Attack Tree-based Threat Risk Analysis

by Terrance R Ingoldsby
Amenaza Technologies Limited

Amenaza®, SecurITree®, as well as the  and  symbols are registered trademarks of Amenaza Technologies Limited

Amenaza Technologies Limited
406 – 917 85th St SW, m/s 125
Calgary, Alberta T3H 5Z9
Canada

1-888-949-9797 toll free US & Canada
+01 403 263 7737 International

E-mail: Terry.Ingoldsby@amenaza.com
Web: www.amenaza.com

Table of Contents

List of Figures

Attack Tree-based Threat Risk Analysis

Introduction

Risk analysis is as old as civilization itself. People quickly learn that there are pros and cons to every choice they make. Repeated observations and experiences lead to an intuitive sense of risk in a given situation. Unfortunately, there are limits to intuition's ability to cope with changing variables or unfamiliar conditions. Reliance on intuition in these situations often leads to poor decisions.

Nowhere is this more true than in the field of hostile risk analysis. Modern civilizations provide an unprecedented level of safety and security to their citizens. Even people working in the security field often have limited first hand experience in fending off attacks. As a result, a variety of methodologies have been developed to help analyze the risks from hostile threats. Unfortunately, many of these systems are based on simple checklists which are overly general in nature. Other approaches are highly subjective and fail to capture the logic behind the analysis. **Attack tree models are a more rigorous, engineering-like approach to hostile threat risk analysis.**

The techniques of attack tree analysis have been known by expert practitioners for over twenty years. A number of papers have been published on the subject. However, there seem to be few publicly available documents that provide comprehensive coverage from basic principles to advanced techniques. This paper attempts to fill that gap.

One glaring problem with many existing hostile risk analysis strategies is that they focus exclusively on the *system* the defender is trying to protect. Knowledge of the characteristics of both the defender's system, and the adversaries that threaten it, allows an understanding of the interplay between the two entities. This greatly improves risk estimation. Accordingly, the techniques described in this document emphasize the roles of both defenders and adversaries.

Attack trees are models of reality. They are a simplified representation of complex real world drivers. The accuracy with which the underlying drivers are known depends on many factors including the time and effort spent studying them. In some cases it becomes necessary to make assumptions based on the best information available. Of course, the accuracy of the analysis will be limited by the correctness of the assumptions. **All models, including attack trees, will break down if they are used beyond their limits.** The conclusions reached by any risk estimation scheme (including attack trees) should be subjected to a reality check and compared to the results from other methodologies.

Despite this note of caution, it should be noted that all predictive mechanisms depend on assumptions. It is a serious problem when analysts begin to treat their assumptions as facts and are surprised (sometimes disastrously) that their conclusions are wrong. Attack trees provide a discipline for declaring and understanding assumptions. Exposing assumptions to review and critique makes unpleasant surprises less likely.

Hostile risk analysis is not the first risk discipline to use tree structures. Fault (or failure) trees

have long[1] been used to understand how component failures affect overall system function. Fault trees are useful for understanding any type of random risk, including incidents caused by Mother Nature, human error and equipment failure. This paper explores mechanisms for merging both hostile and random risks into an integrated tree-based model.

One of the most significant differences between attack tree analysis and some other hostile risk analysis methods is that attack trees are built largely from the point of view of the attacker (instead of the defender). Attack tree models excel at estimating the risk for situations where events happen infrequently or have never happened before.

Security practitioners have always found it challenging to provide convincing evidence that the countermeasures they deploy are responsible for preventing a attacks. It is fundamentally difficult to provide conclusive proof of why an event doesn't happen. This problem is exacerbated further when dealing with unprecedented or infrequent events. In this case, statistics (which are based on a representative sample of events) cannot demonstrate that any risk exists. Nonetheless, as the 9/11 tragedy sadly proved, the absence of a statistical precedent does not provide any assurance that the risk is zero. The adversary may choose to create novel events precisely because the victim is unprepared for them – leading to exceptionally devastating results.

One might reasonably ask whether it would be possible to compare the results predicted by the attack tree methodology to the frequency of common hostile events (for which statistics are readily available). This is not as easy as it might appear. Attack tree analysis incorporates information about a specific defender's adversaries and the benefits they will realize from carrying out an attack against a particular defender. This precision is a virtue because it offers the hope that predictions will be accurate for a given situation. However, this specificity also makes it difficult to compare defender-specific predictions with statistics that are generalized over a wide variety of defenders and attackers. For example, consider two technically identical security systems. The risks associated with a particular attack may differ considerably between the systems because the assets they are protecting also differ. The different potential rewards may attract different adversaries with different skills, resources and motivations. Technically identical failures can have significantly different business impacts on their respective organizations (which also affects the risk).

Despite the caveats mentioned above, the widespread and increasing usage of attack trees by aerospace, electric power, defense and intelligence organizations demonstrates the confidence that they place in the technique. They are just as applicable in less esoteric applications and are becoming more commonly used in commercial, medical and critical infrastructure fields.

Many of the diagrams in this paper are screen shots from a commercial attack tree software tool called Secur*IT*ree®. Secur*IT*ree, a commercial product of Amenaza Technologies Limited, has been designed to implement the modeling functions described in this paper.

---

[1] Fault trees were invented in the early 1960s for use in the Minuteman Missile System. Clifton A Ericson II; Fault Tree Analysis – A History from the Proceedings of the 17th International System Safety Conference, 1999.

Basic Attack Tree Concepts

Attack Tree Origins

Attacks can be modeled using a graphical, mathematical, decision tree structure called an *attack tree.* There is reason to believe that *attack trees* originated in the intelligence community[2]. At least one intelligence agency is believed to have used tree-based attack modeling techniques in the late 1980s. In 1991 Weiss published a paper[3] describing *threat logic trees.* In 1994 Amoroso[4] detailed a modeling concept he called *threat trees.* More recently, Bruce Schneier[5] (a noted cryptographer and security expert) popularized the idea, although he called it *attack trees*. Other researchers have continued to develop the idea of tree-based, threat analysis models [6, 7].

Prerequisites of an Attack

Three conditions must be present in order for an attacker (also known as a *threat agent*) to carry out an attack against a defender's system.

1. The defender must have **vulnerabilities** or weaknesses in their system.
2. The threat agent must have sufficient **resources** available to exploit the defender's vulnerabilities. This is known as **capability.**
3. The threat agent must believe they will **benefit** by performing the attack. The expectation of benefit drives **motivation.**

Condition 1 is completely dependent on the defender.

Whether condition 2 is satisfied depends on both the defender and the threat agent. The defender

---

[2] In the 1998 paper by C.Salter, O.S. Saydjari, B. Schneier and J. Wallner, Toward a secure system engineering methodology, Proceedings of the New Security Paradigms Workshop, ACM Press, September 1998, two of the authors are shown as working for the National Security Agency and a third for DARPA.

[3] J.D. Weiss, A System Security Engineering Process, Proceedings of the 14th National Computer Security Conference, 1991.

[4] Edward G. Amoroso, Fundamentals of Computer Security Technology, pp 15-29, Prentice-Hall, 1994, ISBN0131089293

[5] B. Schneier, Attack Trees, Dr. Dobb's Journal, v. 24, n. 12, December 1999, pp. 21-29.
B. Schneier, Attack Trees: Modeling Actual Threats, SANS Network Security 99 – The Fifth Annual Conference on UNIX and NT Network Security, New Orleans, Louisiana. Wednesday, October 6th, 1999, Session Two, Track One - Invited Talks
B. Schneier, Seminar session given at a Computer Security Institute conference in November, 1997. See also http://www.counterpane.com/attacktrees.pdf

[6] Moore, A., Ellison, R. and R. Linger, "Attack Modeling for Information Security and Survivability", March 2001, http://www.cert.org/archive/pdf/01tn001.pdf

[7] Shelby Evans, David Heinbuch, Elizabeth Kyle, John Piorkowski, James Wallner, "Risk-Based Systems Security Engineering: Stopping Attacks with Intention", November/December 2004, IEEE Security and Privacy

has some influence over which vulnerabilities exist and what level of resources will be required to exploit them. Different threat agents have different capabilities.

Condition 3 mostly involves the attacker. It represents the motivation to carry out the attack. The defender may have a role if their actions provoke a threat agent to carry out an attack.

The threat agent and the defender interact to jointly determine whether an attack occurs. Proper attack analysis requires that we examine all three conditions in order to predict the behavior of adversaries and the likelihood that an attack will occur. Understanding these factors also provides insight into effective ways of preventing attacks.

Attack Tree Vulnerability Models

Attack trees are constructed from the point of view of the adversary. Creating good attack trees requires that we *think like an attacker*. We do not focus on how to defend a system when we initially create the model. Instead, we think of what an attacker wants to achieve and ways to accomplish it. Later, we use the understanding we have gained of the system's vulnerabilities to improve its defenses.

Like most mathematical tree models, attack trees are represented by a diagram with a single *root* node at the top. The root branches downwards, expanding through forks and more branches. This is similar to the *decision trees* used to help with business decisions or the *fault trees* used to understand the reliability of machines and machine-like processes.

In an attack tree vulnerability model, the topmost (*root*) node represents an objective that would be of benefit to one or more threat agents. Unfortunately, accomplishment of the root goal usually brings negative consequences to the defender[8]. If the goal is chosen carefully it is usually possible to analyze a system completely with a single attack tree. In some situations a particular adversary may have several different goals, or different adversaries may have their own unique goals. These situations occasionally require multiple attack trees to carry out a complete analysis.

By itself, the root goal is so lofty or broadly stated that it lends little understanding as to how it might be achieved. It is helpful to break the high level root goal into smaller, more manageable steps. This allows a number of different strategies to be formulated that could achieve the overall goal. These strategies can be expressed as a series of intermediate objectives that singly, or in combination, realize the root goal. This decomposition process continues, breaking the intermediate goals into ever finer grained activities. This is conveniently represented using a graphical format (see **Figure 1**).

[8] If the defender suffers no negative consequences from an attack, there is no reason to spend effort to prevent it.

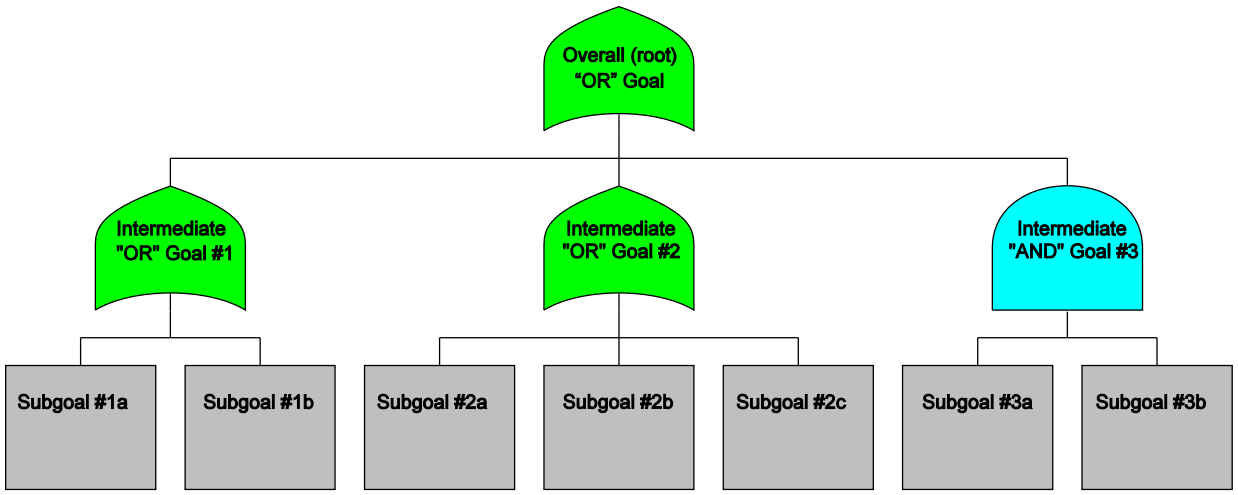


Figure 3 – Goal Oriented Tree

The topmost symbol in the tree represents the adversary's overall goal. It is referred to as the *root* of the tree. The *root* in this particular example is depicted by a green, pointy-topped symbol . The diagram shows how high level goals decompose into increasingly precise subgoals as we descend through the tree.

