TDDC03 Projects, Spring 2006

A Comparison of Attack Trees Threat Modeling and OCTAVE

Xin Zhang Shangjin Xu

Linköpings universitetet, Sweden

Email: {xinzh553,shaxu648}@student.liu.se

Claudiu Duma

A Comparison of Attack Trees, Threat Modeling and OCTAVE

Xin Zhang Shangjin Xu
Linköpings universitetet, Sweden
Email: {xinzh553,shaxu648}@student.liu.se

Abstract

Avoidance and discovery of security vulnerabilities in information systems and managing enterprises requires awareness of typical risks and a good understanding of vulnerabilities and threats and their exploitations. Various methods for characterizing, identifying and managing threats have been presented. Bruce Schneier has invented the Attack Trees, Microsoft call their method Threat Modeling and Carnegie Mellon University developed a solution for managing an entire enterprise named OCTAVE (Operationally Critical Threat, Asset, and Vulnerability Evaluation). In this paper we compare the three methods of Attack Trees, Threat Modeling and OCTAVE, and also compare two softwares using Attack Trees and Threat Modeling.

1. Introduction

Frequent reports about security vulnerabilities show that still many deficits exist in the development of secure software systems. The problem is even more pressing as the attacker activity and the destructiveness of attacks, such as Distributed Denial of Service, have increased over the last years, and more and more new kinds of attacks come out.

In order to avoid vulnerabilities in the first place, developers, administrators, senior managers of organizations etc. have to be aware of the causes of vulnerabilities, possible exploits, and attackers not only from external but also from internal.

The purpose of this paper is to compare the Attack Trees, Threat Modeling and different softwares using different methods. We deeply introduce and discuss the three techniques appropriately in different scenarios and how to use them in suitable situations and what kinds of problems they can solve.

This paper is divided into four main parts. In the following three sections in order to make the comparisons clearly for the readers we will discuss what are Attack Trees, Threat Modeling and OCTAVE, what are the criteria of those three techniques, what are the basic structures of these three techniques and how to implement them. The last part of this paper is a comparison of them, in this section, we mainly compare the three methods in different ways such as in categorizing the threats, identifying threats and managing threats and also compare the tools which are using and extending criteria of the three techniques.

2. Attack Trees

2.1. Definitions

Definition 1: *Attack Trees is a method used as an intuitive aid in threat analysis.* [13]

It has existed in various forms, various names, for many years, but has been most recently described as a systematic method to characterize system security based on varying attacks. It refines information about attacks by identifying the compromise of enterprise security or survivability as the root of the tree.

Definition 2: *Attack Trees is a multi-leveled diagram consisting of one root, several intermediate nodes and many leaves*. [14]

From the bottom up, child nodes are conditions (goals) which must be either all or partially satisfied to make their direct parent leaf true, when the root is satisfied, the attack

is complete. Each parent can be satisfied only by its direct child nodes.

2.2. Structure and Semantics

The main building blocks of attack trees are nodes. We decompose nodes of an attack tree either as:

• a set of attack sub-goals, all of which must be achieved for the attack to succeed, that are represented as an AND decomposition.

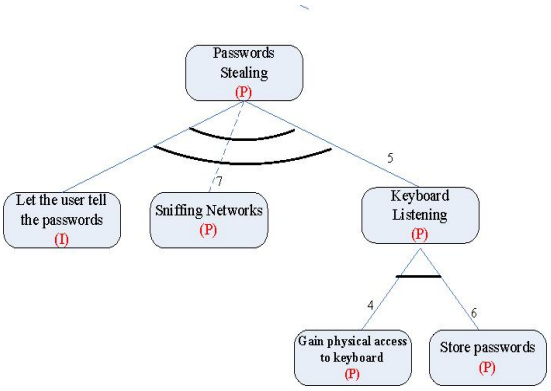

Figure 1

• a set of attack sub-goals, any one of which must be achieved for the attack to succeed, that are represented as an OR decomposition..

Normally, Attack Trees consists of some AND decompositions and OR decompositions.

"AND" node: It means that to achieve the parent goal, all sub goals nodes must be successful. For instance, in the picture figure 1, in order to do "Keyboard listening", the attacker must perform both "gain physical access to keyboard" and "store password" attacks successful.

"OR" node: To achieve the parent goal, either of child goals should be successful. For instance, attackers can do password stealing by sniffing the network or keyboard listening or letting user tell them password. Either one of them is successful, the password can be cracked.

