



# RISK CENTRIC THREAT MODELING

*Process for Attack Simulation and Threat Analysis*

**Tony Uceda Vélez • Marco M. Morana**

**WILEY**



# **RISK CENTRIC THREAT MODELING**



# **RISK CENTRIC THREAT MODELING**

---

## **Process for Attack Simulation and Threat Analysis**

**TONY UCEDAVÉLEZ AND MARCO M. MORANA**

**WILEY**

Copyright © 2015 by John Wiley & Sons, Inc. All rights reserved

Published by John Wiley & Sons, Inc., Hoboken, New Jersey  
Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4470, or on the web at [www.copyright.com](http://www.copyright.com). Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

**Limit of Liability/Disclaimer of Warranty:** While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services or for technical support, please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic formats. For more information about Wiley products, visit our web site at [www.wiley.com](http://www.wiley.com).

***Library of Congress Cataloging-in-Publication Data:***

Tony UcedaVélez

Risk Centric Threat Modeling : process for attack simulation and threat analysis / Tony UcedaVélez,  
Marco M. Morana

pages cm

Summary: "This book describes how to apply application threat modeling as an advanced preventive form of security"—Provided by publisher.

Includes bibliographical references and index.

ISBN 978-0-470-50096-5 (hardback)

1. Data protection. 2. Computer security. 3. Management information systems—Security measures.  
4. Computer networks—Security measures. 5. Risk assessment. I. UcedaVélez, Tony, 1976- II. Title.

HF5548.37.M67 2015

658.4'7011—dc23

2015000692

Cover Image: Courtesy of Fromold Books, <http://www.fromoldbooks.org/>  
Typeset in 10pt/12pt TimesLTStd by SPi Global, Chennai, India

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

1 2015

*To Suzanne, my patient and loving wife, who supported me throughout the five years of writing and research; thank you for your patience and endless support. –Marco*

*To Heidi, Simon, Serina, Sofia, Samson. For all the soccer balls I missed to kick in the backyard, the tea times I failed to sit in, and the date nights I couldn't make due to fulfilling this project, this is for you. Deo gratias. Deus lux Mea. –Tony*

*Special thanks to Sarah Varnell and Caitlyn Patterson (VerSprite) for all of their review, edits, and writing guidance.*





# CONTENTS

<b>Foreword</b>	<b>ix</b>
<b>Preface</b>	<b>xv</b>
<b>List of Figures</b>	<b>xvii</b>
<b>List of Tables</b>	<b>xxiii</b>
<b>1 Threat Modeling Overview</b>	<b>1</b>
Definitions, 1	
Origins and Use, 3	
Summary, 8	
Rationale and Evolution of Security Analysis, 9	
Summary, 19	
Building A Better Risk Model, 19	
Summary, 31	
Threat Anatomy, 33	
Summary, 48	
Crowdsourcing Risk Analytics, 48	
<b>2 Objectives and Benefits of Threat Modeling</b>	<b>63</b>
Defining a Risk Mitigation Strategy, 63	
Improving Application Security, 82	
Building Security in the Software Development Life Cycle, 92	

Identifying Application Vulnerabilities and Design Flaws, 104	
Analyzing Application Security Risks, 118	
<b>3 Existing Threat Modeling Approaches</b>	<b>137</b>
Security, Software, Risk-Based Variants, 137	
<b>4 Threat Modeling Within the SDLC</b>	<b>195</b>
Building Security in SDLC with Threat Modeling, 195	
Integrating Threat Modeling Within The Different Types of SDLCs, 205	
<b>5 Threat Modeling and Risk Management</b>	<b>235</b>
Data Breach Incidents and Lessons for Risk Management, 235	
Threats and Risk Analysis, 259	
Risk-Based Threat Modeling, 282	
Threat Modeling in Information Security and Risk Management Processes, 289	
Threat Modeling Within Security Incident Response Processes, 306	
<b>6 Intro to PASTA</b>	<b>317</b>
Risk-Centric Threat Modeling, 317	
<b>7 Diving Deeper into PASTA</b>	<b>343</b>
Exploring the Seven Stages and Embedded Threat Modeling Activities, 343	
Chapter Summary, 478	
<b>8 PASTA Use Case</b>	<b>479</b>
PASTA Use Case Example Walk-Through, 479	
<b>Glossary</b>	<b>633</b>
<b>References</b>	<b>653</b>
<b>Index</b>	<b>657</b>