The *OR* symbol (whose shape should be familiar to readers familiar with Boolean algebra) indicates that the root *Overall "OR" Goal* can be attained by achieving *Intermediate Goal #1* ***OR*** *Intermediate Goal #2* ***OR*** *Intermediate Goal #3.* The children of the ***OR*** nodes represent the alternative ways in which the ***OR*** subgoal can be realized.

In this example, the ***OR*** nodes in the figure are further decomposed into rectangular shapes, called *leaf* nodes. For example, *Intermediate "OR" Goal #1* is achievable by attaining *Subgoal #1a* ***OR*** *Subgoal #1b. Leaf* nodes represent atomic activities which require no further decomposition to be understood. They represent activities that could be performed by an attacker.

Intermediate Goal #3 is represented by a cyan *AND* symbol . This indicates that both *Subgoal #3a* ***AND*** *Subgoal #3b* must be completed in order to attain *Intermediate Goal #3*. The children of ***AND*** nodes represent a series of steps in a process or procedure that must be performed in order to attain or satisfy the ***AND*** node. Strictly speaking, the order of the ***AND***'s children has no significance. However, a useful convention is that, if order is important to the attainment of the ***AND*** node, then the children are arranged in stepwise order from left to right.

An example will make this clearer.

A Sample Attack Tree

To illustrate the concept of a **capabilities-based attack tree**, let us imagine a hypothetical system we are trying to defend. Consider the home security challenge faced by the residents of a typical, suburban home who are concerned about possible home burglaries. The subject of the example was chosen for its simplicity and familiarity to readers. More interesting examples,

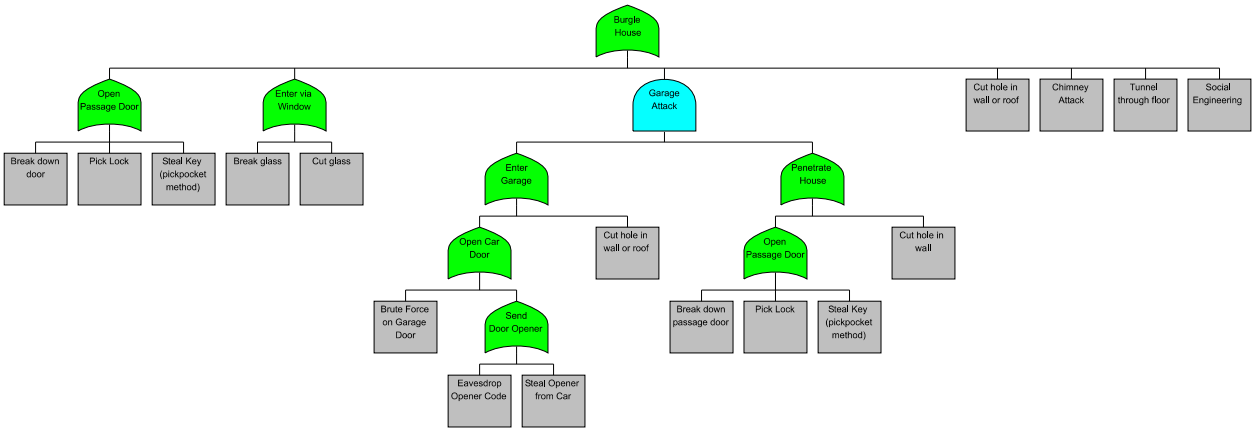


Figure 4 – Approaches to Burglarizing a House

particularly those involving information systems, are frequently too large and complex to fit on a single page.

The house we have in mind is a middle-class dwelling, complete with an attached garage. The issue that concerns us is the possibility of the house being burglarized (see **Figure 2**).

After some consideration, we can think of seven approaches the thief might use to enter the house and commit burglary:

1. Passage doors (i.e., the front and back doors normally used for entry).
2. Windows.
3. Attached garage.
4. Walls (including the roof – it is essentially an angled wall).
5. Chimney.
6. Floor (attacking from beneath).
7. Social engineering (convince the resident to allow entry to the attacker).

These attacks, which have been partially decomposed into more detailed steps, are shown graphically in **Figure 2**. To simplify our example, we have restricted the decomposition to the *Open Front/Back Door*, *Enter via Window* and *Garage* attack vectors. Obviously, greater detail could also be added to the *Cut Hole in Wall or Roof, Chimney Attack, Tunnel through Floor* and *Social Engineering* attacks.

As can be seen in the diagram, there are three types of passage door attacks. The doors can be physically broken, the locks can be picked or the key can be obtained through theft. Similarly, an

intruder can either cut or break the glass in the windows. To enter via the garage, the burglar must first gain entry to the garage and then enter the house (either through the wall or by penetrating the passage door leading from the garage to the house).

Decomposition of higher level events into smaller, more precisely defined events could continue almost indefinitely. For our purposes, it need only continue to the point where further decomposition will not increase the understanding of the intended viewers of the model. For example, the *Break glass* leaf node could be decomposed into the steps of picking up a rock and throwing it at the window. This is unnecessary since almost everyone knows how to break a window using a rock. On the other hand, the leaf node that deals with *Eavesdrop opener code* ought to be decomposed into smaller steps to enhance the analyst's understanding of the actions to be performed by the burglar. We have not done so for reasons of brevity.

It is important to note that the adversaries' interaction with the system they are attacking takes place entirely at the leaf nodes. For that reason, some people call the leaf nodes *attack stabs* or *exploits*. All of the higher, non-leaf nodes in an attack tree represent logical states that the attacker achieves through their efforts in performing leaf node operations.

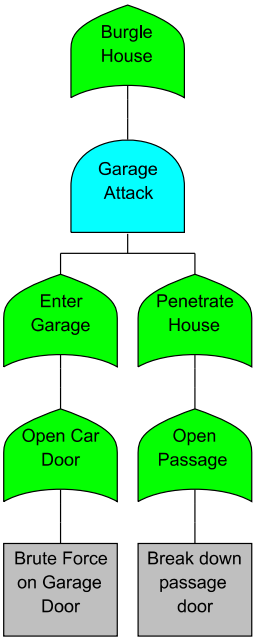


Figure 5 – Attack Scenario Example

Attack Scenarios

An attack tree shows a logical breakdown of the various options available to an adversary. By performing the exploits associated with one or more leaf level events which have been carefully selected to satisfy the tree's AND/OR logic, the attacker can achieve the root level goal. Each minimal combination of leaf level events is known as an *attack scenario*. The combination is minimal in the sense that, if any of the leaf events are omitted from the *attack scenario*, then the root goal will not be achieved.

Associated with each *attack scenario's* set of leaf nodes is the collection of intermediate nodes that are activated along the path (or paths) to the root goal. Strictly speaking, these intermediate nodes are not part of the *attack scenario,* but it is useful to include them in graphical depictions to illustrate the logical states that will be achieved as the attack takes place. As will be seen shortly, negative impacts are often felt by the victim, and positive rewards by the attacker, when the intermediate AND/OR states are achieved.

The complete set of attack scenarios for an attack tree shows all of the attacks that are available to an attacker who possesses infinite resources, capabilities and motivations. One particular attack scenario from the house burglary tree is shown in **Figure 3**. It consists of two leaf level events: *Brute Force on Garage Door* and *Break down passage door*. Both events are required to satisfy the AND node (*Garage Attack*) several levels above.

Behavioral Indicators

To this point, the attack tree shows how attacks could occur, but provides no indication of their likelihood. Intuitively we know that a burglar will choose to break a window rather than digging

a tunnel underground. We suspect this is because it is easier to break windows than to dig tunnels. It seems reasonable to suppose that an attack's level of difficulty affects the behavior of an adversary.

Since all of the direct interaction between the adversaries and the defender's system occurs at the leaf nodes, it is useful to associate metrics with each leaf node operation describing the resources required of the adversary. The types of resources examined are chosen to be factors that influence the behavior of the adversary. For instance, the cost, technical ability, time and noticeability of an exploit are all potential obstacles. Values for these parameters are obtained from subject matter experts (SMEs) who provide estimates based on their expert understanding of the activities.

Attack Scenario Costs

The costs associated with a particular attack scenario (which may involve several leaf level activities) can be calculated by examining the resource requirements of the scenario's leaf level activities. If the exploit consumes a resource then the total requirement for that resource is the sum of the scenario's leaf nodes' resource metrics. If the resource can be reused (such as is the case with technical ability) then the resource cost is the maximum of the scenario's leaf nodes' resource metrics. Other aggregation formulas are conceivable for special cases.

Pruning – A Simple Way to Eliminate Unlikely Attack Scenarios

Whether or not a system's defenses are adequate to thwart an attack depends partly on the capability of the adversary. If an attacker has sufficient resources to perform all of the exploits required for a particular attack then the scenario is possible. If the adversary also has the desire or motivation to carry out the attack, then the attack is probable.

A simple way for evaluating the feasibility of a given adversary performing an attack scenario is to compare the resources available to them with the scenario's behavioral indicator costs. Those scenarios with resource requirements greater than the adversary's capabilities can be safely eliminated from consideration (since it is not possible for that adversary to provide them). The attacks that remain are feasible and, depending whether they are desirable to the *threat agent*, have some, non-zero level of probability. This process is known as *pruning*[9].

For instance, a typical juvenile delinquent might only have $50 available to spend on attacks, and possess limited technical skills. The cost and technical difficulty of digging a tunnel underground would eliminate the tunneling scenario from consideration by juvenile delinquents.

Pruning provides a defender with a quick estimate of the magnitude of their security problem by eliminating from consideration those attack scenarios that are beyond the capability of a particular threat agent. Unfortunately *pruning* is overly simplistic. It treats each of the attacker's resources in isolation whereas it is more realistic that an adversary would consider the combined cost of all of the resources required for an attack. Also, the amount of resources the adversary is willing to spend depends, in part, to the extent that an attack scenario satisfies their goals and

[9] See also, Computer Security Journal, Volume XX, Number 2, Spring 2004 pp 33 - 59.

ambitions. The effects of varying degrees of motivation are not captured by pruning.

Risk Requires Impact

Even assuming that all feasible attacks (identified through pruning) have some non-zero probability, that is still only half of the risk equation. Hostile risk is generally accepted to be the combination of two factors:

$$Attack\ Risk \equiv Attack\ Probability \times Victim\ Impact$$

In order to fully understand risk, our model needs to include the impact each attack scenario will have on the defender. This can be achieved by a simple extension to the attack tree model.