2.3. Advanced Features

According to the basic structure of Attack Trees, we can add some advanced features in practical use.

• **I (impossible) and P (possible)**: The possibilities can be assigned to various leaf nodes. We present real line as P (possible) and dot line as I (impossible).

It is very helpful in identifying threats through vulnerabilities when applying this feature. For the reason that not all vulnerabilities are threats, only a weakness of a system which can be exploited to achieve an attack is called a threat. By involving the I.P. attributes, the security designers can easily identify threats existing in the system.

For instance, as the system illustrated by the figure 1 above, it is easy to identify that both the nodes "Let the user tell the passwords" and "Keyboard Listening" are two possible threats (using real line) and "Sniffing Networks" is one vulnerability but not a threat (using dot line) to that system. Moreover, the security designers of that system could decide whether to make the specific countermeasures to those particular threats.

• **Weight**: There is a cost for each path from the non-root node to its direct parent. The weight denotes how difficult for this node to achieve its goal.

The figure 1 above presents the path value, the non-root node to its parent, shows how "expensive" to accomplish its goal. From the values we can easily calculate which attack is the cheapest one

2.4. Practical Uses

Attack Trees provides a formal, methodical way of describing the securities of systems, based on varying attacks. [15]

• **Risk Analysis:** In Attack Trees, *Attack Trees Analysis is* a specific modeling technique for understanding risk in complex situations. It is categorized to hierarchy models and constructed to show all the ways of attacking or damaging a system. The capabilities of various classes of attackers are compared with the resources which are required to perform the attacks. And also Attack Trees with the advanced features (Weights) can tell which attacks according to the specific system would chiefly occur.

• **Attack Pattern Reuse:** *The practicality of Attack Trees to characterize attacks on the real-world systems depends on being able to reuse previously developed patterns of attack.* [14]

We define Attack Pattern as a generic representation of a deliberate, malicious attack that commonly occurs in specific contexts. Each attack pattern contains the overall goal of the attack specified by the pattern, a list of preconditions for its use, the steps for carrying out the attack and a list of postconditions that are true if the attack is successful. [14]

For example:

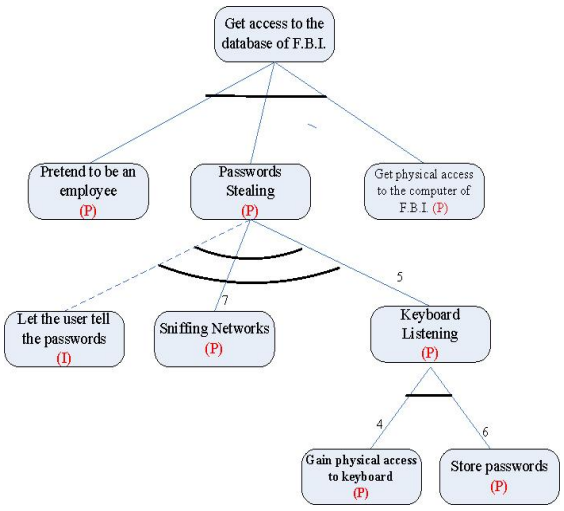

Figure 2

Goal: Get access to the database of F.B.I.
Attack:
AND 1. Pretend to be an employee of F.B.I.
2. Get physical access to the computer of F.B.I.
3. Get the password to that computer.
OR 1. Sniffing Networks.
2. Keyboard Listening.
AND 1. Gain physical access to keyboard.
2. Store passwords

In this sample attack, the attack pattern of the example above (Figure 2) can be reused as a sub-tree of this Attack Trees. The bold AND is the relationship among those three attacks, in another words, to achieve the goal "Get access to the database of F.B.I." 1, 2 and 3 conditions must be satisfied. The bold OR denotes either one of the conditions is true, the goal is achieved. In term of "Get the password to that computer", either "Sniffing Networks" or "Keyboard Listening" is satisfied, the attack is successful.

3. Threat Modeling

3.1. What is Threat Modeling?

Definition: *Thread Modeling involves the understandings of the complexity of the system and identifying all possible threats to the system.*

In another word of saying, Threat Modeling contains these steps when implementing, which are "Identify Assets", "An Architecture Overview" and "Identify Threats". [Figure 3]

Further more, Threat modeling is the practice of working with developers to identify critical areas of applications dealing with sensitive information. The model is used to map information flow and identify critical areas of the application's infrastructure that require added security attention.

