Boston Application Security Conference (BASC) 2019

Threat Modeling Workshop

October 19, 2019
Robert Hurlbut
@RobertHurlbut
Who am I?

Robert Hurlbut
SVP, Threat Modeling Architect / Lead Cyber Security Technology
Bank of America
Legal Disclaimer

- Neither Bank of America nor its affiliates provide information security or information technology (IT) consulting services.
- This material is provided "as is," with no guarantee of completeness, accuracy, timeliness or of the results obtained from the use of this material, and without warranty of any kind, expressed or implied, including, but not limited to warranties of performance, quality and fitness for a particular purpose.
- This material should be regarded as general information on information security and IT considerations and is not intended to provide specific information security or IT advice nor is it any substitute for your own independent investigations.
- If you have questions regarding your particular IT system or information security concerns, please contact your IT or information security advisor.
Agenda

What is Threat Modeling?

Threat Modeling Process

Hands-on Exercises / Labs

What’s next?
What is Threat Modeling?
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
When we ...

think ahead on what could go wrong 
(i.e. the “what if” questions),
weigh the risks,
and act accordingly ...

... we are “threat modeling”
What is threat modeling?, continued

Threat Modeling: Designing for Security by Adam Shostack
https://threatmodelingbook.com/

Asks four questions:

1. What are you working on?
2. What could go wrong?
3. What are you going to do about it?
4. Did you do a good job of analysis?
What is threat modeling?, continued

**Threat modeling** is:

Process of understanding your system and potential threats against your system and related countermeasures

i.e. *Critical Thinking* about Security
What is threat modeling?, continued

Brook Schoenfield (10/15/2019):

“#threatmodeling is a technique to identify the attacks a system must resist and the defences that will bring the system to a desired defensive state” – Secrets Of A Cyber Security Architect, due Fall 2019 or Winter 2020*

* (Quoted from a tweet by Brook Schoenfield: https://twitter.com/BrkSchoenfield/status/1184178678990888965)
You probably (hopefully!) already do these in your security strategy:

Penetration testing
Vulnerability assessments
DAST / SAST tools
Other automated tools ...

But, if not threat modeling – you are missing a lot!
Example Secure Design Issue: How to secure data in the cloud?

Storage?  
Accessed?  
Monitored?  
Configured properly?

*Threat Modeling helps us focus on these questions and answers to lead to secure design*
Common data breach problem

Misconfigured AWS S3 Buckets
Impacted in 2017-2018 *:

- FedEx
- GoDaddy
- Accenture
- Verizon
- American voter data (198 million American voters)
- National Credit Federation
- Booz Allen Hampton
- Dow Jones
- Keeper and Blur (password managers)

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset</td>
<td>Something of value we want to protect</td>
</tr>
<tr>
<td>Threat Agent</td>
<td>Someone or process who could do harm</td>
</tr>
<tr>
<td>Threat</td>
<td>Exploits vulnerabilities (intentional or accidental) to obtain, damage, or destroy an asset</td>
</tr>
<tr>
<td>Vulnerability</td>
<td>Flaw in system helps threat agent realize threat</td>
</tr>
<tr>
<td>Risk</td>
<td>Potential for loss, damage, destruction of asset from threat using vulnerability</td>
</tr>
<tr>
<td>Attack</td>
<td>Motivated and sufficiently skilled threat agent takes advantage of vulnerability</td>
</tr>
</tbody>
</table>
Threat Modeling Vocabulary*


(John Steven – formerly of Synopsys)
Approaches to Threat Modeling

Asset-centric

Software-centric

Attacker-centric
Approaches to Threat Modeling – Asset-centric

Assets

Things of value. For example: Databases which may contain credit card data, personal Identifiable Information (PII), etc.

