Threat Modeling Workshop

Threat Modeling Using OWASP Threat Dragon

AppSecUSA 2017 – Developer Summit

Robert Hurlbut

OWASP

The Open Web Application Security Project

Robert Hurlbut

OWASP
The Open Web Application Security Project

Software Security Consultant, Architect, and Trainer

Owner / President of Robert Hurlbut Consulting Services
Microsoft MVP – Developer Security 2005-2010, 2015-2018
(ISC)2 CSSLP 2014-2017
Co-host Application Security Podcast (@appsecpodcast)

Contacts

Web Site: https://roberthurlbut.com

LinkedIn: RobertHurlbut

Twitter: @RobertHurlbut

ROBERT HURLBUT CONSULTING SERVICES

securedevelop.NET

OWASP
The Open Web Application Security Project

Mikko Hypponen @mikko 9/15/2017 (repost from 2012 tweets)

"It's just not fair when the attackers cheat. They really should be regulated to attack our defenses only the way we want them to attack."

OWASP
The Open Web Application Security Project

Threat Modeling Concepts, Goals, Process, Labs:

a. Threat Modeling purpose, types, definitions
b. Typical Threat Modeling session
c. Threat Modeling Process
d. Threat Modeling Labs – Whiteboard / OWASP Threat Dragon

OWASP
The Open Web Application Security Project

Determine requirements
Determine features
Build software people will use

OWASP
The Open Web Application Security Project

Determine secure requirements

Determine secure features

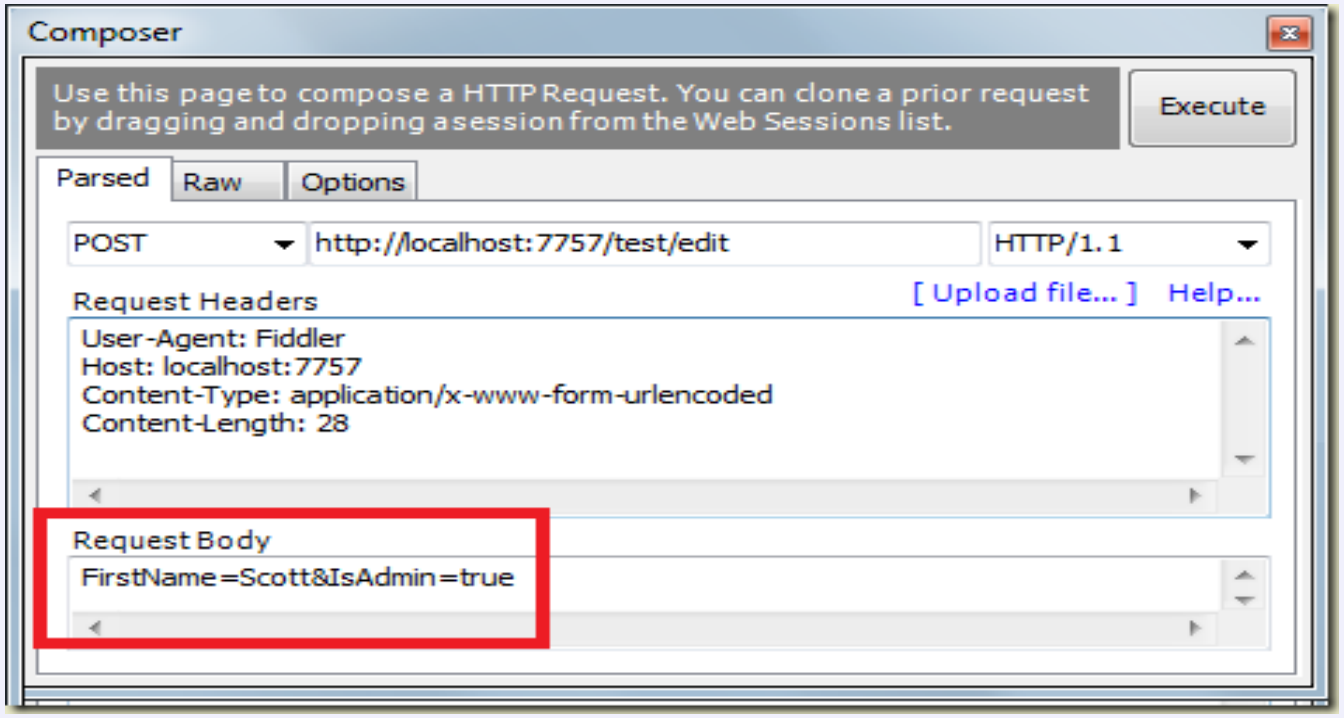
Build software people will use

… and will anticipate mis-use

But, how? And why should we care?