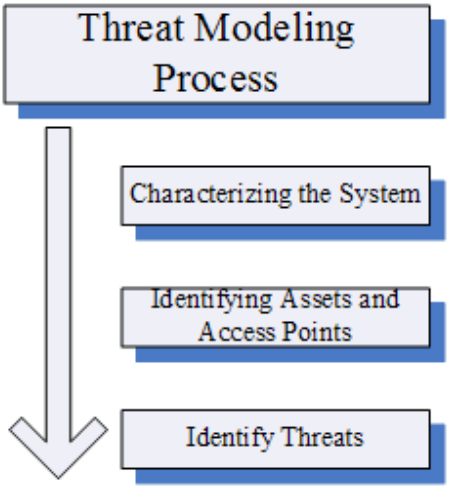

Figure 3

Characterize the system: *Using simple diagrams and tables to document the architecture of your application, including subsystems, trusting boundaries, and data flow.* [18]

Assets: *An abstract or concrete resource that a system must protect from misuse by an adversary*.

Access points: *what the attacker is going to use to perform an attack*

Identify Assets and access points: *Identifying the valuable assets that your systems must protect.* [18]

Identify Threats: Enumerating threats through each of the system's assets, describing all the potential attacks and then reviewing a list of attack goals.

3.2. Threat Modeling Process

3.2.1 Characterizing the System:

The first step of applying threat modeling is to understand the system in details. It contains the understandings of components, boundaries and interconnections of components, usage scenarios, hardware profiles, software profiles and identified assumptions and dependencies. What we need is a system model that reveals the essential characteristics of the system.

Designers can use different methods to model the system. For example, for a model specifying the functionality of software, Data Flow Diagram (DFD) is applicable to build the system model. Unfortunately, it is impossible use DFD as model tools to build a complex network system model. But Network Model can easily describe network system.

3.2.1.1 Data Flow Diagram

Data flow diagrams (DFDs) are introduced and popularized for software application structured analysis and design. It decomposes applications into functional components and indicates the flow of data from external entities into the system.

The DFD approach is easy to identify threats because designers can follow the flow of data and commands when executed by the system. Designers can also trace how the data are parsed, how they are acted on and which assets they are interacting with.

For example,

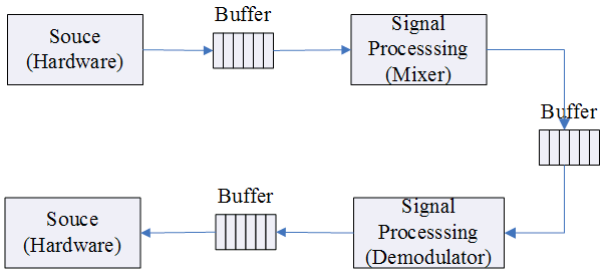

Figure 4

A data flow diagram of FM receiver software application is illustrated in Figure 4 The diagram is used to identify different components and how data is processed. It shows the relationships among various system components. For instance, the “Demodulator” component depends on the data processed by the “ Mixer” component. Knowledge of the relationships is useful in identifying threats

3.2.1.2 Network Model

For a computer network system, it is difficult to dissect the system into different parts. Then Data Flow Diagram is no more applicable to these kinds of situations, because designers usually don’t know how different software components exist in the system and thus no knowledge of data flow can be made.

Network system can be viewed by network model. With the help of network model, designers can examine communications among computers with different roles.

It is necessary to clarify different roles of the computer in the network in the first place and then the correspondences among different roles can be mapped out.

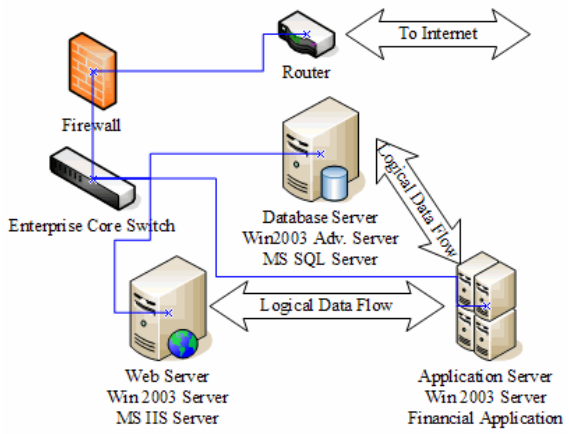

Figure 5

As network system described in figure 5, it is difficult to capture data flow through different components, especially when web server is composed by several computers. In the network model, computers are categorized by roles such as web server, database server and application server. Then communications between them are easily to be differentiated.