Impact can occur at any level in the tree. Although some victim impacts may potentially occur when an attacker performs an exploit (at a leaf node), the impacts generally become greater at higher levels in the tree. For example, businesses are often relatively unaffected by the minor damage to computer systems that is the immediate outcome of a hacker's activities. However they may experience serious or catastrophic losses due to the indirect consequences as business systems become unavailable or information is disclosed.

The pruning strategy can accommodate impact. If you accept the simplification that all attacks remaining after pruning are of comparable probability, then for the scenarios that remain after pruning, the risk equation can be simplified to

$$Attack\ Risk \propto Attack\ Impact$$

However, even if you agree with this questionable approximation, there is no obvious process for deriving a single value for an attack scenario's impact term if the model incorporates multiple victim impacts (e.g., loss of money, deaths, damage to the environment).

Clearly, our model must become more sophisticated to properly deal with the shortcomings we have identified.

Analysis based on Perceived Value

Every day, people (good and bad) are faced with choices and consequences. It is our hypothesis that people generally select one activity over another because they believe that it has a superior cost-benefit[10] ratio to the competing alternatives. However, it is not enough to analyze the raw costs associated with their choices. Our models must reflect the fact that different people (whether attackers or defenders) perceive different values for the same amount of the same commodity.

Attacker Behavior

In order to understand attacker behavior methodically we need to quantify the decision making

[10] Although common vernacular speaks of the cost-benefit ratio, generally it is calculated as $\frac{Benefits}{Costs}$. The greater the benefits (and the lower the costs) the higher the value. Costs and benefits do not have to be monetary.

factors used by attackers.

Ease of Attack

Every attack requires the adversary to expend a variety of resources. The analyst chooses specific types of resources to include in the attack tree model based on the degree to which they influence the adversary's ability to perform the various attack scenarios. These resources can include money, raw materials, talent, time and a willingness to be noticed.

Even though everyone might be forced to spend the same amount of a resource to perform a specific attack, that does not mean that they are equally willing or able to do so. The availability of resources varies. For instance, a relatively poor juvenile computer hacker might consider $100 to be of considerable value and be strongly disinclined to part with it without a substantial benefit. On the other hand, a busy executive in a large company might regard $100 as pocket change. However, the time-crunched white collar worker would be far less willing to part with 25 hours of his or her precious time than the bored adolescent who is happy to while away the wee hours trying to crack a computer system.

The simple pruning mechanism described earlier provided a crude representation of the affinity of threat agents to their resources. For example, suppose the analyst created a profile of the juvenile delinquent threat agent that specified a financial limit of $50. This simplistic profile asserts that the juvenile delinquent is completely and equally willing to spend any sum between $0 and $50, but that they would be utterly unable or unwilling to spend $51 (as shown in **Figure 4**).

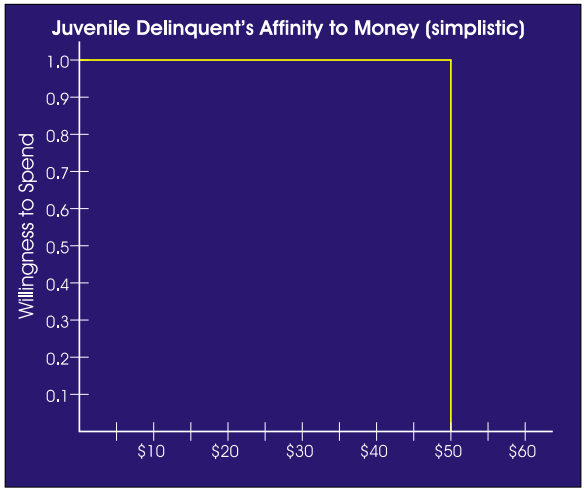


Figure 6 – Pruning-based agent profile

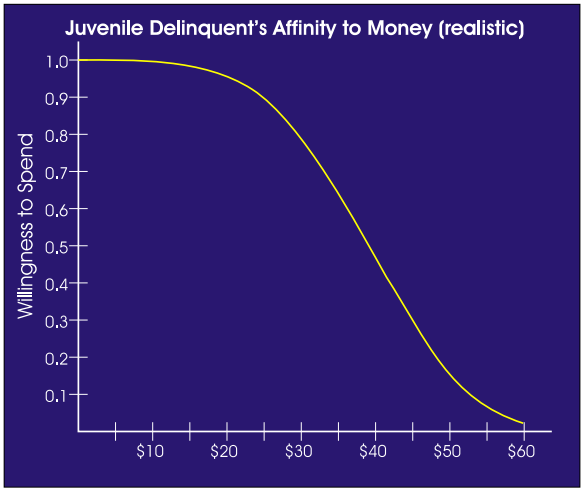


Figure 7 – Value-based agent profile

While we are unaware of any research into the spending habits of juvenile delinquents that would support the exact curve shown[11] in **Figure 5**, it is more plausible than **Figure 4**. Basic

[11] Indeed, a perfectly accurate curve may be unattainable given that there will be variations in the behavior of individuals within the threat agent class of *juvenile delinquent* and variations in a particular individual's behavior from one day to the next..

economics dictates that there is some scarcity to almost every asset, and a corresponding reluctance to spend all of it. We find numerous examples where people's willingness to spend decreases gradually (and monotonically), but very few situations where willingness is binary. In general, people are always well disposed to spend none of a resource to acquire a desirable commodity. This is shown in **Figure 5** by the y-intercept point (1.0) on the graph. No one is able to spend more than they possess (no matter how attractive the goal may be). The limit of their resource is the x-intercept.

We call functions that map a threat agent's attachment to quantities of the commodities required to perform an attack, *resource affinity utility functions*. The domain (x-axis value) of these functions is the resource under consideration. For convenience we establish the convention that the range (output) of the functions will be from 0 to 1.

Resource affinity utility functions can map the value of any commodity. For example, a utility function could be created to map from the domain of raw technical ability rating values (arbitrarily chosen to span from 1-100) to an output range of 0 to 1 (**Figure 6**).

In general, it is the combination of costs associated with a task that creates an overall perception

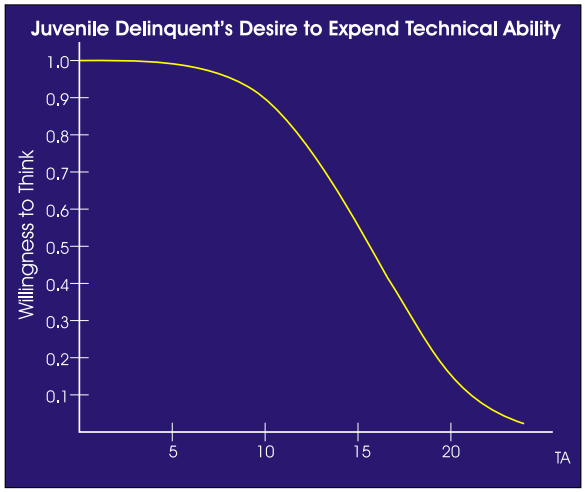


Figure 8 – Juvenile Delinquent's Technical Ability Utility Function

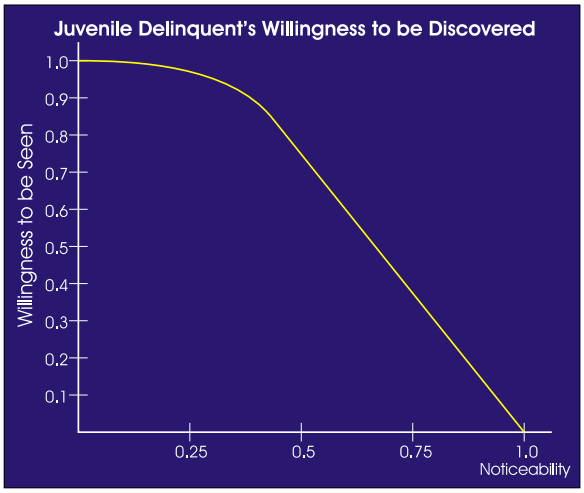


Figure 9 – Juvenile Delinquent Noticeability Utility Function

of ease or difficulty. A good way[12] to estimate the overall difficulty of a specific attack scenario

[12] An alternative is to use a weighted sum. For instance, if there exist three behavioral indicators, and the output of the utility functions for each indicator are *A, B* and *C*, then we might compute an overall value as

$$aA + bB + cC = Overall\ Difficulty \quad \text{where } a + b + c = 1$$

The problem with this approach is that it is does not reflect the fact that the lack of even one resource is enough to prevent an adversary from carrying out an attack. I.e., the attacker's decision is constrained by *AND* node logic.

(as perceived by a particular threat agent) is to compute the product[13] of the outputs of the threat agent's utility functions. For example, consider the Juvenile Delinquent's utility functions for *Cost of Attack, Technical Ability* and *Noticeability* (shown in **Figure 5**, **Figure 6** and **Figure 7**). Suppose that we wish to compare the desirability of various attack scenarios for burglarizing a house. Using the familiar *Burglehouse* attack tree model, we find that the *Break Down Door* attack scenario will cost the adversary $25 (for a steel battering ram), require a *Technical Ability* rating of 10 (out of 100), and expose the miscreant to a 0.3 *Noticeability*. Using the utility functions shown, we discover that

$$f_{cost}(25) = 0.9$$
$$f_{tech\ ability}(10) = 0.9$$
$$f_{noticeabiity}(0.3) = 0.95$$

and therefore

$$Ease\ of\ Attack = 0.9 \times 0.9 \times 0.95 = 0.7695$$

By comparison, the *Steal Opener from Car, Break Down Passage Door* attack scenario requires $30, a *Technical Ability* requirement of 10, and has a *Noticeability* of 0.28. So, the attack is slightly more expensive than the previous case, slightly less noticeable and requires the same technical ability. This yields:

$$f_{cost}(30) = 0.79$$
$$f_{tech\ ability}(10) = 0.9$$
$$f_{noticeabiity}(0.28) = 0.97$$

and therefore

$$Ease\ of\ Attack = 0.79 \times 0.9 \times 0.97 = 0.6897$$

If our assumptions are correct, this attack is slightly harder for a juvenile delinquent than simply *Breaking Down Door.*

Depending on the threat agent, they might value one resource more than another. The model can reflect the different emphasis that the attacker places on different resources by adjusting the shapes of the utility curves.

Since the outcome of this calculation depends heavily on the utility functions it is fair to ask how they are derived. The shape of the curve is based on several assumptions.

We believe that people are completely willing and capable of spending zero resource. Therefore for $x = 0$ we may confidently state that $y = 1$. Although strictly speaking this is an assumption, it is a well founded one. People are very inclined to pursue the goals they desire if they are free.

We also know that the threat agent's willingness (and ability) to spend will reach zero when the

[13] One problem with using a simple product is that the *Ease of Attack* value tends to decrease as additional indicators are added to the model. This effect can be compensated for by taking the n^{th} root of the product – i.e., the geometric mean.

amount of resource equals or exceeds the amount that they control. This means the curve crosses the horizontal axis at the limit of the threat agent's resource. The accuracy of this assumption will depend on the quality of intelligence we have about our adversary's resources. We generally know our adversary well enough to estimate an adversary's resource budget to at least the correct order of magnitude.