# FOREWORD

The cover page of this book includes a drawing from George Kruger Gray's "The Siege of the Castle." The picture depicts castles under siege and illustrates the challenges to protect against the different attacks used in the Middle Ages such as siege equipment; mobile armored shelters, ladders, and wheeled ramps, by attackers trying to scale the walls built to protect the castles. This picture is a stark reminder of the challenges that cyber-security faces to defend from cyber-attacks of the modern era. In the Middle Ages, attackers stormed the castle from different positions, bypassing the defensive walls, and breaking into the main entry castle doors. In the modern era, attackers strike from the different data interfaces that are available, breaking into the applications user and data interfaces, attacking the firewalls, and application access controls. This picture is also a reminder that defenses such as castle walls, fortified gateways, towers, turrets, arrow loops, drawbridges, and moats become obsolete with the emergence of new threats. In the case of castle defenses, this was the increased presence of gunpowder weapons, such as cannons, in the fourteenth century. In the case of cyber-defenses, the emergence of sophisticated cyber-crime tools that can successfully bypass security defenses, such as anti-viruses, firewalls, and user authentication; require that we be vigilant, monitoring, and improving our defenses before they are rendered obsolete.

Today, businesses that conduct operations online (which is almost a requirement in order to remain consumer friendly) are targeted by motivated threat actors seeking to steal customer's personal and private data, and to obtain business's intellectual property for a competitive advantage. Small-medium businesses (SMB) have gone out of business as their bank accounts have been drained. Businesses that accept credit cards online or at Point of Sale (POS) machines, are the target of fraudsters and organized cyber-criminals. Bank customers who are accustomed to checking their

account balance and making payments and money transfers using online banking are the target of fraudsters armed with banking Trojans/malware. Once customers, personal and identifiable information is compromised, customers are notified by the bank of the breach, customer accounts are suspended, and the security incident has to be reported to the data privacy officer(s) and released to public in accordance to the data breach notification law enforced in the specific country. For most consumer customers, banks will take liability for the fraud being committed and repay their customers for losses, while commercial customers might face lawsuits from their clients when they refuse to pay for their losses. When business are found negligent of not applying the standard security controls and found noncompliant with information security standards, they are also impacted with additional fines and audits. Often businesses suffer large data breaches despite being compliant with technology security standards and conducting regular audits by qualified security auditors. This fact also challenges the assumption that adopting traditional security measures, processes, and technology, and compliance checks are enough to protect businesses from cyber-attacks. The assumption that security measures are “good enough” is often backed by evidence of successfully testing networks, systems, and application software for vulnerabilities, which is a factor in reducing the opportunity for an attacker to exploit them in targeted attacks.

Today, the risk mitigation effectiveness of the traditional approach of compliance driven security is challenged by the emergence of new cyber-threats and the fact that these threats have increased in sophistication and damage potential, which have rendered several security measures used today as obsolete. The adoption of sophisticated attack tools also referred to as cyber-crime toolkits for cyber-criminals and fraudsters as well as increased sophistication in the type of attack techniques, procedures, and tactics used are among the causes of an increased number of security breaches and the resulting increased economic losses felt by businesses. These cyber-crime tools are often freely available for attackers to download over the Internet and ready to be used for specific attacks against targets, such as Distributed Denial of Service (DDoS). Attacks tools can be purchased in the black market or rented for a fee such as in the case of malware and botnets. This increased availability of high-tech cyber-crime tools at very low cost severely increases the risks that businesses face when protecting customers data and company intellectual property from these attacks.

Due to this increased level of risk caused by emerging cyber-threats, businesses today cannot base their security on compliance and evidence of assurance followed by traditional information security standards and processes. Chances are that several business today that have audited for information security compliance with ISO 27001, PCI-DSS and have traditional security measures in place can still be targeted by cyber-attacks and experience losses of confidential data and fraud. Public and private organizations whose services and business depend on the web to generate a significant part of their revenues cannot look at compliance alone for security, but also need to consider a risk management approach that is based upon threat analysis, attack modeling, and simulations. This multifaceted risk management approach will reveal the level of resilience to targeted attacks and aid in determining the necessary countermeasures for reducing the risks to a manageable level. In addition,

the analysis of threat actors, the modeling of attacks, and the execution of threat driven tests cannot be restricted to security practitioners but needs the collaboration of the application stakeholders that include information security officers, application architects, software developers, application architects, security testers, and business owners. Engineering systems and software that are resilient enough to withstand the impact of cyber-attacks is necessary, and requires organizations and businesses to adopt new processes such as risk-based threat modeling.