Attack trees

- Attacker gets DB Access
  - SQL Injection
  - Web Server compromise
  - Browser may contain content
  - Authorization failure
    - Apply port scanner
    - Implement data validation
    - Implement HTTP headers
    - Implement authorized checks
Secure Design

Understanding secure activity within an architecture

Data Flow Diagrams (DFDs)
## Approaches to Threat Modeling – Attacker-centric

<table>
<thead>
<tr>
<th>Profiles</th>
<th>Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Script Kiddie</td>
<td>Copies scripts – tries anything</td>
</tr>
<tr>
<td>Hacktivist</td>
<td>Political agenda – deface website</td>
</tr>
<tr>
<td>Nation-state attacker</td>
<td>Money, intellectual property theft - phishing</td>
</tr>
</tbody>
</table>
Threat Modeling your House

Asset-centric
Family, irreplaceable photos, valuable artwork

Software-centric
Physical features (basement door, porch)

Attacker-centric
Who might break in, current security system
What is threat modeling?

**Threat model** includes:
understanding of system,
identified threat(s),
proposed mitigation(s),
priorities by risk
Threat Model – Example – Simple Web Application

**Understanding of system** (requirements, a diagram, etc.)

**Identify threats** - what could go wrong?
- Open HTTP connections -> attacker sees data in transit (Information disclosure)
- Open HTTP connections -> attacker changes data in transit (Tampering)
- Broken authentication -> attacker pretends to be someone else (Spoofing)
- Etc.

**Proposed mitigations:**
- HTTPS (encrypted connections)
- Strong authentication (2FA, centralized)

**Priorities by Risk:**
- Which one do you fix first?
When? Make threat modeling first priority

In SDLC – Requirements and Design phase(s):

**Requirements** > **Design** > Development > Test > Deployment

Threat modeling  ->  new requirements

Incremental threat modeling  ->  Agile / DevOps (User Stories, Attacker Stories)
Teach threat modeling to your teams

Training

Help / Model

Encourage

Follow Up
Threat Modeling: Getting Started
Typical Threat Modeling Session

Domain Knowledge Team
Business / Technical Goals Focused

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

Whiteboard

Visio (or equivalent) – diagraming

Word (or equivalent) / Excel (or equivalent) - documenting threats / mitigations
<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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Other Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Cost</th>
<th>Platforms</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS Threat Modeling Tool</td>
<td>Free</td>
<td>Windows OS Install only</td>
</tr>
<tr>
<td>ThreatModeler</td>
<td>Paid</td>
<td>Web Based</td>
</tr>
<tr>
<td>IriusRisk</td>
<td>Paid</td>
<td>Web Based</td>
</tr>
<tr>
<td>OWASP Threat Dragon</td>
<td>Free</td>
<td>Web Based / Windows, Mac, Linux installs</td>
</tr>
<tr>
<td>Draw.IO</td>
<td>Free</td>
<td>Web Based / Windows, Mac, Linux installs</td>
</tr>
</tbody>
</table>
IEEE Computer Society’s Center for Secure Design (2015) *

Avoiding the Top 10 Software Security Design Flaws: Bugs vs Flaws

Bug – an implementation-level software problem

Flaw – deeper level problem - result of mistake or oversight at design level

In Threat Modeling, we try to identify design flaws to improve secure design
Threat Modeling Process
Threat Modeling Process

1. Diagram / understand your system and data flows
2. Identify threats through answers to questions
3. Determine mitigations and risks
4. Follow through
Threat Modeling Process: Diagram / understand
Draw a picture

Good – but not enough. Let’s explore further.
Understand the system

DFD – Data Flow Diagrams (MS SDL)

External Entity

Data Store

Process

Multi-Process

Dataflow

Trust Boundary

October 19, 2019
Understand the system - create a Data Flow Diagram (DFD)

How do the entities, processes and data stores connect? Connect the info points with the data flow arrows.
Where are the trust boundaries?
For example:

• Browser (entity) sends / receives data (data flow) with a web application (process) which saves / reads data (data flow) using a SQL Database (data store)
• Web application (process) reads (data flow) web configuration file (data store)
• Trust boundaries indicate where trust changes — authenticate / authorize / validate
Your threat model now consists of ...

1. Diagram / understand your system and data flows
Threat Modeling Lab 1: Review case study
Build data flow diagram (DFD)
Threat Modeling Process:
Identify threats
Identify threats – Many Ways

STRIDE
Attack Trees
Bruce Schneier - Slide deck
Threat Libraries
CAPEC, ATT&CK, OWASP Top 10, SANS Top 25
Checklists
OWASP ASVS, OWASP Proactive Controls
Card Games
OWASP Cornucopia, Elevation of Privilege
Use Cases / Misuse Cases
Misuse Cases help with ...