To determine the shape of the curve between the two endpoints it would be ideal if we could reference exhaustive psychiatric studies into the psyches of a wide variety of adversaries. Unfortunately, that base of information does not exist. Another strategy for selecting the curve's shape is required.

A simple, straight line could be used to join the two known endpoints. That, at least, conveys the premise that the adversary has a decreasing willingness to part with the resource as the amount to be spent increases. However, further refinements are possible if we consider the adversary's psychology.

A bold adversary who is easily persuaded to spend their resource can be represented using a convex curve (as viewed from above). Their willingness to spend does not drop off until they nearly reach the limit of their resource. On the other hand, a timid adversary, who is strongly attached to a resource, will be reluctant to spend it. This yields a concave curve (as viewed from above).

Choosing curves based on our assumptions about the way we believe that specific groups of people will behave is admittedly not perfect. However, it is certainly more accurate than the binary style of attacker model (**Figure 4**) used previously in pruning. Since the primitive model gave useful results, we expect that the use of a more accurate utility function will yield even better predictions. At the very least it expresses our understanding of the adversary, and exposes our thought processes for review and discussion.

Due to variances in human behavior within a threat agent class, no curve will ever be a perfectly accurate description of a specific threat agent's decision-making process. While acknowledging the limitations inherent in this method of calculating the *Ease of Attack*, we believe that it can be a useful representation of the ease of an attack as perceived by the adversary.

It is sometimes convenient to speak of *Difficulty of Attack* (as the opposite of *Ease of Attack*) The two terms are (philosophically, if not mathematically) the inverse of one another:

$$Difficulty\ of\ Attack = \frac{1}{Ease\ of\ Attack}$$

Attacker Motivation is Related to Attack Benefits

Earlier it was stated that adversaries make decisions on the basis of *cost-benefit*. The calculation of the *Ease of Attack* value considered the attack scenario costs but it did not weigh the benefits

the attacker expected[14] to gain from an attack scenario. These must also be taken into account in order to understand how desirable an attack appears to an adversary.

In the *Burglehouse* example discussed earlier, the attacker's benefits were primarily monetary. In more complex situations multiple types of benefits may accrue from carrying out attack scenarios. Adversaries will be attracted to specific scenarios depending on the particular combination of rewards. Different scenarios will provide different levels of attacker motivation.

All of the direct interaction between an adversary and their target is captured in the leaf nodes of the attack tree. However, many (and usually most) of the benefits an adversary gains from an attack are associated with higher, logical states in the tree[15]. Usually the greatest attacker benefits are associated with the tree's root node with additional, with side benefits occurring at the various intermediate nodes. Since different attack scenarios traverse different paths between leaf nodes and root, the attacker benefits may differ considerably depending on the attack scenario used. In the house burglary example, the attacker might only discover things to steal in the garage if that attack vector were used. The additional prizes might be overlooked if they broke into the house through a window or the front door.

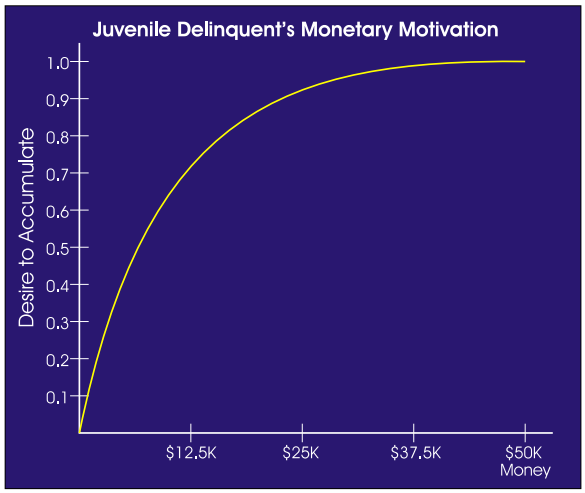


Figure 10 – Juvenile Delinquent's Desire for Increasing Quantities of Money

Most types of rewards exhibit diminishing utility. Even money loses its impact after a certain point. The rarity of billionaires is at least partially due to the fact that most multi-millionaires can't be bothered to continue accumulating wealth. There are very few indulgences available to billionaires that aren't available to hundred-millionaires. There's no point in having toys if you are too busy working to play with them!

The attractiveness of a given reward is subjective. A juvenile delinquent with a minimum wage job (or none at all) may view $50,000 as unlimited wealth. The diminishing *benefit* of wealth to a juvenile delinquent is shown in **Figure 8**. *Benefit* is a measure of the perceived value of a particular resource. Functions which map absolute amounts of the resource to the perceived value are called *attacker benefit utility functions.*

[14] Attackers make their decisions based on their perceptions of the results of a particular course of action. They may, in fact, be wrong in their estimation. In this case, perception is far more important than reality.

[15] Just as the predominant victim impacts tend to occur at higher levels in the tree, so do many of the attackers' benefits.

An industrial spy might have much higher aspirations than the juvenile delinquent. They may even feel that, below a certain threshold, a particular activity isn't worth their effort. The *attacker benefit* curve for such a person is seen in **Figure 9**. **Figure 9** shows that the Industrial Spy doesn't have much motivation to do anything illicit until the reward hits about $500K. Above $500K the desire increases rapidly until about $7M (at which point the industrial spy's greed is becoming satiated).

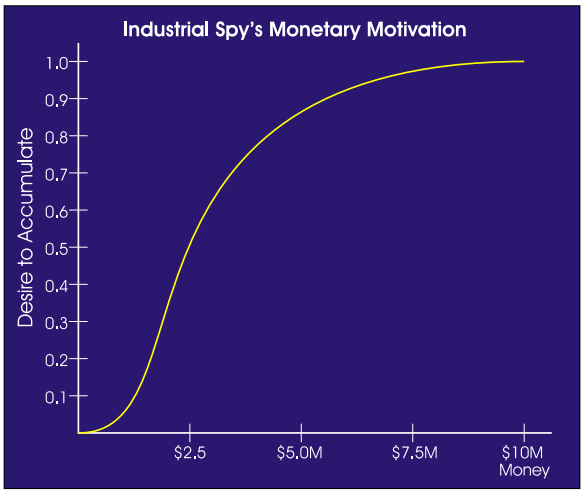


Figure 11 – Industrial Spy's Desire for Increasing Quantities of Money

As mentioned earlier, money is just one of several possible benefits to be garnered through an attack. Revenge, prestige, power and sundry gratifications of desires are all possibilities. These could be represented through other threat agent specific functions (all of which would also yield values between 0 and 1).

Where multiple rewards exist it becomes necessary to combine the output of the corresponding *attacker benefit utility* functions. The multiplication technique used earlier to blend the *ease of attack* functions does not work well for combining the *attacker benefit utility* functions. If multiplication is used, a high score can only result if the attack benefits the attacker in every measurable way. This is not realistic. Even if an attack does not provide every conceivable benefit, an attacker may still find a subset of the potential rewards to be very attractive. For that reason, it is preferable to use a weighted sum to assess the combined value of the *attacker benefit utility* functions.

$$aA + bB + cC + \ldots + nN = \textit{Overall Desirability} \qquad \text{where } a + b + c = 1$$

For example, suppose that the *Burglehouse* attack model incorporated an indicator that measured the amount of destruction that resulted from a particular attack (slashing furniture, soiling carpets, breaking glass) and that the agent profile for a Juvenile Delinquent had a corresponding *attacker benefit* function that reflected the thrill that juvenile delinquents get from this type of activity. How might we combine the value of that thrill with possible monetary benefits?

If we believe that juvenile delinquents like money, but value mayhem and destruction even more, we might assign weighting factors of 0.4 for monetary gain and 0.6 for destruction. If the monetary reward for a particular attack yielded $15,000 then (reading from **Figure 8**) this gives a monetary *attacker benefit* value of approximately 0.78. Using the mayhem benefit function (not shown) we might obtain a value of 0.5 for that particular attack. Calculating a weighted combination of these values yields an *attacker benefit* of $(0.4 \times 0.78) + (0.6 \times 0.5) = 0.612$

It would be simplistic to believe that successful attacks bring only positive benefits to an attacker. Aside from the use of resources, an attack may have one or more detrimental effects on the adversary. The adversary may face time in jail, injury or even death from carrying out the

attack. By applying a similar technique to that used for *attacker benefits*, it is possible to calculate a weighted sum of *attacker detriments*. If the sum of *attacker benefits* and *attacker detriments* is positive, it means that the adversary`s overall perception of the attack is favorable and they will be motivated to try it (within their resource constraints). If the sum is negative, it means that the downside outweighs the benefits and they are repulsed from attempting an attack of that nature. To simplify discussions, we will usually speak only of an attack scenario`s *attacker benefits*, but it should always be remembered that this actually encompasses both the benefits and the detriments.

Capabilistic Propensity of Attack

Earlier we asserted that attackers are more likely to perform attacks that provide a high return with a low expenditure of resources

$$Attack\ Propensity = \frac{Attack\ Benefits}{Attack\ Costs}$$ Given that

$$Attack\ Difficulty\ (i.e.\ perceived\ Attack\ Costs) = \frac{1}{Ease\ of\ Attack}$$

and that attackers will select attacks based on their desirability, this means that

$$Attack\ Propensity = Ease\ of\ Attack \times Attack\ Benefit$$

The Relationship between Capabilistic Propensity and Probability

We use the term *propensity* instead of *probability* to emphasize that, although the meaning is similar to probability, its definition is not derived in the conventional, statistical fashion. Nonetheless, there is good reason to believe that *propensity* corresponds closely to *probability,* or more precisely, *relative frequency*.

In statistics, the fraction of experiments that result in a particular outcome is known as the *relative frequency*. For example, flipping a fair coin 1000 times would be expected to result in about 500 occurrences of "heads", thus yielding a *relative frequency* (or *relative probability*) of 0.5.

In the case of hostile attacks, an "experiment" could be considered to be an encounter between a threat agent and the defender's system. If the fraction of encounters in which the attacker chooses to perform a particular attack scenario corresponds to the *propensity* value for that scenario, then *propensity = relative frequency*.

Unfortunately, it is very difficult to prove this relationship. Controlled experiments are difficult to stage. Also, our method of calculating propensity examines each scenario in isolation. However, one scenario may have a significantly higher propensity (more attractive combination of costs and benefits) than another. In that case, the attractiveness of the higher propensity

scenario will lower the likelihood that the threat agent would choose a competing scenario with a lower propensity value (which might have been chosen had the more attractive scenario not been available). High propensity scenarios mask low propensity scenarios. Only when the propensity values are comparable is there a meaningful choice for the adversary.

Notwithstanding these issues, capabilistic *propensity* generally behaves like a *relative frequency*. When the benefits are high and the perceived capability costs are low the *propensity* will be close to unity. If the benefits are low or the resource costs exceed the attacker's capability then the *propensity* will be close to zero. At least for the boundary conditions *propensity* and *relative frequency* seem to match. The correspondence for other points along the curves will depend largely on the accuracy of our curves.

Although the goal is to provide a quantitative system of metrics that describe attack probability it must be recognized that evaluating human behavior precisely is much more difficult than testing the physical properties (such as strength) of a steel beam. Humans vary significantly (even within classes of threat agents).