Many of the cyber-attacks occurring against web applications today are facilitated by exploitation of design flaws and security bugs in the applications, such vulnerabilities that are introduced because of coding errors in the software components. For this reason, a focus on identifying design flaws using threat modeling is critical and this is best done during the software engineering life cycles. Threat modeling is an activity that can be executed during the early stages of the Software Development Life Cycle (SDLC) to identify and remediate design flaws prior to coding and prior to security testing. The adoption of threat modeling in the SDLC is risk effective in building attack resilient software as well as cost effective, since it allows for identifying design issues as early as possible and provides time to make changes to the design before the application product is built. Today, there is a need to adopt a risk-based threat modeling process to engineer business critical web applications and software. For example, consider software that is used for credit card processing, software that is used in critical industrial systems, such as SCADA, and runs oil, gas, water, and electric utilities; manufacturing controls; traffic controls; and mission critical systems for the military. In the financial sector, this software is used to handle online banking, make payments, and trade stocks and bonds. A little bit closer to our everyday experience as consumers, consider software used for mobile payments and for online purchases, which processes and stores credit card information and other personal data.

The main question for security practitioners and risk managers today is how businesses can design and implement applications and products that are engineered to withstand cyber-attacks and yet be cost effective to build. This is the call for security practitioners to look at engineering software from the perspective of a risk manager, to understand the threats and types of attacks and be able to identify solutions that are cost effective, yet still able to mitigate the impact of attacks. There is also a call for “cyber-threat application security and software awareness” since businesses and organizations still focus on protecting the network infrastructure and the perimeter, and overlook how web and mobile applications are built and how they securely store and process sensitive customer and corporate data. Today it seems that there is disconnect among the information security practitioners and risk managers between the escalation of emerging cyber-threats and the effectiveness of the countermeasures implemented. This disconnect can be bridged by the adoption of new approaches, such as risk-based threat modeling. For threats that specifically target applications and enterprise software, it is important to build countermeasures into products during the software development life cycle, rather than bolt on security at the end. In order to understand how these attacks can be prevented and detected, the identification of countermeasures to mitigate threats needs to be driven by threat analysis and

modeling of the attacks. For awareness sake, when making the case of software security to executive management, you can make parallels between the software industry and the car industry. “If applications today were built as resilient to cyber security attacks as cars are built resilient to car accidents, we would have software that is built with security controls equivalent to that of the car air bag. The car still needs to be repaired, but the consumer, the data, is protected.”

Traditionally, threat modeling as methodology has been advocated by software security consultants to model threats to software and to identify design flaws that could be remediated during the SDLC. Examples of these threat modeling methodologies include Microsoft Threat Modeling that is based on categorizing threats as STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privileges). Software developers can use MS STRIDE methodology to design software with countermeasures for these threats such as data and channel encryption, digital signatures/hashes, mutual authentication, authentication, and authorization. This is certainly a good start, but it is not enough to design applications that can withstand sophisticated threats, such as one used today against business-critical software and web applications, the designs must be implemented. Consider the example of a threat that exploits the business logic of an application, such as a financial application that uses credit card data to validate the identity of a customer to conduct a specific financial transaction. An attacker can abuse this functionality to enumerate which credit card numbers previously stolen are valid so that they can be used for online purchases or counterfeit credit cards. This type of threat exploits abuse of functionality and logic flaws, and can be analyzed for specific threat actors and specific attacks using methods such as “use and abuse cases.” This type of threat can also create a set of attack vectors and attack-driven test cases that are based upon this dynamic type of analysis, not a static assumption of threats. Another important reason for a risk-based threat modeling is the modeling of attacks to derive a set of tests that can be used to emulate the attack, and identify the presence of vulnerabilities and design flaws that need to be remediated. This modeling of attack starts by considering the product surface that is the available point of attacks for a threat actor such as the data interfaces and data channels. An attacker will seek to compromise the application by identifying the path of least resistance, exploring different channels that lead to the data assets, such as online, mobile, B2B channels, and in the cloud where data is either stored or processed.