No one would ever do that!
Why / who would ever do that?!
## STRIDE* Framework – Data Flow

<table>
<thead>
<tr>
<th>Threat</th>
<th>Examples</th>
<th>Property we want</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Spoofing</strong></td>
<td>Pretending to be someone else</td>
<td>Identity Assurance</td>
</tr>
<tr>
<td><strong>Tampering</strong></td>
<td>Modifying data that should not be modifiable</td>
<td>Integrity</td>
</tr>
<tr>
<td><strong>Repudiation</strong></td>
<td>Claiming someone didn’t do something</td>
<td>Non-repudiation</td>
</tr>
<tr>
<td><strong>Information Disclosure</strong></td>
<td>Exposing information</td>
<td>Confidentiality</td>
</tr>
<tr>
<td><strong>Denial of Service</strong></td>
<td>Preventing a system from providing service</td>
<td>Availability</td>
</tr>
<tr>
<td><strong>Elevation of Privilege</strong></td>
<td>Doing things that one isn’t suppose to do</td>
<td>Least Privilege</td>
</tr>
</tbody>
</table>

* STRIDE was invented by Loren Kohnfelder and Praerit Garg (1999)
Applying STRIDE to a DFD

Each part of STRIDE applies to specific elements or interactions.

You can also look at STRIDE per interaction.
Identify threats — Mapping STRIDE to DFD

<table>
<thead>
<tr>
<th>Threats</th>
<th>Data Flows</th>
<th>Data Stores</th>
<th>Processes</th>
<th>Entities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spoofing</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Tampering</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Repudiation</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Information Disclosure</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Denial of Service</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Elevation of Privilege</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>
Identify Threats – Functional

Input and data validation
Authentication
Authorization
Configuration management
Data Classification

- Public, Proprietary, Confidential
Identify Threats – Functional

Session management
Cryptography
Parameter manipulation
Exception management
Auditing, logging, and monitoring
Who’s interested in app and data (threat agents)?

What goals (assets)?

What attack methods (how)?

Any attack surfaces (trust boundaries) exposed?

Any input/output (data flows) missing?
Is there anything keeping you up at night worrying about this system?
Scenario – Configuration Management

Internet

Browser

HTTPS Request
HTTPS Response

Web App

Web Config

File Read

SQL Read

SQL Save

Data

Internal Company

SQL Database
Scenario – Configuration Management

Data Files such as configuration files
Scenario – Configuration Management

System: Web application uses configuration files

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

Questions to identify threats:
How does the app use the configuration files?
What validation is applied?
Implied trust?
Can anyone update / change the files?
Your threat model now consists of ...

1. Diagram / understand your system and data flows
2. Identify threats through answers to questions
Threat Modeling Lab 2: Identify threats
Threat Modeling Process:
Determine mitigations and risks
Determine mitigations and risks – Mitigations mapped to STRIDE

<table>
<thead>
<tr>
<th>STRIDE</th>
<th>Example mitigations</th>
</tr>
</thead>
</table>
| Identity Assurance (Spoofing) | • Authentication based on key exchange  
• Decide on single-factor, two-factor, or multi-factor authentication  
• Offload authentication to another provider  
• Restrict authentication to certain IP ranges or locations |
| Integrity (Tampering)    | • Data protected from tampering with cryptographic integrity mechanisms  
• Only enumerated authorized users may modify data |
| Non-Repudiation (Repudiation) | • Maintain logs  
• Digital signature |
| Confidentiality (Information Disclosure) | • Data in files / database will only be available to authorized users  
• Name / existence of database will only be exposed to authorized users  
• Content and existence of communication between Alice and Bob will only be exposed to these authorized users |
| Availability (Denial of Service) | • Rate limiting or throttling access to a service  
• Real-time monitoring of log files and other resources to note sudden changes |
| Least Privilege (Elevation of Privilege) | • System has a central authorization engine  
• Authorization controls stored with item being controlled using ACLs  
• System limits who can write data to higher integrity level  
• System uses roles / accounts or permissions to manage access |
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 and threat identified?
Determine mitigations and risks