OWASP
The Open Web Application Security Project

GitHub - Mass Assignment



OWASP
The Open Web Application Security Project

Jeep Cherokee Hack

https://youtu.be/ysAam9Zmdv0

Jeep Cherokee Hack – Stopping Cars

OWASP
The Open Web Application Security Project

OWASP
The Open Web Application Security Project

Pacemakers recall - unauthenticated access

Hacking risk leads to recall of 500,000 pacemakers due to patient death fears

FDA overseeing crucial firmware update in US to patch security holes and prevent hijacking of pacemakers implanted in half a million people

Abbott / St Jude Medical's Accent MRI pacemaker, one of the affected devices that had to be recalled. Photograph: Abbott / St Jude Medical

https://www.theguardian.com/technology/2017/aug/31/hacking-risk-recall-pacemakers-patient-death-fears-fda-firmware-update

What is threat modeling?

OWASP
The Open Web Application Security Project

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?

OWASP
The Open Web Application Security Project

When we ...

think ahead on what could go wrong,
weigh the risks,

and act accordingly ...

... we are “**threat modeling**”

What is threat modeling?

OWASP
The Open Web Application Security Project

Historically, threat modeling came from military usage:

- Who is the enemy?
- What are their motives?
- What are their methods?
- Let's plan our strategy / defense

Where does threat modeling fit?

OWASP
The Open Web Application Security Project

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?

OWASP
The Open Web Application Security Project

Threat modeling is:

Process of understanding your system and potential threats against your system

What is threat modeling?

OWASP
The Open Web Application Security Project

Threat model includes:

understanding of a system,

identified threat(s),

proposed mitigation(s),

priorities by risk

OWASP
The Open Web Application Security Project

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! ☺"

OWASP
The Open Web Application Security Project

Brook Schoenfield @BrkSchoenfield June 29, 2015

"As I practice it, threat modeling cannot be the province of a tech elite. It is best owned by all of a development team."

Threat modeling helps …

OWASP
The Open Web Application Security Project

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

OWASP
The Open Web Application Security Project

Asset

Something of value to valid users and adversaries alike

Threat Agent

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

Definitions

OWASP
The Open Web Application Security Project

Threat

Anything that will exploit a vulnerability, intentionally or accidentally, and obtain, damage, or destroy an asset

Vulnerability

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

OWASP
The Open Web Application Security Project

Threat <> Vulnerability

If a vulnerability is not present, neither is the threat

but …

when the vulnerability is present, threat can be realized.