Pain Factor – the Victim's Perspective

The discussion above has focused entirely on the adversary and the choices they will make. This was necessary to determine which attack scenarios were most likely to be chosen by an adversary (i.e., the *propensity*). However, from the victim's perspective, there is also a concern about how much an attack will hurt – the perceived impact.

Attack trees model victim impacts by calculating the amounts and types of losses that a victim will incur as an attacker achieves particular states (nodes) in the tree. In most cases, the victim impacts are cumulative as the attacker traverses an attack path through the tree. The overall impact of an attack scenario is calculated by summing the damage associated with the leaf nodes specified in the scenario, and then moving upward through the intermediate nodes traversed by the attack scenario's logic until the root node is reached.

Of course, as was the case with the various classes of attackers, different victims place differing values on the losses they experience. A wealthy multi-national corporation might treat a $100,000 one-time loss as insignificant whereas a small business would be devastated by such a blow. Again we employ the concept of a utility function to translate raw impact values into the subjective pain or damage experienced by the victim. The weighting mechanism used to derive an overall value for aggregated victim impact is similar to that used previously to calculate the positive impact (benefits) obtained by the attacker.

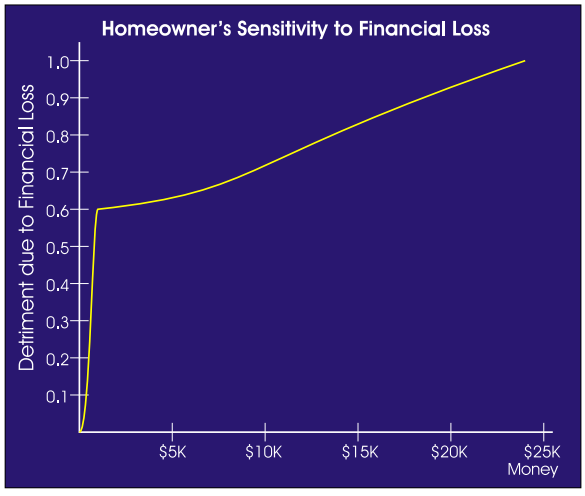


Figure 12 – Homeowner's Perceived Impact

Recalling the house burglary example, the utility function representing the homeowner's perceived damage due to financial loss from burglary is seen

in **Figure 10**. The shape seems unusual until you recall that most homeowners have some form of property insurance. If there is a loss, the homeowner pays the initial damage costs up to the policy's *deductible*. This is represented in the utility function by the steep rise from $0 to $1000 (the deductible). At that point the curve levels out because the insurance company begins to pay the damages. If life were perfect, at this point the curve would become flat. However, experience has shown that insurance companies rarely cover the entire loss. Sometimes there is a disagreement over the value of the item for which compensation is being paid and not all items may be covered. In many cases items that are stolen or damaged are irreplaceable or hold sentimental value. Thus the curve continues to rise gently[16] and approaches 1. Of course this curve would be very different if the homeowner did not have insurance!

The *{Break down door}* scenario has an actual damage figure of $15,250. The intruder does $250 damage to the door and steals $15,000 of goods once inside the house. Using the curve in **Figure 10** this translates to a perceived impact of 0.83. The *{Steal opener from car, Break down passage door}* scenario, with $18,500 of damage, has a perceived impact of 0.9.

Recall that when we considered the behavior of the adversary we created distinct sets of functions for each threat agent. Similarly, in *victim impact* analysis, it is also possible that there are several victims. In the case of the house burglary example it is obvious that the homeowner suffers as a result of an attack. However, if the loss exceeds the insurance policy's deductible then the insurance company also experiences a loss. An impact curve describing the pain felt by the insurance company for varying levels damage could be created.

Scenario Risk Value

Recalling that

$$\textit{Relative Attack Risk} \equiv \textit{Attack Probability} \times \textit{Attack Impact}$$

and given our hypothesis that

$$\textit{Attack Propensity} = \textit{Attack Probability}$$

we are now in a position to calculate the attack risk of each scenario.

$$\textit{Attack Scenario Relative Risk} = \textit{Attack Scenario Propensity} \times \textit{Attack Scenario Impact}$$

Calculations for the two house burglary related scenarios discussed earlier are shown below.

Calculation of Juvenile Delinquent Risks for Two Attack Scenarios

First, determine how difficult the attack scenarios are to carry out based on the perceived value of the resources expended by the juvenile delinquent.

[16] We have assumed that sufficient insurance has been purchased to cover the total destruction of the property. If it is possible that damage could exceed coverage, then the curve would again rise sharply at the point coverage ended.

Resource Requirements for Attacks by Juvenile Delinquent					
		{Break down door}		*{Steal opener from car, Break down passage door}*	
		Raw Cost	Ease to Juv. Del.	Raw Cost	Ease to Juv. Del.
Attacker Cost	Cost of Attack	25	0.9	30	0.79
	Technical Ability	10	0.9	10	0.9
	Noticeability	0.3	0.95	0.28	0.97
	(Product Rule)	**Overall Ease**	0.7695	**Overall Ease**	0.6897

Then, examine the perceived benefits the juvenile delinquent anticipates from the attacks.

Benefits Accrued by Juvenile Delinquent from Specific Attacks					
		{Break down door}		*{Steal opener from car, Break down passage door}*	
		Raw Gain	Perceived Value	Raw Gain	Perceived Value
Attacker Benefit	Money gained	$15,000	0.78	$18,000	0.82
	Satisfy Destructive Urge	5	0.5	7	0.8
	Compute Weighted Sum 0.4 × Money + 0.6 × Satisfy Urges	**Overall Benefit**	0.612	**Overall Benefit**	0.808

Combine the perceived difficulty and the perceived benefits using a cost-benefit computation. This yields the relative frequency or *propensity* that an encounter between a juvenile delinquent and the house would result in the scenario being carried out.

Capabilistic Propensity of Attack by Juvenile Delinquent			
{Break down door}		*{Steal opener from car, Break down passage door}*	
Ease × Benefits	Propensity	Ease × Benefits	Propensity
0.7695 × 0.612	0.471	0.6897 × 0.808	0.557

Next, determine the level of suffering the attacks will cause (as perceived by the victim)

Suffering Experienced by Victim			
{Break down door}		*{Steal opener from car, Break down passage door}*	
Actual Damage	Perceived Damage	Actual Damage	Perceived Damage
$15,250	0.83	$18,500	0.9
Overall Suffering (Single impact, no need to calculate weighted sum)	0.83	$18,500	0.9

Finally, combine the *propensity* of each attack scenario with the perceived victim impact to find the relative risk.

Risk of Two Selected Attack Scenarios (Propensity × Impact)		
Attack Scenario	*Propensity × Impact*	*Relative Risk*
{Break down door}	0.471 × 0.83	0.391
{Steal opener from car, Break down passage door}	0.557 × 0.9	0.50

Relative Risk vs Absolute Risk

As stated earlier, risk is a combination of both probability and impact. If *propensity* is used as the probability term, and *pain factor* as the impact term in the risk equation, the result is a measure of the risk associated with an encounter between the defender's system and a particular threat agent. We call this *relative risk* because it is calculated relative to each encounter between the system and the attacker.

The difference between *relative risk* and *absolute risk* can be illustrated with an example from the insurance industry. A prominent insurance company advertises special, discounted auto insurance rates for seniors, ostensibly because they are such experienced drivers. Closer investigation discloses that seniors are not generally good drivers – their accident rate per mile is surpassed only by new drivers. However, seniors are usually retired and do not use their cars for daily commuting. The total number of miles they drive is much lower than the average motorist. This, it turns out, more than offsets their diminishing physical capabilities. So, they have a high *relative* (per mile) accident rate but a low *absolute* collision rate. The insurance company only cares about the absolute losses and are thus willing to offer seniors better rates.

In similar fashion, the *absolute risk* associated with a particular attack scenario to a system defender will depend on the scenario's *propensity* (the relative frequency for each encounter), the scenario's pain factor, and the number of encounters that will take place in a given time period.

Opportunity – Estimating the Number of Encounters

There are three primary factors that determine the number of encounters that a system will undergo with a specific class of threat agents in a period of time:

1. The number of adversaries who have plausible access to the defender's system. For physical attacks this generally means the attackers and the target are within the same geographic region. For electronic attacks it implies some degree of mutual network connectivity (Internet, dial-up lines).
2. The number of targets (within the region) competing for the attention of the defender.
3. The period of time over which the number of encounters will be estimated.

Additional factors that influence the number of encounters include:

- The nature of the attacker and the fraction of the time they dedicate to attacks.
- The characteristics of the exploit:
  - Will performing the exploit permanently deplete the attacker's resources? This limits each attacker to perform only one attack, referred to as a *single-shot* attack
  - How long does it take for the attacker to do the attack and then regroup for another attack? We call these attacks *single threaded* or *sequential* attacks.
  - Can an attacker attempt to attack multiple targets concurrently? *Multi-threaded* attacks are very common in electronic (computer) attacks.

For each type of exploit, the number of encounters in a given time period[17] can be estimated by

$$\#\,Single\ shot\ encounters = \frac{\#\,Adversaries}{\#\,Targets}$$

$$\#\,Single\ threaded\ encounters = \frac{(\#\,Adversaries)\,(Duty\ Factor)}{(Attack\ Time + Recovery\ Time)\,(\#\,Targets)}$$

$$\#\,Multi\text{-}threaded\ encounters = \frac{(\#\,Adversaries)\,(Thread\ Factor)\,(Duty\ Factor)}{(Attack\ Time + Recovery\ Time)\,(\#\,Targets)}$$

These formulae are approximations. It is expected that future research will provide refinements.

[17] For *single shot* attacks it is assumed that the time period is long enough to allow all of the actors time to have an encounter and decide whether or not they want to irrevocably spend their resources. The formulae for *single threaded* and *multi-threaded* yield the number of encounters per the time period units used to specify the attack and recovery times.

Frequency or Rate of Occurrence (RO)

The number of times a particular attack scenario will occur in a given time period is proportional to the scenario's *propensity* (relative frequency) and the number of encounters that occur in the time period. That is

$$Rate\ of\ Occurrence = Propensity \times Rate\ of\ encounters$$

or

$$Rate\ of\ Occurrence = Propensity \times \#\ encounters/time\ period$$

If the time period chosen is one year, then the frequency is known as the *Annual Rate of Occurrence (ARO).* This term is widely used in conventional risk analysis.

Calculating Absolute Risk (or Cumulative Risk)

As mentioned earlier, a scenario's absolute risk is a combination of its propensity and the rate of encounters (over a particular time period). So, for a given scenario,

$$Absolute\ Risk = (Propensity \times \#\ Encounters/time\ period) \times Time \times Impact$$

or

$$Absolute\ Risk = Rate\ of\ Occurrence \times Time \times Impact$$

The amount of risk accumulates over time, so absolute risk can also be called *cumulative risk.*

Annual Loss Expectancy

Another popular metric is *Annual Loss Expectancy (ALE).* Given the *Annual Rate of Occurrence* it is possible to calculate the *ALE* associated with an attack scenario for each victim impact indicator.

$$Annual\ Loss\ Expectancy = \#\ Occurrences\ (per\ year) \times Scenario\ Impact$$

or

$$Annual\ Loss\ Expectancy = ARO \times Scenario\ Impact$$

There are several caveats associated with this calculation. First, it is assumed that the effect of multiple losses is additive. This may not always be the case. Second, the *ARO* is computed for each scenario as if it were completely independent from all other scenarios. This is clearly not true since one scenario may be sufficiently attractive to distract an adversary from an otherwise attractive attack.