3.3. Identifying Assets and Access Points

In order to perform threat modeling, designers should find all vulnerabilities of the system that can be exploited by attackers. It can be easily done by fully understanding the whole system. With the information gathered in previous steps, designers can characterize the system in term of access risk, system tolerance, resources and objectives.

The analyst of attackers is also needed and designers should try to reach following questions:

Who are attackers?
What’s their motivation and goals?
How much inside information do they have?

Identifying assets and access point is a critical step in threat modeling. It defines the targets of threats. For a threat, target is essential. It is impossible to get a threat without target.

Asset can be tangible such as certain components, process and data. It can also be abstract concept, for instance, data consistency.

Access points are what the attacker is going to use to perform an attack. RPC interfaces, system ports, configuration files stored in server and coding interfaces are examples. Trust boundaries in the system should also be determined. *A trust boundary is a boundary across which there is a varied level of trust.* For example, in the figure 3.1.1.2, only application server can have access to database server. Usually, trust boundary can be determined by trust levels which indicate how much trust is required to get access to portions of system. For the complex network system, network model is an aid to define trust boundary by examining the data communicated among different computers with roles.

3.4. Identifying threats

After "characterizing the system" and "identifying assets and access points" have been completed, designers should think about threats to the system. Threats can be from authorized users or unauthorized users spoofing as authorized users or using some tricks to bypass security mechanisms; threats can also come from intentional or unintentional actions.

Usually, it is started from generating lists of threats in similar systems. The threats are categorized into three categories: network threats, host threats and application threats.

Network Threats:

1) Denial of service attack
2) IP spoofing
3) Error configuration of rules or in Access Control Lists.
4) Sensitive Data flowing unencrypted though the network.

Host Threats:

1) Vulnerabilities that can be exploited by attackers
2) Lack of clearly stated trust boundary

Application Threats

1) Code that's prone to buffer overflows, SQL injection.
2) Defective or missing data encryption resulting password compromise.

Attack Trees is useful in helping determine threats, it provides a reusable pattern. Furthermore, Attack Trees helps to determine whether the system is susceptible to the threats. Although working with known pattern Attack Trees generates common threats, threats corresponding to specific system require deep analysis of unique qualities of the system being modeled.

4. OCTAVE

4.1. What is OCTAVE?

OCTAVE is the abbreviation of Operationally Critical Threat, Asset, and Vulnerability Evaluation and developed by The Networked Systems Survivability (NSS) Program of the Software Engineering Institute (SEI) of Carnegie Mellon University and registered in the United States Patent and Trademark Office.

The OCTAVE framework is designed for the proposes of describing the evaluations of information security risks and being a self-directed activity for organizations.

Information security risk evaluation mainly focuses on identifying vulnerabilities in organizations' computing infrastructures and addresses assets and threats implicitly.

A Self-directed Activity means that the people inside the organization are in the best position to lead the evaluations and make decisions. It is a small team (normally called analysis team) comprising representatives from both the business departments and IT departments of the organization.

4.2. OCTAVE Method

By following the OCTAVE Method, an organization can make information protection decisions based on risks to the C.I.A. (Confidentiality, Integrity, and Availability) of critical information technology assets.

Using a three phase approach, OCTAVE examines not only technical but also organizational issues to assemble a comprehensive picture of the information security needs of an organization. Each phase of OCTAVE is designed to produce meaningful results for the organization.

Phase 1: *Build Enterprise-Wide Security Requirements.* [19]

In phase 1 of OCTAVE, It has four processes to examine the enterprise by gathering information from people in different units and levels within the organization.

Process 1: *Identify Enterprise Knowledge*. [19]

This process mainly identifies what senior managers perceive to be the key assets and their values, the threats to those assets, indicators of risk, and the current protection strategy employed by the enterprise. To achieve the goal, these 5 activities have to be done.

1. *Characterize key enterprise assets.*
2. *Describe threats to assets.*
3. *Describe current and planned strategy to protect assets.*
4. *Identify risk indicators.*
5. *Select operational areas to evaluate.* [19]

Activity 1 brings out and prioritizes the key assets in the organization from the perspective of senior management. The outputs will be a prioritized list of enterprise assets with relative values.

Activity 2 draws out a description of the threats to the identified assets in the organization from the outputted list. And this activity outputs an Enterprise Threat Profile (ETP).

According to the generated ETP and the knowledge of senior managers which is concerning important assets, threats, current protection strategies and potential risk a new enterprise protection strategy outputted by activity 3.

