Threat Modeling For Secure Software Design

OWASP Rhode Island
June 9, 2016
Robert Hurlbut

RobertHurlbut.com • @RobertHurlbut
Robert Hurlbut

Software Security Consultant, Architect, and Trainer

Owner / President of Robert Hurlbut Consulting Services
(ISC)2 CSSLP 2014-2017

Contacts

Web Site: https://roberthurlbut.com
Twitter: @RobertHurlbut
Secure Software Design

Building secure systems is difficult
Design is key
Build appropriate security through secure design
But, how? And why should we care?
Secure Software Design

Breakdowns in secure design:
GitHub Mass Assignment
Jeep Cherokee Hack
Target Breach
What is threat modeling?

Something we all do in our personal lives ...

... when we lock our doors to our house

... when we lock the windows

... when we lock the doors to our car
What is threat modeling?

When we think ahead of what could go wrong, weigh the risks, and act accordingly we are “threat modeling”
Threat modeling helps …

Bridge gaps between builders, breakers, and defenders:

- Helps builders focus on security features
- Helps breakers know most critical attack surfaces
- Helps defenders understand critical attack patterns
- Helps many areas security / business

© 2016 Robert Hurlbut Consulting Services
Where does threat modeling fit?

One of the security tools
We know about penetration testing, fuzzing, analysis / code reviews, detection (lots of automated tools)
Threat modeling is a process – a “way of thinking“ (not automated)
Tool useful for secure design
What is threat modeling?

Threat modeling is:
Process of understanding your system and potential threats against your system.
What is threat modeling?

**Threat model** includes:
- understanding of system,
- identified threat(s),
- proposed mitigation(s),
- priorities by risk
Michael Howard @michael_howard
Jan 7, 2015

A dev team with an awesome, complete and accurate threat model gets my admiration and not much of my time because they don’t need it! 😊
As I practice it, threat modeling cannot be the province of a tech elite. It is best owned by all of a development team.
Definitions

Threat Agent

Someone (or a process) who could do harm to a system (also adversary or attacker)
Definitions

Threat
An adversary’s goal
Definitions

Vulnerability
A flaw in the system that could help a threat agent realize a threat
Definitions

Attack
When a motivated and sufficiently skilled threat agent takes advantage of a vulnerability
Definitions

Asset

Something of value to valid users and adversaries alike
When?

Make threat modeling first priority:
In SDLC – Requirements and Design phase
Threat modeling uncovers new requirements
Agile Sprint Planning
Teach your developers to threat model
When?

What if we didn’t?

It’s **not too late** to start threat modeling (generally)

It will be more difficult to change major design decisions

Do it anyway!
Typical Threat Modeling Session

Gather documentation

Gather your team:
- Developers, QA, Architects, Project Managers, Business Stakeholders (not one person’s job!)

Understand business goals and technical goals (threat modeling must support goals, not other way around)

Agree on meeting date(s) and time(s)

Plan on 1-2 hour focused sessions at a time

**Important:** Be honest, leave ego at the door, no blaming!
Simple Tools

Whiteboard
Visio (or equivalent) for diagraming
Word (or equivalent) or Excel (or equivalent) for documenting

Look at Dinis Cruz’ Simple Threat Model One Page Template and Concepts

http://blog.diniscruz.com/2016/05/threat-modeling-template-and-concepts.html
Simple Threat Model – One Page

[Diagram of a Threat Model template]

© 2016 Robert Hurlbut Consulting Services

Simple Threat Model – Concepts

© 2016 Robert Hurlbut Consulting Services

## Threat Model Sample Worksheet

<table>
<thead>
<tr>
<th>ID</th>
<th>Risk Level (H, M, L)</th>
<th>Threat</th>
<th>Description / Impact</th>
<th>Countermeasures</th>
<th>Components Affected</th>
<th>Follow Up Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2016 Robert Hurlbut Consulting Services
Other Tools

Microsoft Threat Modeling Tool 2016
ThreatModeler – Web Based (in-house) Tool
ThreadFix
IriusRisk Software Risk Manager
Review Security Principles

1. Secure the weakest link
2. Defend in depth
3. Fail securely
4. Grant least privilege
5. Separate privileges
6. Economize mechanisms

Review Security Principles*

7. Do not share mechanisms
8. Be reluctant to trust
9. Assume your secrets are not safe
10. Mediate completely
11. Make security usable
12. Promote privacy
13. Use your resources

IEEE Computer Society’s Center for Secure Design

Take a look at:


© 2016 Robert Hurlbut Consulting Services
Threat Modeling Process

1. **Draw your picture** – understand the system and the data flows
2. **Identify threats through answers to questions**
3. **Determine mitigations and risks**
4. **Follow through**
Draw your picture
Understand the system

DFD – Data Flow Diagrams (MS SDL)