The overall *ALE* (for all scenarios) from a given threat agent is at least as much as the maximum *ALE* of any of the scenarios in the set. It may be as much as the sum of the *AROs* of all of the scenarios (if they are seen as independent choices by the threat agent).

Of course the choice of the *annual* time frame is arbitrary. The number of encounters and losses can be evaluated for any time period of the analyst's choosing. In general, the number of encounters, losses and risk will increase with exposure time. An exception to this might be a

highly targeted attack. In that case the encounters are not random nor ongoing. There may be no encounters from a particular adversary until some real world event occurs which triggers interest in the target.

Scenarios Involving Both Intentional and Random Events

Attack tree analysis was developed to analyze deliberate, hostile activities against a system. An earlier, tree-based approach (known as *fault tree analysis*) has long been used to model random events. It would be highly useful to combine the two techniques to analyze incidents that occur due to either (or both) random and intentional events.

There are fundamental differences in the properties of fault trees and attack trees. In a fault tree, each leaf node has an associated probability value (usually obtained from statistics). Using statistical formulae[18] it is possible to calculate the statistical probability of reaching any node in the tree (taking into account the individual probabilities of all of the paths that lead to that state). Unlike an attack tree, it is unnecessary to examine each path separately (although this can also be done). In fault trees the leaf level events are treated as independently occurring incidents.

In an attack tree, the leaf level events are heavily interdependent, particularly where *AND* nodes are involved. If several operations are required to carry out an attack an intelligent adversary will not waste time performing a preliminary activity (even if it is easy) unless they believe they are also capable of carrying out more difficult subsequent steps in the attack[19]. Sometimes the attacker is not even in a position to attempt some of the leaf nodes unless they first complete other prerequisite steps. Having completed one of a series of steps in a procedure, the next operation becomes the focus of the attacker.

The interdependencies between leaf level events make it impossible to determine, in isolation, the probability of each of the substeps in an attack. Aside from the practical consideration that the statistics may not exist, the probability of the adversary performing operation *A* fundamentally depends on their expectations of the feasibility of *B*. This is one of the main reasons why attack tree analysis focuses on *attack scenarios.* A *scenario* is a complete set of steps that constitutes a successful attack. The steps are considered as a package.

Consider the general case. Imagine a tree model with leaf level events that include both random, probabilistic incidents (e.g., natural disasters, equipment failures, human errors) and hostile, resource constrained attacker activities. Any given attack scenario for this tree may consist of

i Probabilistic events – events with known probability (often acts of nature or routine malfunctions)

[18] Typically *OR* nodes in a fault tree combine probabilities using 1-[(1-a)(1-b)(1-c)...(1-n)], where *a, b* and *c* represent the probability values of children . *AND* nodes combine as the product of the probabilities of the children.

[19] They may also do a preliminary step if it leads to an intermediate goal they feel is worth achieving. This can be understood through subtree analysis.

Calculating the probability of a scenario that contains only probabilistic events for is trivial so long as the appropriate statistics exist. The well known statistical method for computing the probability of multiple, independent events is to simply multiply the individual probabilities together.

Note that probabilistic events have no resource cost values. For example, there is no cost or technical ability required (from anyone) for a hurricane to occur. The probability of these events is independent of threat agents.

ii Capability-constrained events – events that require the adversary to expend resources

The indicator definitions relating to adversary resources make it easy to calculate[20] the total amount of the various resources that will be expended in the particular attack scenario. Passing these costs through the resource constraint utility functions discussed earlier allows an estimation of the difficulty of the scenario. Combining this difficulty estimate with the desirability value (obtained by using the attacker benefit utility functions) yields the *propensity* value for the attack scenario. If the utility functions are accurate, *propensity* is equivalent to *probability* (or more precisely, *relative frequency*).

We emphasize that the *propensity* is calculated from the set of hostile operations that the attacker must perform. This makes sense because the attacker chooses whether or not to perform the component hostile actions by considering the attack scenario, with all of its operations, as a single unit. So, although we do not know the propensity of each individual operation, we can determine the propensity of the set of operations that make up an attack. In other words, we do not (generally) know the propensity for an adversary to reach particular intermediate nodes in a tree. We do know the propensity that the adversary will reach the tree's root using a particular attack scenario.

iii A mix of both Probabilistic and Capability-constrained events

When we talk about a mixed incident, we are usually referring to a hostile attack that requires some random event as a prerequisite or corequisite. These situations are quite plausible. For instance, areas situated along the Gulf of Mexico typically experience several hurricanes per year. During a hurricane, a facility that is normally manned may be deserted. Disruptions to power and communication lines may mean that burglar alarms do not function (or their operators may believe alarms are a result of the storm). This may embolden an adversary to carry out an attack that they would otherwise not consider.

[20] *OR* nodes in an attack tree scenario simply inherit the resource requirements passed up by the child that is participating in that scenario. *AND* nodes combine their children's resources through an analyst specified formula.

In a sense, the random event does not actually change the hostile portion of the attack scenario so much as it opens a restricted time window during which it may occur.

Most of the statistics for random events are given as a frequency and a duration. For instance, a component may have a *Mean Time Between Failure (MTBF)* specification and a *Mean Time To Repair (MTTR)*. Hurricanes have a frequency (number of hurricanes per year) and a duration (a few days).

Since the hostile parts of the mixed attack are only plausible during the interval in which the random situation is in effect it means that we should calculate the number of hostile encounters based on the shortened interval. For instance, if two hurricanes per year are expected, with a duration of two days each, then there will be approximately four days of hurricane per year. Analysis of a scenario with a hostile component that depends on a hurricane would require us to calculate the number of expected encounters over a four day period, not 365 days. This would mean that only about 1% of the encounters would occur. The overall probability of the mixed scenario would be about 1% of the same scenario without the random component.

Note that, in the discussion above, the threat agents are not responsible for creating or instigating the random events. They merely take opportunistic advantage of these events when they transpire. There is another important random factor that has not been dealt with so far.

Probabilistic Outcomes of Capabilistic Activities

In some cases, a probabilistic factor is introduced because of a capabilistic action or actions of an adversary. In a sense, it is as if the adversary rolled a dice. The outcome (orientation of the dice) is determined by probability but there would have been no possibility of an outcome unless someone rolled the dice. The adversary creates the event but not the outcome. We call these events *probabilistic outcomes*. Although *probabilistic outcomes* are most often observed at leaf nodes, they can occur at any node in the tree that has a capabilistic component.

At the leaf node level there is a direct interaction between the adversary and the target. In the discussion thus far, leaf level actions have been completely deterministic in nature. If the adversary applied the resources specified in the node's capabilistic requirements, the outcome was assured. This is overly simplistic. In some cases, despite the application of the requisite resources, the leaf level operation may fail due to random factors that are not entirely predictable and beyond the attacker's control.

Similarly, there may be random factors that influence whether or not an attack progresses past an intermediate AND or OR node. Despite all of the necessary conditions being satisfied by leaf nodes or subtrees below, the AND/OR node may occasionally still fail to be achieved due to random factors.

To model this random component, capabilistic nodes can be given an attribute called the *attack success efficiency (ASE)*. The *attack success efficiency* is input as a value between 0 and 1 that

specifies the likelihood that the application of the specified resources will cause the node to succeed.

The user can specify one of two ways in which the *attack success efficiency* term could be interpreted. It could affect either the *ease of attack* coefficients or the *attacker benefits* of an attack scenario. The correct interpretation depends on whether the threat agent is astute or naive.

If the adversary is clever they will recognize that one or more of the operations in an attack scenario have a possibility of failure that is beyond their control. They will be aware that the average return will be less than the nominal return associated with a completely successful attack. The effect on their motivation is best understood by devaluing the raw benefits before they are translated to perceived benefits via their respective utility functions. We call this approach, *attacker benefit-based ASE*.

Note that the attack detriments are not affected since they usually apply whether or not the attack is successful. The resulting reduction in perceived attacker benefits will make the attack scenario less desirable and thus reduce the propensity of the attack scenario. This affects both the relative risk (i.e., the risk on a per encounter basis) and the absolute risk.

In other cases, the adversary may be naive or optimistic, not recognizing that the exploits and attack scenario they hope to use have less than a 100% success rate. If this is the case, the ASE should be applied to the scenario frequency term. The number of encounters is multiplied by the *attack success efficiency* (yielding an effective # encounters term) which is used to calculate the the expected scenario frequency. The relative risk is unchanged, but the cumulative risk of all scenarios involving this node is reduced by the *attack success efficiency* factor. We call this *encounter-based ASE*.

It is possible that a given scenario may have both *attacker benefit-based ASE* and *encounter-based ASE* components. In that case it will be necessary to accumulate the attacker benefit-based ASEs separately from the encounter-based ASE terms. The former will be multiplied together to get an overall attacker benefit-based ASE which will reduce the attacker benefits before they are transformed by the utility functions. The latter will be multiplied together and used to compute an overall effective # of encounters term.

In both *attacker benefit-based ASE* and *encounter-based ASE* cases, the cumulative risk will be decreased for any attack scenario that has components with ASE values < 1.

What Do the Numbers Mean?

The formulas and processes described above allow risk values to be calculated for both hostile and random scenarios. But there has never been any discussion of what the numbers mean. For example, is a cumulative risk value of 0.5 acceptable or very dangerous? The answer depends partly on the risk tolerance of the defender, the time frame of the study and how the analyst has calibrated the model.

$$Absolute\ Risk = Rate\ of\ Occurrence \times Time \times Impact$$

When *Rate of Occurrence* × *Time = 1*, the event in question will have happened exactly once[21]. If the event has an impact of 1, then absolute risk will also equal 1.

Whether or not this is tolerable to the defender depends on how the analyst has calibrated the model and the amount of time involved. For example, suppose an *impact* value of 1 is chosen to correspond to an outcome that is no longer survivable by the defender. For a company president this might mean the company going bankrupt. A military commander might rate losing a major battle as a 1. The leader of a nation will associate it with the collapse of the country. The actual rating will vary depending on scope and context.

Whatever terrible impact is associated with the value "1", when a time period *T = 1/Rate of Occurrence* has elapsed, the event will have happened once. If that time period is much longer than the period of interest, then the risk may be acceptable. If the time period is of the same order of magnitude as the period of interest, then the risk is likely excessive.

For instance, suppose a particular computer system is built to keep track of scores during the Olympic games. In that case, *impact = 1* might be defined to be someone breaking into the computer and altering a score. If the model showed that risk would only approach 1 after a century of exposure, and given that the games only last for two weeks, it might be decided that the level of risk would be acceptable. I.e., the risk after two weeks would only be 3.86×10^{-4}.

Of course, a defender may choose to take action upon encountering risk values of much less than 1, even if the time period is longer than they need to worry about. For instance, a cumulative risk value of 1 (over a period of ten years) could also correspond to events with damages of 1/10 the impact of the worst case scenario occurring once every year. Only the stakeholders can identify what is acceptable and what is not. In many cases it is a question of economics. If it is cheaper to prevent the scenario occurring than it is to recover from it, then the risk may be unacceptable.