A threat model can be used to emulate a real attack and test critical application functionality and software. To be realistic, the threat model needs to imitate the threat actors, tools, and attack techniques used, in order to derive a set of test cases that can be used by security testers to test the application resilience. This book advocates the use of risk based threat modeling, which is the analysis of threats and modeling of attacks in the context of information security and management of application and software as business assets. The main drive for the adoption of risk centered threat modeling is that using threat analysis and attack modeling allows risk managers to focus on the emerging threats to determine which countermeasures are most effective in mitigating these threats. Such a risk-based threat modeling process “bakes in”

all the essential ingredients of compliance; threat analysis, business impact analysis, software security, and risk management; and can be proven in the field by application security practitioners and risk managers.

Though there is not a silver bullet or a single solution for the complexities of software development, the authors offer a new application threat-modeling methodology, the "Process for Attack Simulation and Threat Analysis" (PASTA), which is documented in this book. PASTA is a risk-centered threat-modeling process that focuses on understanding first and foremost the business context and inherent risk profile of the application that needs to be secured. Secondly, PASTA factors threats and attacks and risk managers in designing web and mobile applications that are resilient to the emerging cyber-threats. As application security is a journey and not a destination, I also hope that the risk-based threat modeling methodology documented in this book will be useful as one of the ways to mitigate risks of the numerous emerging threats targeting your web applications and software.

HON. HOWARD A. SCHMIDT





# PREFACE

“The Senate determined to bring eight legions into the field, which had never been done at Rome before, each legion consisting of five thousand men besides allies. ... Most of their wars are decided by one consul and two legions, with their quota of allies; and they rarely employ all four at one time and on one service. But on this occasion, so great was the alarm and terror of what would happen, they resolved to bring not only four but eight legions into the field.”

*Polybius, The Histories of Polybius*

Battle of Cannae in 216 BC [1] when Hannibal employed defense in depth in order to encircle and destroy 10 Roman Legions all at once, resulting in the largest single slaughter of Roman troops in the history of the republic. Edward Luttwak used the term to describe his theory of the defensive strategy employed by the Late Roman army in the third and fourth centuries AD.

This book introduces the Process for Attack Simulation and Threat Analysis (PASTA) threat modeling methodology, an asset, or risk-centric approach. Its purpose is to provide a framework for risk mitigation based upon viable threat patterns against various types of threats. This book was written to usher in a new approach on threat analysis and risk mitigation. Both the methodology and the book have been inspired by more than 50 years of collective IT and Information Security experience where lackluster risk management measures and predictable security testing has yielded bloated and ineffective responses to instill organic security controls. The PASTA methodology is for both IT and Security professionals alike who recognize that there is no such thing as a “risk-free” utopia. The methodology appeals to IT, Security, Compliance, and Risk leaders who want to mitigate the residual risks that

matter and understand the causal threat factors that make them relevant in the first place.

This book intends to illustrate how the impact, attributed to threat scenarios, has never been properly addressed. It shares the status quo problem of risk today and how risk management today is simply the shuffling of best guesses and control gaps that do not speak to the heart of the risk equation. While there are many threat modeling methodologies, PASTA presents a step-by-step, risk-centric threat modeling approach that is centered around understanding business impact, focused on threat research, and concerned about countermeasures that truly demonstrate risk reduction. PASTA is an iterative, maturing process that can be measured and aligned to several different frameworks and existing best practices. Its design centers on the understanding that threat motives and targeted attacks are truly unpredictable and require a more sophisticated method for identifying their possible target assets. PASTA is supported by a logical consideration around attack patterns, and considers the multiple ways in which threat successes can be achieved across a myriad of targeted exploits. With this broad understanding, PASTA aims to provide a linear approach to attack simulation while considering impact levels attributed to compromised data, infrastructure, and even reputation.

From CISOs to Security Engineers, this book provides a wrapper to enterprise security processes that work together under the framework of PASTA. We hope you may consider a risk-centric approach to threat modeling as your next evolution to targeted threat analysis and response.

## REFERENCE

1. Polybius, Friedric Otto Hultsch (1889). *The Histories of Polybius*, Vol. 1, Macmillan and Company.

# LIST OF FIGURES

1.1	Relating Environmental Factors to Attacks	11
1.2	Developing Metrics in Threat Modeling	25
1.3	Development Factors Affecting Scalability	25
1.4	Cyber Crime Motives	34
1.5	Simple Data Flow Diagram supporting Threat Model	35
1.6	More Evolved Data Flow Diagram supporting Threat Model	36
1.7	STRIDE Threat Classification Visual Example	39
1.8	Incorporating Vulnerabilities within the Threat Model	40
1.9	Vulnerability Mapping	42
1.10	Sample Attack Tree	48
1.11	Deriving Risk via the Application Threat Model	60
2.1	Example of Use Case Diagram 1	85
2.2	Manual and Automated Vulnerability Assessments	106
2.3	Example of Data Flow Diagram	110
2.4	Root Causes versus Symptoms	115
3.1	Essential Process Areas for Threat Modeling	139
3.2	Security Areas for Greater Unity via Threat Modeling	141
3.3	Process Overview of Vulnerability Assessment Integration to Threat Modeling	147
3.4	Building Security Process in System/Network Administration from Threat Modeling	152
3.5	Security Centric DFD for Distributed Attacks	159