- External Entity
- Process
- Multi-Process
- Data Store
- Dataflow
- Trust Boundary

© 2016 Robert Hurlbut Consulting Services
Understand the System

Understand logical and component architecture of system
Understand every communication flow and valuable data moved and stored
Understand the system

1. User
   - Request
   - Response

2. Authn Service
   - Authn Request
   - Authn Info
   - Get Creds

3. Credentials
   - Set/Get Creds

4. Web App
   - Request
   - Requested Audit Data File(s)
   - Audit Data
   - Authn Info
   - Audit Requests

5. Mnmgt Tool
   - Set User Data
   - Verify User Data

6. Data Files
   - Requested Audit Data

7. Audit Service
   - Audit Read
   - Audit Write
   - Audit Requests

8. Audit Service
   - Audit Info

9. Admin
   - Set User Data

(Trust boundary)
Understand the system

External Entities:
Users, Admin

Processes:
Web App, Authn Svc, Audit Svc, Mnmgt Tool

Data Store(s):
Data Files, Credentials

Data Flows:
Users <-> Web App
Credentials
Admin <-> Audit Svc

© 2016 Robert Hurlbut Consulting Services
Your threat model now consists of …

1. Diagram / understanding of your system and the data flows
Identify threats

Most important part of threat modeling (and most difficult)
Many ways – determine what works best for your team
Identify threats

Attack Trees
  Bruce Schneier - Slide deck

Threat Libraries
  CAPEC, OWASP Top 10, SANS Top 25

Checklists
  OWASP ASVS, OWASP Proactive Controls

Use Cases / Misuse Cases
Identity threats - Games

OWASP Cornucopia

Suits:
Data validation and encoding
Authentication
Session Management
Authorization
Cryptography
Cornucopia

13 cards per suit, 2 Jokers
Play a round, highest value wins
### STRIDE Framework – Data Flow

<table>
<thead>
<tr>
<th>Threat</th>
<th>Property we want</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>S</strong>poofing</td>
<td><strong>A</strong>uthentication</td>
</tr>
<tr>
<td><strong>T</strong>ampering</td>
<td><strong>I</strong>ntegrity</td>
</tr>
<tr>
<td><strong>R</strong>epudiation</td>
<td><strong>N</strong>on-repudiation</td>
</tr>
<tr>
<td><strong>I</strong>nformation Disclosure</td>
<td><strong>C</strong>onfidentiality</td>
</tr>
<tr>
<td><strong>D</strong>enial of Service</td>
<td><strong>A</strong>vailability</td>
</tr>
<tr>
<td><strong>E</strong>levation of Privilege</td>
<td><strong>A</strong>uthorization</td>
</tr>
</tbody>
</table>

© 2016 Robert Hurlbut Consulting Services
Identity threats - Games

Elevation of Privilege (EoP)

The EoP game focuses on the following threats (STRIDE):

- Spoofing
- Tampering
- Repudiation
- Information Disclosure
- Denial of Service
- Elevation of Privilege
Identify threats

P.A.S.T.A. – Process for Attack Simulation and Threat Analysis (combining STRIDE + Attacks + Risk Analyses)
Identify Threats – Functional

Input and data validation
Authentication
Authorization
Configuration management
Sensitive data
Identify Threats – Functional

Session management
Cryptography
Parameter manipulation
Exception management
Auditing and logging

© 2016 Robert Hurlbut Consulting Services
Identity Threats - Ask Questions

Who would be interested in the application and its data (threat agents)?
What are the goals (assets)?
What are attack methods for the system we are building?
Are there any attack surfaces exposed - data flows (input/output) we are missing?
Identity Threats – Ask Questions

How is authentication handled between callers and services?
What about authorization?
Are we sending data in the open?
Are we using cryptography properly?
Is there logging? What is stored?
Etc.
One of the best questions …

Is there anything keeping you up at night worrying about this system?
Scenario – Configuration Management

1. User
2. Authn Service
3. Credentials
4. Web App
5. Mnmgt Tool
6. Data Files
7. Audit Service
8. Audit Service
9. Admin

Trust Boundaries:
- (Trust boundary)
- (Trust Boundary)
Scenario – Configuration Management

Data Files such as configuration files

© 2016 Robert Hurlbut Consulting Services
Scenario – Configuration Management

System: Web application uses configuration files

Security principles:
  Be reluctant to trust, Assume secrets not safe

Questions:
  How does the app use the configuration files?
  What validation is applied? Implied trust?