Activity 4 will output that there may be a potential for assets to be at risk through the new enterprise protection strategy.

From activity 5, the key operational areas (those affecting the highest priority enterprise assets) will be presented and examined in the evaluation as well as managers and key staff of those areas.

Process 2: *Identify Operational Area Knowledge.* [19]

The goal of this process is to understand the perspective of operational area managers within the enterprise. Process 2 has 6 activities to do.

1. *Characterize key operational area assets*
2. *Characterize assets in relation to enterprise assets.*
3. *Describe threats to assets.*
4. *Describe current and planned strategy to protect assets.*
5. *Identify risk indicators.*
6. *Select staff to evaluate.* [19]

Activity 2 produces existing relationships between the operational area assets identified in the previous activity with the enterprise assets identified in Process 1.

Activity 6 outputs the key staff (those affecting the highest priority operational area assets). This can include project and support function team leaders as well as key project and support function team members.

Process 3: *Identify Staff Knowledge.* [19]

The goal of this process is to understand the perspective of the staff in the enterprise.

1. *Characterize key staff assets.*
2. *Characterize assets in relation to operational area and enterprise assets.*
3. *Describe threats to assets.*
4. *Describe current and planned strategy to protect assets.*
5. *Identify risk indicators.* [19]

Activity 2 elicits existing relationships between the staff assets identified in the previous activity with the operational area and enterprise assets identified in previous processes.

Process 4: *Establish Security Requirements.* [19]

This fourth process establishes security requirements which are built on the information gathered in the first three processes by involving these activities below.

1. *Map assets identified in prior processes.*
2. *Combine threats identified in prior processes.*
3. *Collect protection strategies.*
4. *Collect risk indicators.*
5. *Establish security requirements.* [19]

Activity 1 examines the relationships among the assets which are identified by personnel from different levels and units in the enterprise. The result is a mapping of relationships taking the different perspectives into account and also identifies those assets that are most important to the enterprise.

Activity 2 combines the threats identified by the staff of the organization. Threats indicate what or whom the assets are being protected from. The output is threat profiles.

Activity 3 collates the current protection strategies employed by the enterprise. The protection strategy outlines what is being done to protect the organization's important information assets. And will generate new current protection strategies.

Activity 4 will result risk indicators which concern from different levels within the enterprise indicating that there may be potential for assets to be at risk.

Activity 5 identifies the requirements with respect to Confidentiality, Integrity and Availability of the identified assets.

Activity 6 combines the outputs from the previous activities in Process 4 to produce a blueprint for the protection strategy. The blueprint outlines the following for each asset: threats, risk indicators, current protection strategies and security requirements.

Phase 2: *Identify Infrastructure Vulnerabilities.* [19]

Phase 2 of OCTAVE uses the asset and threat information from Phase 1 to identify the high-priority components of the information infrastructure (both the physical infrastructure and the computing infrastructure), it also evaluates the information infrastructure to identify vulnerabilities. The ultimate goal is to identify missing policies and practices as well as infrastructure vulnerabilities.

The following two processes comprise Phase 2:

Process 5: *Map High-Priority Information Assets to Information Infrastructure.* [19]

It defines the activity of taking the asset and threat information from Phase 1 and identifying the high-priority components of the infrastructure so that they can be examined for vulnerabilities.

The activities for Process 5 are the following:

1. *Identify configuration of the information infrastructure.*
2. *Consolidate identified assets with identified infrastructure.*
3. *Examine all access paths.*
4. *Examine data flows.*
5. *Identify related assets.* [19]

Activity 1 examines documented artifacts and the knowledge of the staff concerning the information infrastructure. The documented artifacts used as inputs to this activity might not be current. The purpose of this activity is to produce updated documentation to reflect the state of the present computing and physical infrastructures.

Activity 2 maps the important assets to the computing and physical infrastructures.

Activity 3 traces paths to the important assets via the computing and physical infrastructures.

Activity 4 traces data flows of the important assets via the computing and physical infrastructures.

Activity 5 identifies any assets that might relate an important asset in some way. For example, operating system or database software might be needed to access important assets, making it a related asset to those important assets.

.

Process 6: *Perform Infrastructure Vulnerability Evaluation.* [19]

This process defines the activity of evaluating the vulnerability of the high-priority information infrastructure components identified in Process 5 and its goal is to identify the vulnerabilities present in the existing infrastructure and to identify missing policies or practices.