3.6	Components Represented by DREAD Risk Model	168
3.7	Stages of PASTA Threat Modeling Methodology	173
3.8	Cone of Fire Encompassing Multiple Targets	176
3.9	Relationship among Assets, Use Cases, Actors in Application Decomposition	181
3.10	Interrelated Asset Variables within an Application Environment	182
3.11	Factors Influencing Attacks	183
4.1	Threat Tree	203
4.2	Use and Misuse Case of User Log-on	208
4.3	Sketched Architectural Diagram	210
4.4	Data Flow Diagram	212
4.5	Mapping Threats Vulnerabilities and Countermeasures	213
4.6	RUP SDLC	218
4.7	Integrating Security in the Agile SDLC	220
4.8	Integrating Security in the Agile Sprints	222
4.9	Integration of Threat Modeling in MS SDL	224
4.10	SDL Phases	227
4.11	Generic Online Banking Application Threat Model	232
5.1	HPY Stock Price at the Time of the Data Breach Disclosure (January 20, 2009 <a href="http://datalossdb.org">datalossdb.org</a> )	243
5.2	Characterization of Risk by considering Threats, Vulnerabilities, and Assets	262
5.3	Five (5) Level Risk Calculation Heat Map	266
5.4	Threat-Vulnerability-Asset Risk Calculation Heat Map	268
5.5	Overall Threat-Vulnerability Domain	279
5.6	PASTA Threat Modeling Phases and Activities	285
5.7	Risk Calculation and Management Heat Map	293
5.8	NIST Risk Assessment mapping to Application Threat Modeling	299
5.9	Dissecting Cyber-Threats	302
5.10	Phases of Security Incident Handling Process (NIST via Coordinated Response)	309
6.1	Impacting Factors Across PASTA: A Checklist for Success	320
6.2	Threat Modeling Team Selection	323
6.3	Business Cross Section of a Threat Modeling Team	324
6.4	IT Operations Cross Section of a Threat Modeling Team	325
6.5	Security Operations Cross Section of a Threat Modeling Team	327
6.6	GRC Cross Section of a Threat Modeling Team	329
6.7	Givens Before PASTA Walk-Through	337
6.8	PASTA RACI Model	341

7.1	Deriving Use Cases from Business Objectives	348
7.2	Converging Security, Compliance, and Privacy Requirements in Stage I	350
7.3	Hierarchy of Objectives Addressed by PASTA	354
7.4	Relating Compliance to Business Impact	359
7.5	Business and InfoSec Balance in Stage I	363
7.6	PASTA Roles for Stage I	364
7.7	PASTA Risk-Centric Threat Modeling – Stage I – (DO) Definition of the Objectives	367
7.8	Software/Data Enumeration Containers	370
7.9	Stage III Application Containers	379
7.10	PASTA Risk-Centric Threat Modeling – Stage II – (DTS) Definition of the Technical Scope	392
7.11	Enumeration of Use Cases, Services, Stored Procedures, Batch Scripts, and Actors	393
7.12	Use Case to Application Component Mapping	395
7.13	Common Syntax of Symbols for DFDS	400
7.14	Data Flow Authentication Example	401
7.15	Data Flow for Data Exchange Across Two Entities	401
7.16	DFD Example Using Physical Boundaries for Organizing Components	403
7.17	Whiteboard DFD of User Self-Enrollment	404
7.18	DFD Health-Care Example Using Container Approach	405
7.19	DFD Using Architectural Considerations for Component Grouping	406
7.20	Spectrum of Trust for Defining Trust Boundaries Across Architecture	409
7.21	Decomposing Mobile Web App Example	412
7.22	API from Stores Local Transaction Server with the Following Metadata	413
7.23	PASTA Risk-Centric Threat Modeling – Stage III – (ADA) Application Decomposition and Analysis	417
7.24	Areas to Consider around Threat Evaluation	421
7.25	Sample Threat Possibilities per Industry	423
7.26	Mapping of Threat Agents to Asset Targets	436
7.27	PASTA Risk-Centric Threat Modeling – Stage – IV (TA) Threat Analysis	440
7.28	Missing Architectural Countermeasures among Application Components	449
7.29	Abuse Cases & Vulnerability Branch to Attack Tree Added	450
7.30	Logical Flow Considering Threats to Assets to Vulnerabilities	454
7.31	Targeted Application Testing in Web Applications	455