Total Risk vs Scenario Risk

To this point, our focus has been on scenarios. Purely probabilistic scenarios are independent. That means that likelihood of one probabilistic scenario occurring has little effect on another. For instance, the risk to a building located on the coast from a hurricane is independent from the risk of an earthquake. The overall risk from probabilistic scenarios is the cumulative risk (or the sum of the individual risks). This is reflected in the behavior of insurance companies who have fixed costs for adding clauses or "riders" to a policy for specified risks. They treat each risk independently.

---

[21] Strictly speaking, this is true only in an average sense. When we speak of an event happening once per time period (e.g., once per year), there is a probability distribution function that reflects that, in any given case, the event may happen sooner or later than expected. Even though we are calculating *propensity* (instead of *probability*) based on assumptions of how an adversary will behave based on resource costs, attack desirability and so forth, certain factors that are still random. For instance, there is a certain amount of randomness on how an attacker chooses between similar competing targets. So, it is acknowledged that some probability distribution function (which we have not defined) must exist to allow variability from the frequencies predicted by our model.

We know that capabilistic scenarios are not independent. However, to simplify things, we have considered each capabilistic scenario in isolation from all others. This was helpful because it reduced every situation to a binary decision: Given a particular adversary (with specified resources and goals) who faces a particular opportunity (scenario), will they do it or will they walk away? However, the reality is that it is not a binary decision for the attacker. Like a hungry diner in a restaurant with an extensive menu, there are multiple choices (with varying prices and flavors) that will satiate their hunger. But, once the diner orders their favorite dish, all other choices cease to matter. Yet, if the waiter were to inform the patron that the preferred plate was not available (or it had never been on the menu) then the diner would have made the next best choice. Oddly, one way of improving sales of a less popular menu item, is by deleting a more popular meal from the menu!

In similar fashion our attacker is faced with a menu of attack scenarios which compete for his or her attention. Otherwise acceptable choices might be ignored if better choice is available. So, what makes a choice "best" for an adversary? Simply put, it is the scenario that the attacker perceives as providing the highest combination of ease of attack and desirability, i.e., capabilistic propensity.

In some cases the interaction between an adversary and a target is a zero sum gain. That is, the attacker can only gain at the expense of the victim. When that is true it is essential that the attack model show the victim's losses as both an attacker benefit and as a victim impact. However, in many cases, adversaries do not gain satisfaction from the suffering of the victim -- the damage to the victim is simply a side effect of the attacker achieving their primary goals. Particularly in these situations, the attack scenario with the highest capabilistic propensity may not be the scenario with the highest cumulative risk.

Since the lower propensity (but higher risk) scenario will be ignored by the adversary (who has, from their point of view, made a better choice), the overall cumulative risk from the set of competing choices available to that adversary will be the value associated with the scenario with the highest cumulative probability. This means that, given a set of competing scenarios, it is not appropriate to simply add up the cumulative risk associated with each scenario (as we did for probabilistic scenarios). Instead, the cumulative risk from a given threat agent is the cumulative risk value of the highest propensity scenario available to that threat agent. Theoretically, that is the only scenario they will ever choose.

While this method of risk estimation is theoretically correct, there are some practical issues that make it necessary to factor in the risks associated with more than just the highest propensity scenario. We must remember that all models contain uncertainty. If a number of scenarios have similar propensity values, then it is hard to determine which is really the highest. Individual members of a threat agent class may also vary somewhat in their resources, skills and goals. These, and other factors, make it advisable to examine all of the scenarios that have propensity values near that of the highest propensity scenario. This acknowledges that we may not be able to definitively say which scenario is the attacker's best choice, but that we have confidence in stating that it will be one of a small set of choices. If any of the comparable scenarios have

higher cumulative risk values, then it is prudent to assume that the higher value is the risk contribution from that scenario set.

It has already been shown that probabilistic scenarios are treated differently than capabilistic scenarios. What about mixed scenarios -- those that have both capabilistic and probabilistic components?

Mixed scenarios are very similar to capabilistic scenarios. They are essentially capabilistic scenarios with a time window (like a time lock on a bank vault). Consider where a burglar has managed to obtain the vault combination. Obviously, the capability requirement (of entering the combination) is very low. But if the lock only accepts combinations for 1 minute per day, that may make the propensity appear low. But, during that minute, the propensity is very high and it is short circuiting other scenarios that normally have higher propensities. So, sometimes it preempts and sometimes it doesn't – what to do? There is no perfect answer, but a good answer is that if, during the time window, the capabilistic portion of the overall probability of the mixed scenario is comparable with other capabilistic scenarios, then it should be included as a candidate for having the highest cumulative risk. If it is significantly less than other capabilistic scenarios, it can be thrown away - it will only get less likely once the time constraint is applied. Although equal weight to the capabilistic propensity of a mixed scenario (that may only be available for a fraction of the exposure time of a purely capabilistic scenario) clearly overemphasizes its importance, this is largely compensated for by the fact that the cumulative risk value of a mixed scenario is reduced by multiplying it with the probability value.

The process above hinges on the ability to create a list of scenarios that compete for a threat agent's preference. Clearly it is necessary to create a separate set for each threat agent. Might it be necessary to subdivide the scenarios further?

Sophisticated threat models tag leaf level events as single shot, single threaded or multi-threaded. Scenarios that involve multiple events of different types degrade to the lowest type. Usually there are more multi-threaded encounters than single threaded (or single shot). For instance, an attacker using a computer as an automated tool may reach many more potential victims than they could possibly access in physical attacks. Might we have to subdivide these attacks into different sets and consider them separately? It turns out that this is not necessary because the higher number of encounters of the multi-threaded scenario scales the overall risk. This is equivalent to saying that the number of encounters has not increased, but the desirability has. So, it appears that no subclassing is required.

Total Cumulative Risk = Probabilistic Risk + Capabilistic Risk

Probabilistic risk is calculated by adding the individual cumulative risks of all probabilistic attack scenarios (which are repeated in our current tables for each threat agent).

A good estimate of capabilistic risk can be found by

1. Sorting the scenario table (of capabilistic and mixed mode scenarios) by capabilistic probability.
2. Throw away the bottom 80%.

3. Find the maximum cumulative risk (in the remaining 20% of scenarios).

Countermeasures and Controls

The discussion above has shown how attack trees can model an adversary's behavior with respect to a target, and even how an assessment of risk can be performed. However, surely it is the point of the exercise to prevent attacks or mitigate the effects if they occur. The attack tree models described previously use three different techniques to model controls and countermeasures.

1. The capabilistic resource requirements associated with a leaf node can be increased to reflect an improvement in the leaf level component the attacker was attempting to exploit.

   For example, if an inexpensive, hollow core door (with a cheap lockset) was replaced with a high quality, solid core door (equipped with top grade hardware) then battering through or prying open the door would become more difficult. The *Technical Ability* and *Cost of Attack* indicator values for the *Batter Door* and *Pry Open Door* leaf nodes would increase to reflect the improvements in the door components.

2. Changes can be made to the defender's system to make an attack scenario more complex and challenging.

   In cases where an attack required a series of steps (depicted by an AND node with several children), then additional children could be added beneath the AND node representing new activities contrived by the defender[22]. The new activities would be chosen by the defender to be as difficult as possible.

   For instance, if an attack scenario for obtaining electronic information involved the steps of: *{Enter computer room, Steal data tape, Read tape}* then the attack could be made much more difficult by encrypting all data on tapes. The revised attack scenario would then be: *{Enter computer room, Steal data tape, Read tape, Break encryption}*. Hopefully, the *Break encryption* step would be very challenging to the attacker. The *Break encryption* procedure could be a single leaf node or, more typically, in a subtree describing various approaches to breaking encryption.

   This approach is very useful when a security analyst has been charged with securing a system that is either poorly understood or cannot be changed. Essentially, the analyst agrees to concede that the adversary will prevail against these unknown or unchangeable components. Instead of trying to fix the unfixable, the defender changes the system's architecture such that it no longer matters that the adversary will prevail against the original components. The system is protected by new, hardened mechanisms that cannot be easily subverted, and that prevent the attacker from climbing the tree to the root node (or other high level, high impact nodes).

[22] If, in the original system, no AND node existed because only a single attack step was required, then an AND node would be inserted at the appropriate location and both the previously existing step, and the new additional steps, placed beneath it.

3. Use Boolean capabilistic indicators and attacker capabilities to filter attacks that will be stopped by certain defenses.

   For instance, certain leaf level activities in a tree might be technically straightforward and low cost, but only feasible for a trusted, authorized insider. These operations would have a *Breach of Trust* indicator value of *True*. The threat agent profile for an insider would reflect the insider's capability to perform these privileged operations whereas an outsider's profile would lack that cability. So, if an organization should implement special procedures to eliminate hostile insiders (background checks, regular polygraph examinations, procedures to ensure that critical activities are always performed by two randomly chosen personnel) then the countermeasure would be represented by setting the attacker's *Breach of Trust* capability to be *False*. This would prevent any of the leaf activities that require *Breach of Trust* from being performed.

These three techniques have proven to be effective in a wide variety of circumstances. However, they are implicit and may not be recognized as countermeasures by someone reviewing the model. It would be useful to be able to represent controls in a more direct fashion.

Countermeasure nodes

A variety of academic papers[23] have been published describing extensions to the attack tree model. Research continues in this area and no single approach has emerged as the ultimate solution.

One simple strategy is based largely on the principles demonstrated earlier for merging capabilistic and stochastic threats. This approach seems fairly compatible with the previously cited technique detailed by Roy et al.

Seen from an attacker's point of view, a countermeasure can be seen as an additional obstacle that has been added to an attack procedure by the defender. Since procedures are depicted in an attack tree as AND nodes, it follows that a countermeasure must always fall beneath an AND node. In most cases, the AND node already exists (as the parent of the original attack steps). If the attack previously had only a single step (represented by a leaf node) then an AND node parent must be introduced and the leaf node and the countermeasure placed beneath.

From the defender's point of view, a good countermeasure is a system that operates correctly and fulfils its mission as much of the time as possible. Although an ideal countermeasure would work 100% of the time, few real world systems achieve this level of performance. For example, X-ray scanners (or the humans operating them) can easily miss dangerous items in luggage. Network intrusion detection systems can similarly fail to identify a certain fraction of malicious packets. In both cases a 99% success rating would be considered good. However, this also means that malicious payloads are getting past the control 1% of the time!

---

[23] Of particular interest is the 2011 paper published by Roy, Kim and Trivedi (ACT: Towards unifying the constructs of attack and defense trees, Arpan Roy, Dong Seong Kim and Kishor S Trivedi, Security and Communication Networks, 2011; 3:1-15). In the paper, Roy et al discuss an extended tree model they call an attack countermeasure tree.

Most of the failures in a countermeasure are due to human error, technical limitations or other random factors and not due to the wiles of the adversary. If this is true, the different failure modes in a countermeasure can be represented using a probability-based fault tree. A fault tree is constructed representing the various ways in which the countermeasure can fail. The leaf level events causing the failure are assumed to be independent events, so the standard statistical formulae apply.

In the earlier section describing how capabilistic and probabilistic threats could be combined in a single tree, all of the paths of failure subtrees were expanded in conjunction with other capabilistic components in the attack scenario list. While this is not wrong, it does not allow the analyst to see the overall effectiveness of the countermeasure system acting as a unit. What is needed is the ability to compute the overall probability of the countermeasure failing (from any and all causes) and to show the effect of the failure on a deliberate attack.