The activities are following:

1. *Select intrusion scenarios.*
2. *Set scope of the infrastructure examination.*
3. *Examine infrastructure.* [19]

Activity 1 identifies potential intrusion scenarios based on the characteristics of the enterprise. Characteristics include important assets, threats to the assets, risk indicators that might affect the assets, physical configuration of the information infrastructure, and high-priority infrastructure components

Activity 2 defines the extent of the infrastructure evaluation by considering existing policies and practices, missing policies and practices and vulnerabilities for which the enterprise should be examined.

Activity 3 performs the infrastructure evaluation to identify which vulnerabilities are presented.

Phase 2: *Determine Security Risk Management Strategy.* [19]

It analyzes assets, threats, and vulnerability information in the context of intrusion scenarios to identify and prioritize the risks to the enterprise. In addition, a protection strategy is developed and implemented in the enterprise.

Its goal is to identify risks to the enterprise and develop a protection strategy to mitigate the highest priority risks.

To achieve the goal, two processes much be done:

Process 7: *Conduct Multi-Dimensional Risk Analysis.* [19]

It means to generate a prioritized list of risks based on impact and probability. The activities for process 7 are the following:

1. *Determine points of vulnerability in potential intrusion scenarios.*
2. *Examine assets exposed by the validated intrusion scenarios.*

3. *Examine threats to the exposed assets*
4. *Construct a statement of risk.*
5. *Determine priority risks to the enterprise.*[19]

Activity 1 examines potential intrusion scenarios which are based on the identified vulnerabilities and identifies which intrusion scenarios are possible based on the vulnerabilities.

Activity 2 identifies assets which are exposed by the validated intrusion scenarios and determines the impact of exposed assets to the enterprise.

Activity 3 assigns probabilities for each threat which is based on the exposed assets and the possible intrusion scenarios. The highest threat probability for each exposed asset will be considered in later activities.

Activity 4 defines statements of risk which are based on the knowledge of the staff along with an understanding of validated intrusion scenarios, exposed assets, impacts of exposed assets, threats to the exposed assets and threat probabilities.

Activity 5 prioritizes the risks based on their impacts and probabilities.

Process 8: *Develop Protection Strategy.* [19]

The goal of this process is to produce a protection strategy for reducing risk and a risk management plan for managing risk on a continual basis.
The activities for Process 8 are:

1. *Identify candidate mitigation approaches.*
2. Develop protection strategy.
3. Develop a comprehensive plan to manage security risks.
4. Implement selected protection strategy.

Activity 1 develops candidate approaches for mitigating the highest-priority risks by considering existing and missing policies and practices, threats, assets, vulnerabilities and available technology.

Activity 2 selects mitigation approaches to improve the security of the enterprise by considering the following: candidate mitigation approaches, impact on assets, the number of assets at risk, the cost of solutions and resources available.

Activity 3 develops a comprehensive security risk management plan by considering how to implement the protection strategy and manage risks on a continual basis.

Activity 4 implements and monitors the protection strategy for effectiveness.

5. Comparisons

5.1. A Comparison of three methods

	Attacking Trees	Threat Modeling	OCTAVE
Characterizing System	No	Yes	Yes
Identifying assets	No	Yes	Yes
Identifying threats	Yes	Yes	Yes
Prioritizing threats	No	No	Yes

Characterizing System and identifying assets:

As stated in the previous chapters, only threat modeling and OCTAVE perform these two operations.

Identifying threats:

All of them identify threats in the system. Furthermore attack trees can be used in attack modeling and OCTAVE to determine the possibilities of threats to the system.

Prioritizing threats:

Only OCTAVE prioritizes threats. It is required in the risk assessment.

	Attacking Trees	Threat Modeling	OCTAVE
Knowing system details	No	Yes	Yes
For potential attacks	No	Yes	Yes
Security requirements	N/A	As basis	building

Knowing the system details:

Attacking trees do not characterize system and identify assets; the generation of attack trees is performed without knowledge of details of the system.

Both threat modeling and OCTAVE understand system in details and examine vulnerabilities based on unique quality of specific system. These are achieved by performing characterizing the system and identifying assets.

For potential attacks:

The generation of attack trees is for limited, known or artificial scenario.

Threat modeling reveals threats based on the vulnerabilities existing in the system, so it covers a broader range of attacks than attacking trees. For instance, it can uncover functional threats existing in the inner system.

Security requirements:

For the limitations of attacking trees, it can not be as basis for security requirements.

Threat modeling can be an aid for the formation of security requirements.