7.32	PASTA Risk-Centric Threat Modeling– Stage V – (WVA) Weakness and Vulnerability Analysis	458
7.33	Linearly Thinking about Attack Patterns	460
7.34	Snapshot of Related Control from CAPEC ID in Library	463
7.35	Completed Attack Tree	465
7.36	MITRE CAPEC Library Snapshot – CAPEC 117	466
7.37	Vulnerability Portion of Attack Tree	469
7.38	Attack Pattern Portion of Attack Tree	469
7.39	PASTA Risk-Centric Threat Modeling – Stage VI – (AMS) Attack Modeling and Simulation	470
7.40	Visualization of Attack and Countermeasures	472
7.41	Data Flow Diagram With Architectural Risk Analysis of Vulnerabilities and Controls	473
7.42	Completed Attack Tree w/Countermeasures	474
7.43	PASTA Risk-Centric Threat Modeling – Stage VII – (RAM) Risk Analysis and Management	477
8.1	PASTA Threat Modeling: Stages and Activities	481
8.2	Entering Business Functional Requirements/Use Cases Using the ThreatModeler™ Threat Modeling Tool	491
8.3	ThreatModeler™ Tool Wizard Capturing the Level of Risk for the Project HackMe Bank	497
8.4	HackMe Bank Users	509
8.5	Representation of a Bank Account Query Transaction Through the Different Tiers of an Online Banking Application	510
8.6	Internal Services Deployed with the Application Architectural Components	512
8.7	ThreatModeler™ Association of Widgets with Client Components	512
8.8	Architecture of Online Banking Application	514
8.9	Component-Based Functional Use Cases of Online Web Application	519
8.10	Data Flow Diagram for Online Banking Application	521
8.11	Functional Component Trust Boundaries Using ThreatModeler™	523
8.12	Campaign of DDoS Attacks Against Banking Sites Announced by AQCF Threat Agent Group	532
8.13	Ontology of (STIX) Structured Language for Cyber Threat Intelligence Information (Courtesy of MITRE Corp)	537
8.14	Example of Kill-Chain (Courtesy of MITRE corp)	541
8.15	Web Incident Hacking Database Attack Library	542
8.16	ThreatModeler™ Tool Threat Library	543
8.17	Threat Model Using STRIDE per Element	546
8.18	Threat Risk Factors	549

8.19	Threat Dashboard with Threat Risk and Status	549
8.20	OSVDB Open Source Vulnerability Database source <a href="http://www.osvdb.org">http://www.osvdb.org</a>	555
8.21	Architectural Risk Analysis Component of ThreatModeler™	560
8.22	Architectural Risk Analysis of Authorization Controls	560
8.23	Threat Tree (Source OWASP)	562
8.24	Mapping of Threats with Vulnerabilities of Different Application IT Assets	563
8.25	Number of Attack Observed in 6 Months by Imperva 2013 WAAR	565
8.26	Test Cases to Validate Vulnerabilities at Component Functional Level ThreatModeler™	568
8.27	Sequence of Events Followed in Banking Trojan Attacks	576
8.28	Anatomy of Account Takeover and Fraudulent Wire Transfer	577
8.29	Attack Vectors Used in Banking Trojan Malware, Source OWASP Anti-Malware Knowledge Base	578
8.30	CVEs Exploited by Drive-By-Download Attacks	593
8.31	CAPEC Attack Pattern for HTTP DoS	595
8.32	Engineering for Attacks Source MITRE	598
8.33	WHID Attack Library in ThreatModeler™	599
8.34	Banking Malware Attack Tree	605
8.35	Use and Abuse Cases for MFA Controls	608
8.36	Threat-Level Security Test Cases	613
8.37	Threat and Risk Dashboard	623
8.38	Risk Calculation Heat Map	624
8.39	ThreatModeler™ Threat-Risk Management Dashboard	625
8.40	ThreatModeler™ Threats to Functional Components and Security Controls That Mitigate These Threats	626
8.41	High Level View of Threats-Attacks-Vulnerabilities-Countermeasures of Online Banking Application	627