Possible controls/mitigation:
  Set permissions on configuration files.
  Validate all data input from files. Use fuzz testing to insure input validation.
Your threat model now consists of …

1. Diagram / understanding of your system and the data flows

2. Identify threats through answers to questions
Determine mitigations and risks

Mitigation Options:
- Leave as-is
- Remove from product
- Remedy with technology countermeasure
- Warn user

What is the risk associated with the vulnerability?
Determine mitigations and risks

Risk Management

FAIR (Factor Analysis of Information Risk) – Jack Jones

Risk Rating (High, Medium, Low)
Risk Rating

Overall risk of the threat expressed in High, Medium, or Low.
Risk is product of two factors:
  Ease of exploitation
  Business impact
## Risk Rating – Ease of Exploitation

<table>
<thead>
<tr>
<th>Risk Rating</th>
<th>Description</th>
</tr>
</thead>
</table>
| **High**    | - Tools and exploits are readily available on the Internet or other locations  
- Exploitation requires no specialized knowledge of the system and little or no programming skills  
- Anonymous users can exploit the issue |
| **Medium**  | - Tools and exploits are available but need to be modified to work successfully  
- Exploitation requires basic knowledge of the system and may require some programming skills  
- User-level access may be a pre-condition |
| **Low**     | - Working tools or exploits are not readily available  
- Exploitation requires in-depth knowledge of the system and/or may require strong programming skills  
- User-level (or perhaps higher privilege) access may be one of a number of pre-conditions |
## Risk Rating – Business Impact

<table>
<thead>
<tr>
<th>Risk Rating</th>
<th>Description</th>
</tr>
</thead>
</table>
| **High**    | • Administrator-level access (for arbitrary code execution through privilege escalation for instance) or disclosure of sensitive information  
• Depending on the criticality of the system, some denial-of-service issues are considered high impact  
• All or significant number of users affected  
• Impact to brand or reputation |
| **Medium**  | • User-level access with no disclosure of sensitive information  
• Depending on the criticality of the system, some denial-of-service issues are considered medium impact |
| **Low**     | • Disclosure of non-sensitive information, such as configuration details that may assist an attacker  
• Failure to adhere to recommended best practices (which does not result in an immediately visible exploit) also falls into this bracket  
• Low number of user affected |
## Example – Medium Risk Threat

<table>
<thead>
<tr>
<th>ID - Risk</th>
<th>RT-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threat</td>
<td>Lack of CSRF protection allows attackers to submit commands on behalf of users</td>
</tr>
<tr>
<td>Description/Impact</td>
<td>Client applications could be subject to a CSRF attack where the attacker embeds commands in the client applications and uses it to submit commands to the server on behalf of the users</td>
</tr>
<tr>
<td>Countermeasures</td>
<td>Per transaction codes (nonce), thresholds, event visibility</td>
</tr>
<tr>
<td>Components Affected</td>
<td>CO-3</td>
</tr>
</tbody>
</table>
Scenario – Configuration Management

Data Files such as configuration files

Web App

Requested File(s)

Data Files

(Trust boundary)
Scenario – Configuration Management

System: Web application uses configuration files

Security principles:
   Be reluctant to trust, Assume secrets not safe

Questions:
   How does the app use the configuration files?
   What validation is applied? Implied trust?

Possible controls/mitigation:
   Set permissions on configuration files.
   Validate all data input from files. Use fuzz testing to insure input validation.

Risk Rating:
   We own the box (Medium/Low), Hosted on cloud (High)
Your threat model now consists of …

1. Diagram / understanding of your system and the data flows
2. Identify threats through answers to questions
3. Mitigations and risks identified to deal with the threats
Follow through

Document what you found and decisions you make
File bugs or new requirements
Verify bugs fixed and new requirements implemented
Did we miss anything? Review again
Anything new? Review again
Your threat model now consists of …

1. Diagram / understanding of your system and the data flows
2. Identify threats through answers to questions
3. Mitigations and risks identified to deal with the threats
4. Follow through

A living threat model!
Your challenge

Use threat modeling for:
secure design before new features
driving your testing and other review activities
understanding bigger picture
Resources - Books

**Threat Modeling: Designing for Security**
*Adam Shostack*

*Brook S.E. Schoenfield*

**Risk Centric Threat Modeling: Process for Attack Simulation and Threat Analysis**
*Marco Morana and Tony UcedaVelez*

**Measuring and Managing Information Risk: A FAIR Approach**
*Jack Jones and Jack Freund*
Resources - Tools

Microsoft Threat Modeling Tool 2016

ThreatModeler – Web Based (in-house) Tool
http://myappsecurity.com

ThreadFix

IriusRisk Software Risk Manager
https://iriusrisk.continuumsecurity.net
Resources - Tools

Attack Trees – Bruce Schneier on Security

Elevation of Privilege (EoP) Game

OWASP Cornucopia
https://www.owasp.org/index.php/OWASP_Cornucopia

OWASP Application Security Verification Standard (ASVS)

OWASP Proactive Controls 2016
https://www.owasp.org/index.php/OWASP_Proactive_Controls
Questions?

Contacts
Web Site: https://roberthurlbut.com
Twitter: @RobertHurlbut
Email: robert at roberthurlbut.com