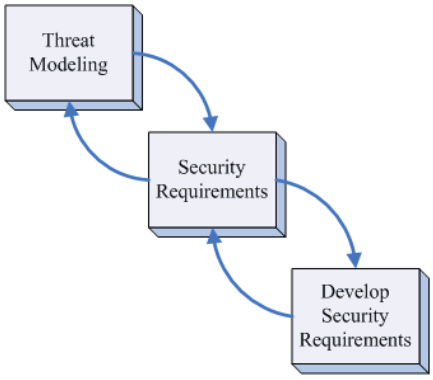

Figure 6

During the formation of security requirements, threats are analyzed based on their impact and probability and a decision is made. It means risk analysis and risk assessment are performed. One necessary condition for risk assessment is threats must be prioritized. **Identifying threats helps develop realistic and meaningful security requirements. If the security requirements are faulty, the system cannot be secure**.

In OCTAVE, security requirements are built in the first phase.

	Attacking Trees	Threat Modeling	OCTAVE
Risk Assessment	No	No	Yes
Risk Management	No	No	Yes
Countermeasure	No	No	Yes

Risk Assessment, **Risk Management,Countermeasure**

Risk assessment is performed to map each threat either into a mitigation mechanism or an assumption that it is not worth worrying about. To access the risk of threats, the threats must be prioritized.

Usually there are four possible ways to manage a risk: accept the risk, transfer the risk, remove the risk and mitigate the risk. In risk measurement, countermeasures must be given to corresponding ways.

In attacking tree and threat modeling, threats are not prioritized and thus risk management is not performed as well as mitigations.

5.2. Attacking trees vs. Threat Modeling

Attack trees model a chosen set of finite state machine and are feasible only in small scenario. It is created by simply brainstorming an attacker's intentions.

Threat modeling is required for:

1) Complex software systems that integrate multiple infrastructure and technologies.
2) Customized application solutions
3) All other cases where it is unacceptable to implement pre-compiled "to-do" lists provided by vendor or standard committee.

Threat modeling is systematic to ensure that as many possible threats and vulnerabilities are discovered by developers rather than attackers. It reveals a list of **potential** threats that needs to be compiled prior to generating attack trees.

5.3. OCTAVE vs. (Attacking trees & Threat Modeling)

*OCTAVE is **different from typical technology-focused** assessments. It focuses on organizational risk and strategic, practice-related issues, balancing operational risk, security practices, and technology.*

It means that attack trees and threat modeling are focusing on the "identifying threats"; OCTAVE covers the dot line area described in figure 6.3.1.

5.4. A Comparison of Tools

5.4.1 SecurITree---Attack Trees Based Risk Analysis

SecuITree is a graphical attacking tree modeling tool to show all ways of attacking or damaging the system. The tool only models attacks within the

capabilities of adversaries. It means results provided by the tool are the attacks that must be worried about.

Steps:

1) Define the overall goal of the attacker and then decompose it into several sub-goals.
2) Continue the step-wise decomposition into smaller and smaller tasks.
3) The goals can be either “And” or “Or” condition nodes which are represented by different shape.
4) Each node (goal) has associated with it additional information.

Analysis:

The tool is based on the theory of attack trees. The goal of the tool is to tell designers which attack is most likely to happen or which attack is worth worrying about mostly based on the constructed attacks model.

Based on known attack patterns: Yes. It constructs the tree in the view of attackers. Unknown attack pattern will not be included.

Ability to find potential attacks: No. It does not include attacks beyond the ability of adversaries. Moreover, it is done without identifying assets of system and thus without knowledge of all vulnerabilities of the system.
Giving countermeasures to attacks: No countermeasure is given.
Rating attacks: The level of impacts is manually defined by designers.
Identify different groups or categories of attackers: Yes. The groups will vary depending on the situation.

5.4.2 Microsoft Threat Analysis & Modeling v2.0

The Microsoft Application Consulting & Engineering (ACE) team has, over the past few years, evolved and optimized a process of threat modeling to help empower businesses to do effective application risk management during the **software development lifecycle** and beyond.

Steps:

Step 1: Identify security objectives.
Step 2: Create an application overview. Itemizing your application's important characteristics and actors helps you to identify relevant threats during step 4.
Step 3: Decompose your application. A detailed understanding of the mechanics of your application makes it easier for you to uncover more relevant and more detailed threats.
Step 4: Identify threats. Use details from steps 2 and 3 to identify threats relevant to your application scenario and context.
Step 5: Document threats.
Step 6: Rate threats

Analysis

Microsoft makes great improvement on the threat modeling. It integrates risk management into the process and gives countermeasures.