# LIST OF TABLES

1.1	Correlating Environmental Factors to Attack Motives – SAMPLE	12
1.2	Correlating Motives to Application Threat Vectors	16
1.3	Recommended Frequency for Environmental Threat Factor Analysis	17
1.4	Key Reasons App_Sec Fails Today	20
1.5	Threat Modeling Benefits for Various Roles	27
1.6	Threat Model Stack	35
1.7	Taxonomy of Attack Terms	46
1.8	Tools for Testing	54
1.9	Elements of Risk – Generic Listing of Key Risk Components	58
2.1	Application Security Roles, Responsibilities, and Benefits	69
2.2	Example of Threats and the Technical and Business Impacts	74
2.3	Criteria for Threat Modeling Scope	92
2.4	Criteria for Application Threat Modeling Updates	93
2.5	Mapping of Threats to Vulnerabilities	132
3.1	Example of Mapping Threat Modeling Efforts to Security Processes	143
3.2	Security Experience Meets Threat Modeling	148
3.3	Factors Affecting Time Requirements for Threat Modeling	155
3.4	DFD Symbols (Microsoft ACE Team) (59)	156
3.5	Traditional Network-Based Denial of Service Attacks	163
3.6	STRIDE Threat Categorization Table (60)	164
3.7	Example of STRIDE Classification Model	166
3.8	Threat Rating Table Example	169

3.9	Sample Risk Rating Exercise Using DREAD	169
3.10	DREAD Risk Rating Applied to Sample Threat	170
3.11	Security Objectives in support of Business Objectives	175
3.12	Application Decomposition for Mobile J2ME App	180
3.13	MITRE's Security Content	189
5.1	Example of Assignment of Risks Of A Threat Event based upon probability of the event and impact on the asset	265
6.1	Enterprise Process Mapping to PASTA Threat Modeling Methodology	334
6.2	Artifacts for Making PASTA	338
7.1	Relating Business Objectives to Security Requirements	346
7.2	Enumeration of Business Requirements to Understood Use Cases	349
7.3	Governance Artifacts Relevant to Stage I of PASTA	352
7.4	Considerations for Factoring Business Impact	357
7.5	Possible Inherent Risk Issues by Application Type	361
7.6	Simple CRUD Mapping Across a Product Application	373
7.7	Software Enumeration from Automated Tools	375
7.8	Free Hardening Guidelines/Tools for Inherent Risk Mitigation or Blind Threat Modeling (Stage II – PASTA)	381
7.9	Sample Identification of Use Cases for Health-Care Application	394
7.10	Hypothetical Functional Requirements/Objectives for Marketing Application	397
7.11	Deriving Use Cases from Functional Requirements	399
7.12	Sample Threat Considerations for Various Applications	422
7.13	VERIS Framework of IR Metrics	433
7.14	Threat Analysis of a Mobile Based Loan Application Serving Higher Ed	438
7.15	Threat Analysis for Bluetooth Enabled Medical Device	438
7.16	Threat Analysis Artifact against a Single Asset/ Use Case	443
7.17	Labeling Relevant Threat Modeling Variables during Targeted Assessment Efforts	456
7.18	Attack Considerations for POS at Restaurants	461
7.19	Residual Risk Analysis	476
8.1	Sensitive Data Analysis and Business Requirements of Online Banking Application	492
8.2	Online Banking Application Risk Profile	500
8.3	Online Banking Application Components S/W Technology Stack	508
8.4	Online Banking Web Application: Data Interfaces	509
8.5	Security Function Transactional Analysis	525