Risk Management

FAIR (Factor Analysis of Information Risk) – Jack Freund, 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>3 - Medium</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Threat</strong></td>
<td>Lack of CSRF protection allows attackers to submit commands on behalf of users</td>
</tr>
<tr>
<td><strong>Description/Impact</strong></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><strong>Countermeasures</strong></td>
<td>Per transaction codes (nonce), thresholds, event visibility</td>
</tr>
<tr>
<td><strong>Components Affected</strong></td>
<td>CO-3</td>
</tr>
</tbody>
</table>
Scenario – Configuration Management

System: Web application uses configuration files

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

Questions to identify threats:
   How does the app use the configuration files?
   What validation is applied?
   Implied trust?
   Can anyone change / update the files?

Possible controls / mitigations:
   Set permissions on configuration files.
   Validate all data input from files.
   Use fuzz testing to insure input validation.
Scenario – Configuration Management

System: Web application uses configuration files

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

Questions to identify threats:
- How does the app use the configuration files?
- What validation is applied?
- Implied trust?
- Can anyone change / update the files?

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

Risk Rating:

On-Premises (Medium/Low) vs. Cloud (High)
Your threat model now consists of ...

1. **Diagram / understand your system and data flows**
2. **Identify threats through answers to questions**
3. **Determine mitigations and risks**
Threat Modeling Lab 3: Determine mitigations
Threat Modeling Process: Follow through
Follow through

Document findings and decisions

File bugs or new requirements

Verify bugs fixed / new requirements implemented

Did we miss anything? Review again

Anything new? Review again
Your threat model now consists of ...

1. Diagram / understand your system and data flows
2. Identify threats through answers to questions
3. Determine mitigations and risks
4. Follow through

A living threat model!
What next?
What next?

Look at tools that can help take you further (DFDs):
• MS Threat Modeling Tool
• OWASP Threat Dragon
• Draw.IO – see Michael Enriksen’s article:
  https://michenricksen.com/blog/drawio-for-threat-modeling
What next?, continued

Learn more about:
  • Attack Trees
    – Bruce Schneier’s 1999 article
  • Incremental Threat Modeling
    – Agile approaches – Irene Michlin (@IreneMichlin)
  • Lateral Movement
    – “The Industrial Revolution for Lateral Movement” BlackHat 2017
  • Using MITRE ATT&CK for Threat Modeling
    – Brook Schoenfield “Secrets Of A Cyber Security Architect”, due Fall 2019 or Winter 2020
What next?, continued

Learn more about:
• List vs Graph Thinking
• Recursive Threat Modeling
– John Lambert (@JohnLaTwC) at Microsoft

Modern defenders know security controls create attack surface. Beware the attack graph you make practicing InfoSec:

Beware the Attack Surface of InfoSec by @JohnLaTwC
Traditional defenders see security controls as solving InfoSec problems.
Attackers see security controls as an attack graph of points of compromise.
See Both.

© 2019 John Lambert
What next?, continued

Learn more about:

• Threat Modeling as Code
  – ThreatPlaybook (@abhaybhargav)
  – ThreatSpec (@ThreatSpec, @zeroXten)
  – PyTM, CTM (@izar_t)
BASC 2019
Effective Threat Modeling with CTM (Continuous Threat Modeling)
with
Izar Tarandach
3:00 pm – 4:50 pm
Conclusion

Get started with Threat Modeling today:

Start with secure design as goal

Ask the “what if” questions

Understand 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 - Books

Agile Application Security
Laura Bell, Michael Brunton-Spall, Rich Smith, Jim Bird

Upcoming books:

Secrets of a Cyber Security Architect (Fall, 2019 or Winter, 2020)
Brook S.E. Schoenfield

Threat Modeling (April, 2020)
Izar Tarandach, Matthew J. Coles
Resources - Tools

Microsoft Threat Modeling Tool
https://aka.ms/threatmodelingtool

ThreatModeler
https://threatmodeler.com

IriusRisk Software Risk Manager
https://iriusrisk.com/threat-modeling-tool/

OWASP Threat Dragon
https://www.owasp.org/index.php/OWASP_Threat_Dragon
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 Top 10 Proactive Controls 2018
https://www.owasp.org/index.php/OWASP_Proactive_Controls
Questions?

@RobertHurlbut
Thank you!