The tool is intelligent to automatically contextualized threats and countermeasures based on the library of known attacks. What designers should do in the threat modeling is to organize and consolidate already known information such as roles, components and data.

5.4.3 Comparison

	SecuITree	Threat Modeling
Revealing potential attacks	No	Yes
Giving countermeasures	No	No
Risk Management	No	Yes
Only used in software development	No	Yes

SecuITree is based on the attack tree but make no further improvement. It forces designers to state assumptions explicitly and helps designers understand attacks to the system more clearly but gives no countermeasure. It can be used in a lot of areas.

Threat modeling tool developed by Microsoft is highly intelligent. It automatically gives countermeasures based on known attacks and it is used in the whole application development lifecycle.

6. Conclusions

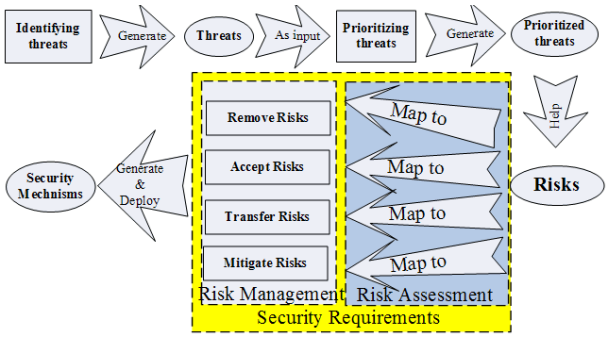

Figure 7

Attack tree does not involve all potential threats to the system. It builds the model on the finite set of attacks which are usually known to attackers.

Threat modeling identifies all threats to the system in the view of developer and it is as basis for security requirements.

Both of attack trees and threat modeling only cover "identifying threat".

OCTAVE is a risk-based strategic assessment and planning technique for security. It covers all steps described in the Figure 7.

7. References

[1] F.Swiderski and W.Snyder. *Threat Modeling*. Microsoft Press,2004

[2] Christopher Alberts, Audrey Dorofee. *An introduction to OCTAVE Method*

[3] Christopher Alberts, Audrey Dorofee, James Stevens, Carol Woody. *Introduction to the OCTAVE® Approach*.

[4] Christopher Alberts, Audrey Dorofee. *OCTAVE Method Implementation Guide v2.0*. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2001.

[5] Christopher Alberts, Audrey Dorofee. *OCTAVE Criteria v2.0.* (CMU/SEI-2001-TR-020, ADA 396654). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2001.

[6] Schneier, B., *Attack Trees: Modeling Security Threats*, Dr.Dobb's Journal, December 1999.

[7] TANAT, *Threat And Attack Tree Modeling plus Simulation*, 2004. http://www13. informatik.tu-muenchen.de:8080/tanat/.

[8] Amaneza Technologies Limited. *A quick tour of attack tree based risk analysis using SecurITree*. Technical report, 2002.

[9] Alexander Opel. *Design and implementation of a support tool for attack trees*, 2005

[10] G. Obradovic, *Threat modeling and data sensitivity classification for information security risk analysis*, in Presentation at Data Protection '03, 2003

[11] Microsoft, *Threat modeling for drivers*, http://www.microsoft.com/whdc/driver/security/threatmodel

[12] O. Sheyner, J. Haines, S. Jha, R. Lippmann, and J.Wing. *Automated Generation and Analysis of Attack Graphs*. In Proc. of IEEE Symposium on Security and Privacy, April 2002.

[13] Sjouke Mauw and Martijin Oostdijk. *Foundations of Attack Trees*.

[14] Andrew P.Moore, Robert J. Ellison, Richard C. Linger. *Attack Modeling for Information Security and Survivability*, March 2001.

[15] Bruce Schneier. *Attack Trees*, Dr. Dobb's Journal December 1999

[16] Suvda Myagamar, Adam J.Lee, William Yurcik, *Threat Modeling as a Basis for Security Requirements*, National Center for Supercomputing Applications(NCSA)

[17] Ruby Qurashi, MCI NetSec,Eight steps for integrating security into application development,Computerworld,December 06, 2005

[18] J.D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla and Anandha Murukan, *Improving Web Application Security: Threats and Countermeasures*, June 2003, Microsoft Corporation.

[19] Christopher Alberts, Sandra Behrens, Richard Pethia, William Wilson. *Operational Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE) Framework, Version 1.0*. June 1999.