OWASP
The Open Web Application Security Project

Risk

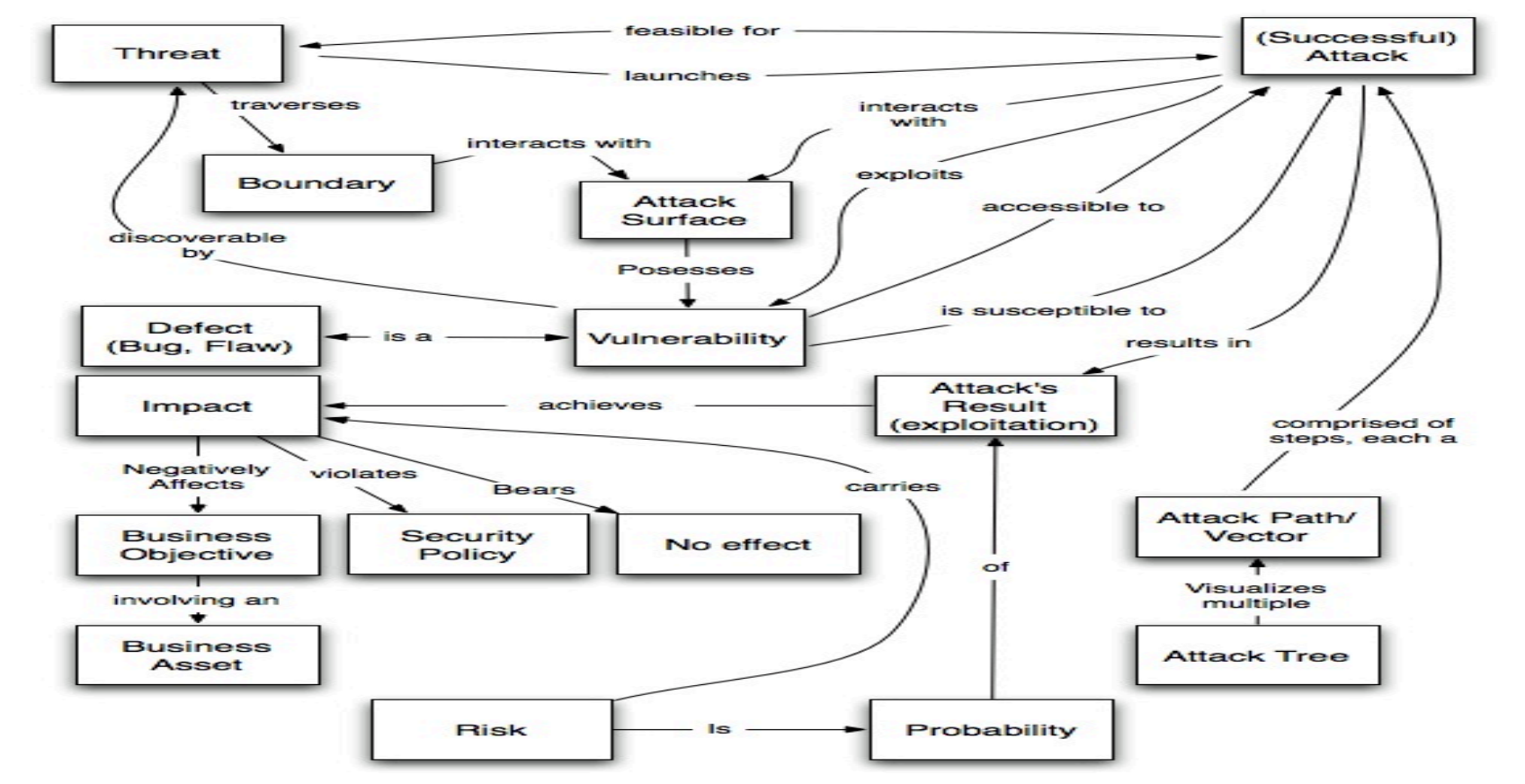
The potential for loss, damage, or destruction of an asset as a result of a threat exploiting a vulnerability

Attack

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

Threat Modeling Vocabulary*

OWASP
The Open Web Application Security Project



* https://www.cigital.com/blog/threat-modeling-vocabulary/ (John Steven, Synopsis)