8.6	Overall Cyber-Threat Scenarios Affecting Financial IT Systems and Applications	534
8.7	Structured Threat Information eXpression (STIX) Architecture vs 3.0	538
8.8	Example of Description of Browser Exploit Threat Using STIX	540
8.9	STRIDE Threat List	546
8.10	Application Security Frame	547
8.11	Secure Architecture Design Guidelines	559
8.12	Mapping of OWASP-WASC and CWE Source CriticalWatch: OWASP to WASC to CWE Mapping, Correlating Different Industry Taxonomy	565
8.13	Malware Banking Trojan Kill-Chain and Security Measures	588
8.14	Attack Vectors Used By Banking Malware	593
8.15	DDoS Attack Vectors Extracted from the Analysis of DDoS Attacks Against Web Applications	594
8.16	CAPEC SQL Injection Attack Sequence 1. Determine User-Controllable Input Susceptible to Injection	596
8.17	CAPEC SQL Injection Attack Sequence 1. 2. Experiment and try to exploit SQL Injection Vulnerability	596
8.18	CWEs Exploited in SQL Injection Attacks (CAPEC SQL Injection)	597
8.19	CAPEC-66 Security Requirements For Mitigation of Risk of SQL Injection Attacks	597
8.20	Attack Surface of Online Banking Application	601
8.21	Malware-Based-Attack-Driven Security Test Cases	610
8.22	DDoS Attack Driven Security Test Cases	612
8.23	Security Measures Proposed for Mitigate the Risks of Malware Banking and DDoS Threats	628



---

# 1

---

## THREAT MODELING OVERVIEW

### DEFINITIONS

[Application] Threat Modeling – a **strategic process** aimed at considering possible **attack** scenarios and **vulnerabilities** within a proposed or existing **application environment** for the purpose of clearly identifying **risk** and **impact** levels.

Definitions for any type of terminology are necessary evils. While seemingly elementary and potentially annoying, they provide a common ground from which to build. Providing a well-constructed definition also level-sets threat modeling's intended design as a process-oriented control for application security, versus interpretations that mutate its intent and true capability.

In this book, the expression “threat modeling” is reserved for software development and application security efforts. Within the topical boundaries of application security, the aforementioned definition provides some fundamental terms that should resonate with anyone who understands the very nature of security risk management and has implemented the threat modeling machine.

A closer examination of the definition provided reveals greater insights into the essential components that are threat modeling. The first emphasized term, *strategic*, describes a quality of threat modeling reflected in its ability to anticipate threats via

calculated and simulated attack patterns. Each major function within the threat modeling process requires a great deal of consideration and anticipation of multiple risk factors influenced by threat, vulnerability, and impact levels.

*Process* is one of threat modeling's key, distinguishing qualities. A chain-like reaction of tactical events is conducted across multiple domains (business objectives, system/database administration, vulnerability management, etc.) where additional review, input, and contribution is provided by other stakeholders within the process – all in relation to a protected application environment. To date, the lack of process within information security efforts has accounted for several shortcomings in mitigating security risks introduced by deficiencies in application security, and in many cases acted as causal factors to those noted deficiencies. Although there are isolated victories in traditional security efforts, a growing sentiment is that the war against software exploitation is being lost. Threat modeling is intended to greatly revitalize the effort in securing data via a collaborative, strategic process.

The next term, *attack*, reflects a major science to threat modeling – the discipline of researching how attack patterns can potentially exploit software vulnerabilities and/or poorly designed countermeasures. The hierarchy of an attack becomes dissected via threat modeling techniques, exposing faults in application design and/or software development, as well as other practical yet key areas, such as unveiling plausible motives for which an attacker initially sought to launch their assault.

*Vulnerabilities* is a term used far more prevalently within other information security efforts. In the scope of threat modeling, however, its use extends the manner in which software vulnerabilities are understood. Vulnerabilities at the platform and software levels are aggregated and correlated to possible attack scenarios. As a result, this term is an essential component to its definition, as we will see in later chapters.

The *application environment* expression serves as the object of the threat modeling process. Other traditional security procedures simply address a single aspect of an entire application environment, thereby negating a more holistic approach to application security. This is not to state that these more isolated procedures are not important, but rather that the sum of their individual benefits is encompassed in the process of threat modeling and applied to the entire application environment.

The term *risk* serves as the object of key interest to threat modeling. Threat modeling, as a supportive role in fulfilling business objectives, seeks to identify risks associated with the cumulative effects of an ever-evolving threat environment, compounded by software/network vulnerabilities, and fueled by attack motives or interest in business information – all managed and/or driven by an application environment. Threat modeling provides greater precision in conveying risk through providing a clear path on how a business application environment could be compromised and the probability of the actual risk. In essence, risk becomes the common glue that unifies security and business professionals in a collaborative effort to protect the enterprise.

Within the threat modeling definition, *impact* is the ability to answer the question “How bad is it?” Unless security professionals consider all possible threat scenarios in order to generate a prioritized, risk-based analysis, they cannot provide an effective and credible answer. As answers morph into speculations and continue downhill, security professionals are again unable to convey an adequate and plausible answer