire of NRM-relevant claims.

4.2 System Security Engineering Capability Maturity Model

The System Security Engineering Capability Maturity Model (SSE-CMM) “describes the essential characteristics of an organization’s security engineering process that must exist to ensure good security engineering” [7]. Among its objectives is to generate confidence based on the maturity of processes used. The Model is intended to enable what it calls “capability-based assurance,” which it defines as “trustworthiness based on confidence in the maturity of an engineering group’s security practices and processes.” A specific claim that could be based on SSE-CMM evidence is, *The security engineering organization’s process is mature.*

This claim’s subject, the process, is broken down by the SSE-CMM into ten security-relevant process areas: Specify Security Needs, Verify and Validate Security, Provide Security Input, Assess Threat, Assess Vulnerability, Assess Impact, Assess Operational Security Risk, Build Assurance Argument, Monitor System Security Posture, Administer Security Controls, and Coordinate Security.

The claim’s predicate property, “maturity,” is broken down into various properties related to maturity, including planned, tracked, defined, coordinated, measured, controlled, and improving. The model also discusses evidence for supporting these attributes, including the appraisal method itself. Thus, a hierarchy of subclaims can be constructed according to the methods described in Section 3.

These subclaims can then be used to support a wide variety of higher level claims about an enterprise’s security. For example, a claim that a component is free from defects can be supported by an SSE-CMM subclaim that validation process used in building the component was well-defined. Similarly, a claim that a component is resistant to penetration can be supported by the subclaim that the development process is planned and tracked.

4.3 Common Criteria

The Common Criteria (CC) “represents the outcome of efforts to develop criteria for evaluation of IT security that are widely useful within the international community. It is an alignment and development of a number of source criteria: the existing European, US, and Canadian criteria (ITSEC, TCSEC and CTCPEC respectively)” [8]. It represents a major step toward an internationally recognized standard and to worldwide mutual recognition of evaluation results. Version 1.0 of the CC was published for comment in January 1996; Version 2.0 was published in May 1998; and a mutual recognition document was signed in October 1998.

A claim that can be based on the CC is, *The information technology has a specific set of security features and assurance evidence.*

The top level subject, “information technology,” can be decomposed as described above. However, the predicate, has a set of security features and assurance evidence, is more difficult to break down. Ideally, claims about features should be tied to the particular event that the feature is intended to counter. Similarly, claims about evidence should also be linked to the problem that the evidence addresses. Building the infrastructure that links these claims back to a top-level claim will involve difficult questions about the nature of the risk and the feature or evidence.

Summary

As we move into the next millennium, systems and enterprises will continue to become more complex. New techniques are required for managing the increasing amount of evidence necessary to achieve any measure of assurance. We have presented the “assurance argument” as just such a technique.

Assurance arguments allow practitioners to keep track of the reason why each piece of evidence is required and what it establishes. This understanding facilitates informed reasoning about the costs and benefits of the various alternative types of evidence available. It also facilitates communication of confidence in an easily understood manner.

Ultimately, a better understanding of assurance leads to better risk management decisions.

Acknowledgments

No paper is the sole product of its authors: this one is the result of many discussions with many people over several years, whose contribution we gratefully acknowledge. We also wish to acknowledge and thank the National Security Agency without whose funding support this paper would never have been written.

References

- [1] Williams, Jeffrey R. and George F. Jelen. *A Framework for Reasoning about Assurance*, Document Number ATR 97043. Arca Systems, Inc. 23 April 1998.
- [2] National Institute of Standards and Technology. Invitational Workshop on Information Technology (IT) Assurance and Trustworthiness. March 1995.
- [3] Jelen, George F. and Jeffrey R. Williams. “A Practical Approach to Measuring Assurance.” *Proceeding of the 14th Annual Computer Security Applications Conference*. Los Alamitos, CA: IEEE Computer Society, 1998.
- [4] Toulmin, Stephen. *The Uses of Argument*. Cambridge University Press, 1958.
- [5] Boebert, W.E. “Assurance Evidence.” Technical Report Contract 1021-02-91. Secure Computing Technology Corporation. 1 June 1992.
- [6] “The Network Rating Methodology: a Framework for Assessing Network Security.” 11 September 1997.
- [7] Systems Security Engineering Capability Maturity Model. *Model Description, Version 2.0 beta*. 5 October 1998.
- [8] Syntegra (on behalf of the Common Criteria Implementation Board). *Common Criteria: An Introduction*.