This is solved by defining a new countermeasure node type. A countermeasure subtree (comprised of probabilistic countermeasure nodes) is placed directly beneath an AND node. Some unique symbology can be applied to distinguish the countermeasure nodes – perhaps a shield or other similar emblem.

Since the countermeasure subtree is essentially a fault tree, it is possible to compute a single probability value for the countermeasure as a whole. The probability value represents the likelihood that the countermeasure will have failed. It is during that fraction of time that the adversary will be able to elude the countermeasure. So, the overall likelihood of the attack scenario succeeding is

$$\text{Propensity of Attack}_{\text{Countermeasure}} = \text{Propensity of Attack}_{\text{Without Countermeasure}} \times \text{Probability}_{\text{Countermeasure Failure}}$$

Only the root node of the countermeasure subtree will appear in attack scenario lists.

This abstraction should be general enough to accommodate many of the schemes proposed in the literature for representing countermeasures. For instance, one of the proposed mechanisms for describing countermeasure failures involves a detection step and a mitigation step. To represent this, a countermeasure tree could be structured with two separate logical components beneath an AND node, one branch representing failures to detect a hostile event and the other the failure to mitigate it.

Impacts (attacker benefits/detriments and victim impacts) can be injected into nodes in a countermeasure subtree, just as with any other node in the tree. However, the way the impact values are handled differs somewhat from the approach described earlier.

Where an AND node appears in a countermeasure subtree, the method of calculating impacts is straightforward because all of the children must occur to satisfy the logic. So, whatever AND function is defined for the impact indicator (typically sum or maximum) should be used to compute the impact.

Situations involving an OR node are more complex. Any non-null subset of the OR node's children could occur, and might have an impact. Since each of the children's probability is independent from its siblings, there is no requirement that the probabilities of the OR's children

tally to 1 - and, in fact, they usually do not. One approach is to use a form of Monte Carlo analysis to roll the dice and see which children become active on a given trial. Then, of the subset of children that are active, further analysis is used to determine which child's impact will be chosen as that of the trial. This approach takes the view that, in the case where several of the OR's children become active, they would not (in the real world) all become active at the same instant in time - one would be first and that is the one that would cause the impact. Monte Carlo analysis can select which of the active children is most likely to occur first based on the active children's relative probabilities. Choosing the first active child's impact is not necessarily correct in all situations, but it seems reasonable in most situations. The Monte Carlo analysis is repeated a sufficient number of times that the impact converges toward a particular value. Further research will likely improve on this strategy.

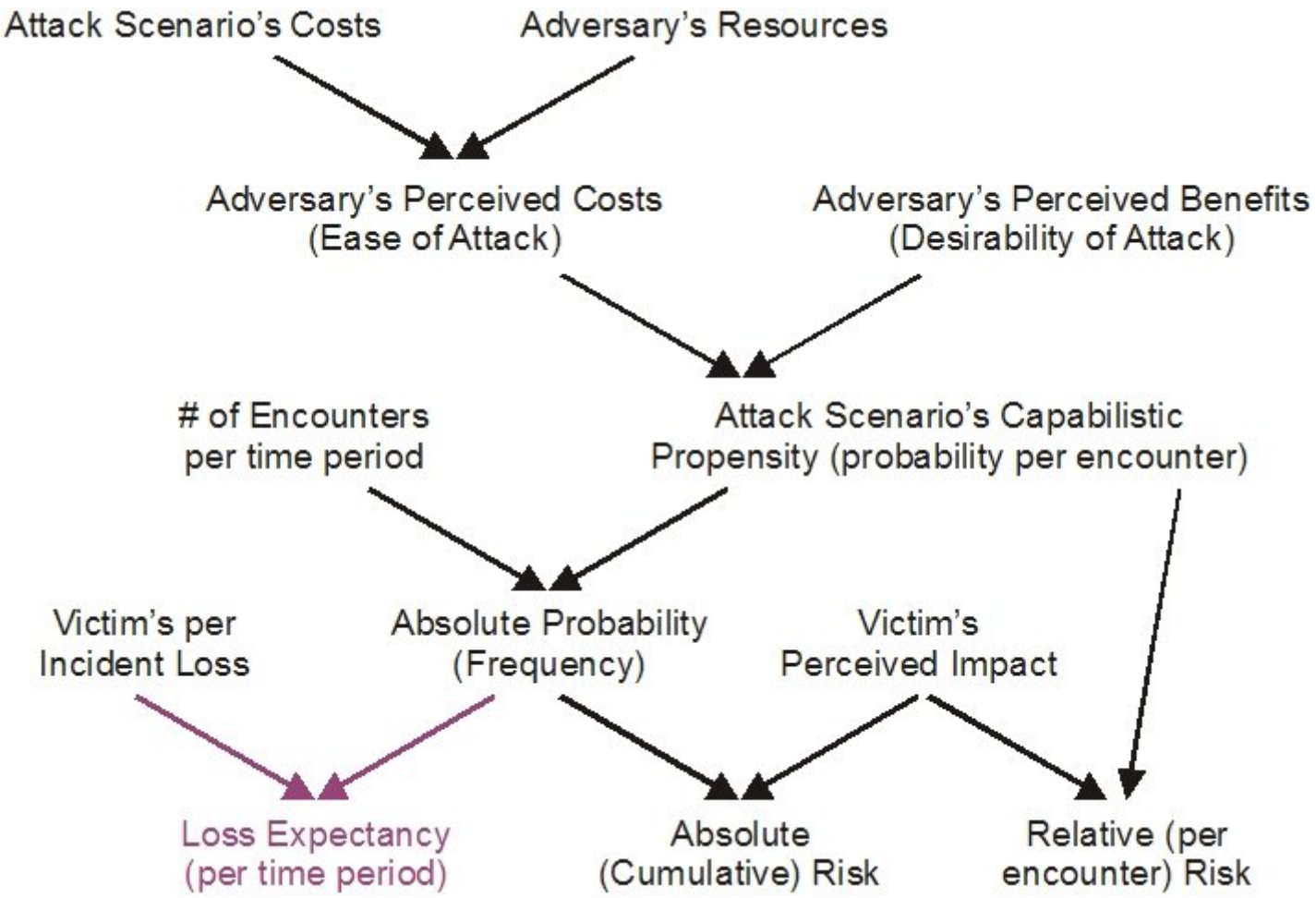
One way of avoiding these issues is to simply place all impacts in the countermeasure subtree's root node. Indeed, in many cases the impact of a countermeasure failure will be the same regardless of how it fails, so might be a better strategy than assigning impacts at various levels of the countermeasure subtree.

Conclusion

There are numerous benefits to attack tree-based risk analysis. They provide an easy, convenient way to capture subject matter experts' expertise and reasoning. Analysts have stated that being forced to describe their system in an attack tree model enhanced their understanding of security issues. The clear logic of the attack tree model enhances discussions between security experts. Recording the assumptions and information that were available at the time of the analysis is valuable for proving due diligence. Most importantly, properly applied attack tree models allow analysts to turn a diverse collection of disconnected facts and assumptions into understanding.

Adversaries have long been willing to use sophisticated tools (particularly in information technology-based attacks). Hopefully the ideas presented in this paper will provide defenders with similar benefits.

Appendix I – Basic Hostile Attack Risk Analysis Flowchart



Glossary

absolute risk — for a deliberate, malicious event, it is the risk associated with an attack scenario for the number of encounters between the adversary and the target system anticipated in a given time period. For a stochastic event, it is the risk associated with the number of events of that type expected in the given time period. If the number of adversarial encounters or the number of stochastic events increases with time, the *absolute risk* will also increase. For that reason *absolute risk* is also known as *cumulative risk*. See also, *risk* and *relative risk.*

annual loss expectancy — also known as *ALE.* The average losses associated with a given scenario over a one year time period.

attack scenario — a minimal collection of *leaf* level attack tree events that is sufficient to satisfy the *AND / OR* logic of the attack tree and result in the *root* node being achieved. Strictly speaking, an attack scenario consists only of the specified leaf nodes. Often, however, the parent *AND / OR* nodes above the leaf nodes are included in a description of the attack scenario in order to show more clearly the path taken to *root***.**

attack success efficiency — the fraction of which an adversary's actions to perform an exploit that will result in success. This can also apply to the likelihood with which an attack will advance upward past an *AND* or *OR* node given the successful fulfilment of the Boolean conditions of the node's children.

attack tree — an attack tree is a mathematical, tree-structured diagram or model representing a system that an adversary might want to attack. The model describes the choices and goals available to an attacker. Similar to many other tree-based models, attack trees consist of a top level *root* node that represents the overall objective of the adversary (and usually what the defender wishes to prevent). There are generally a number of different approaches the attacker might use to achieve the high level goal and the diagram is extended to show these alternatives. Alternative approaches for achieving goals are denoted through the *OR* nodes. Processes or procedures are represented through *AND* nodes. The bottommost levels of the tree, *leaf* nodes, describe operations performed by potential adversaries to exploit some vulnerability in the system's defenses. If a set of *leaf* level operations cause the Boolean *AND/OR* logic of the leaf nodes' parents and ancestors to be satisfied, the high level root node is goal is attained and a successful attack has occurred.

behavioral indicator	parameters representing the resources and abilities a *threat agent* would need to provide in order to execute an *attack scenario*. The scarcity of resources may make it more difficult for adversaries to perform a given attack, thus affecting their behavior.
capabilistic propensity	a metric of the likelihood that, given an opportunity to do so (an encounter with the target system), a *threat agent* will perform an attack scenario. The metric is based on a combination of the scenario's *ease of attack* to the adversary and its desirability. *Capabilistic propensity* is closely related to the concepts of *relative frequency* or *relative probability* in statistics. For instance, just as there is a 1 in 6 (or 0.1667) chance that throwing a set of dice will result in a 6, an attack scenario with a *capabilistic propensity* of 0.1667 means that, for every six encounters between an adversary and a target, there is a 0.1667 likelihood they will execute the scenario (subject to caveats discussed in the text).
cumulative risk	See *absolute risk.*
ease of attack	a metric calculated by comparing the set of resources needed to carry out a particular attack scenario with a given threat agent's ability and willingness to spend those resources. The opposite of *ease of attack* is *attack difficulty.*
exploit	(n) a detailed procedure for acting on a vulnerability; (v) to perform a procedure that takes advantage of a vulnerability.
impact	the positive or negative effects on the attacker, or the negative effects on the victim, that result from the execution of an attack scenario. See also *attacker benefit, attacker detriment* and *victim impact.*
pruning	a method for evaluating the likelihood and feasibility of an adversary performing a given attack scenario. If the resources required to perform the scenario are beyond the adversary's means, then the scenario is infeasible for that adversary.
relative risk	For a deliberate, malicious event, it is the risk associated with an attack scenario given a single encounter between the adversary and the target system. For a stochastic event, it is the risk associated with a single event of that type. See also, *capabilistic propensity, risk* and *absolute risk.*
risk	the combination of the likelihood of an event and the resulting (usually negative) impact. See also, *absolute risk* and *relative risk.*
threat	a potential source of danger to a system.
threat agent	a class of people who represent a threat to a system.