Approaches to Threat Modeling

OWASP
The Open Web Application Security Project

Software-centric

- Secure design, DFDs

Asset-centric

- Attack trees

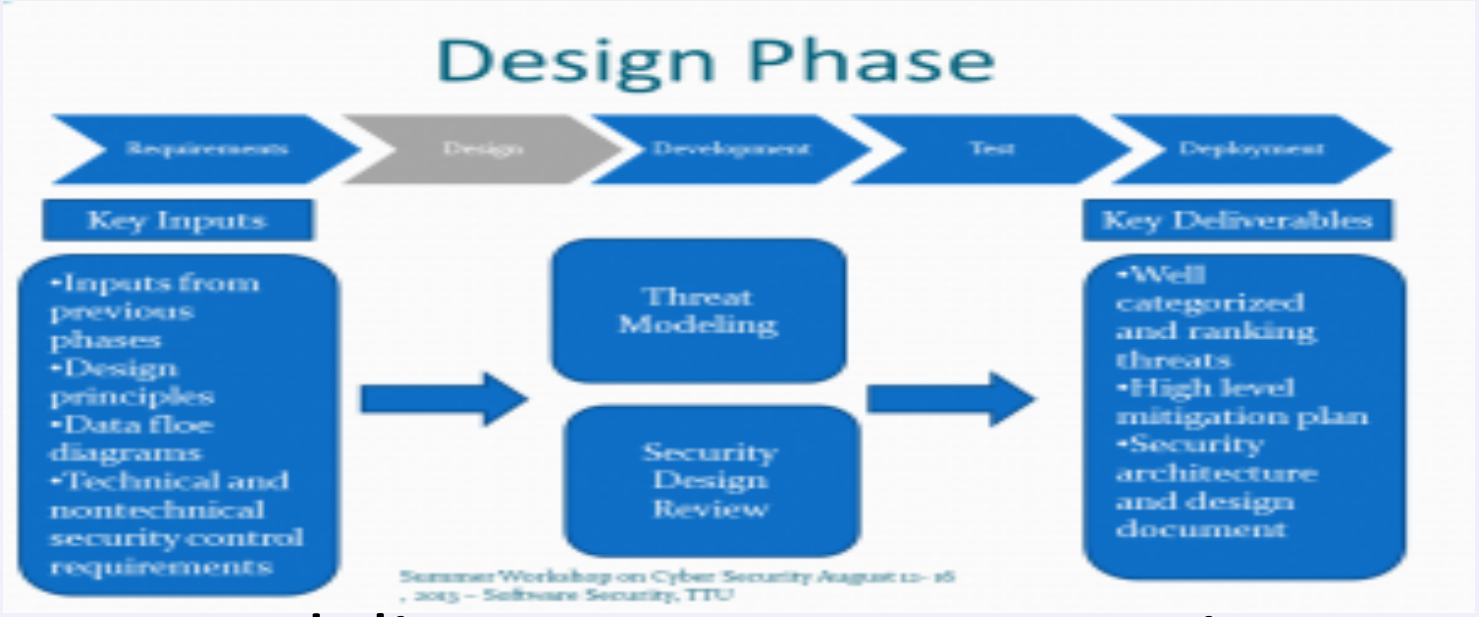
Attacker-centric

- Profile, patterns

When? Make threat modeling first priority

OWASP
The Open Web Application Security Project

In SDLC – Requirements and Design phase

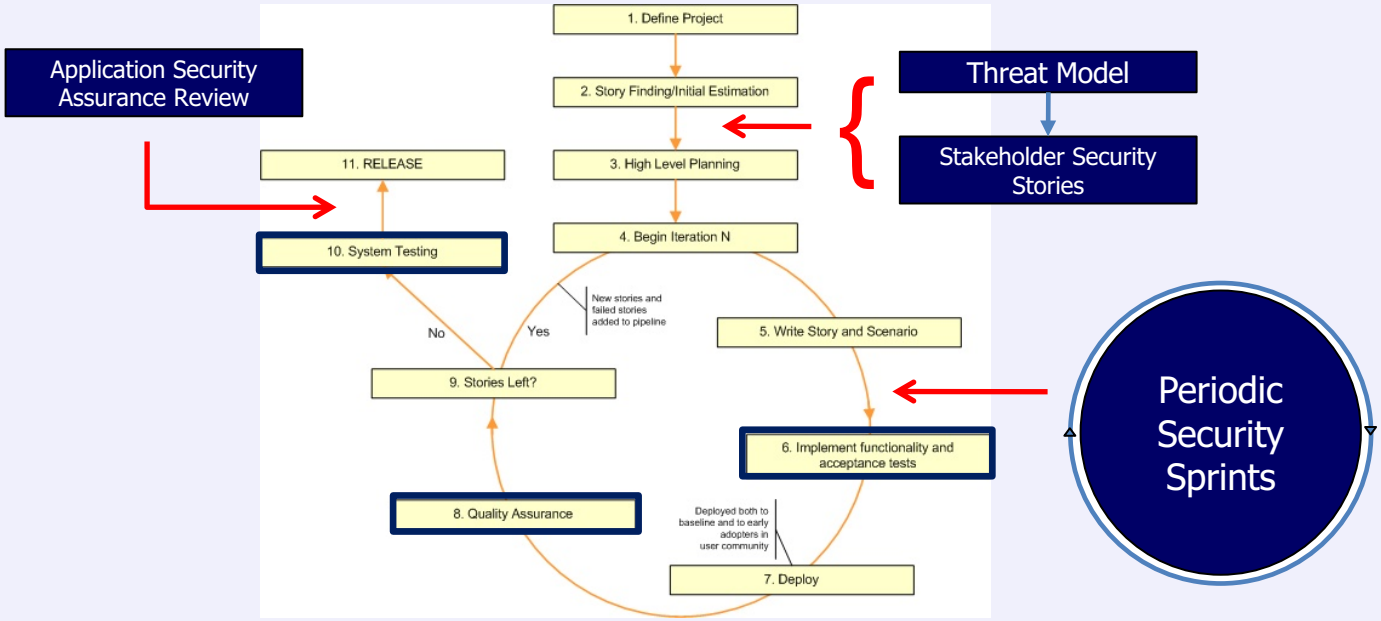


Threat modeling uncovers new requirements

When? Make threat modeling first priority

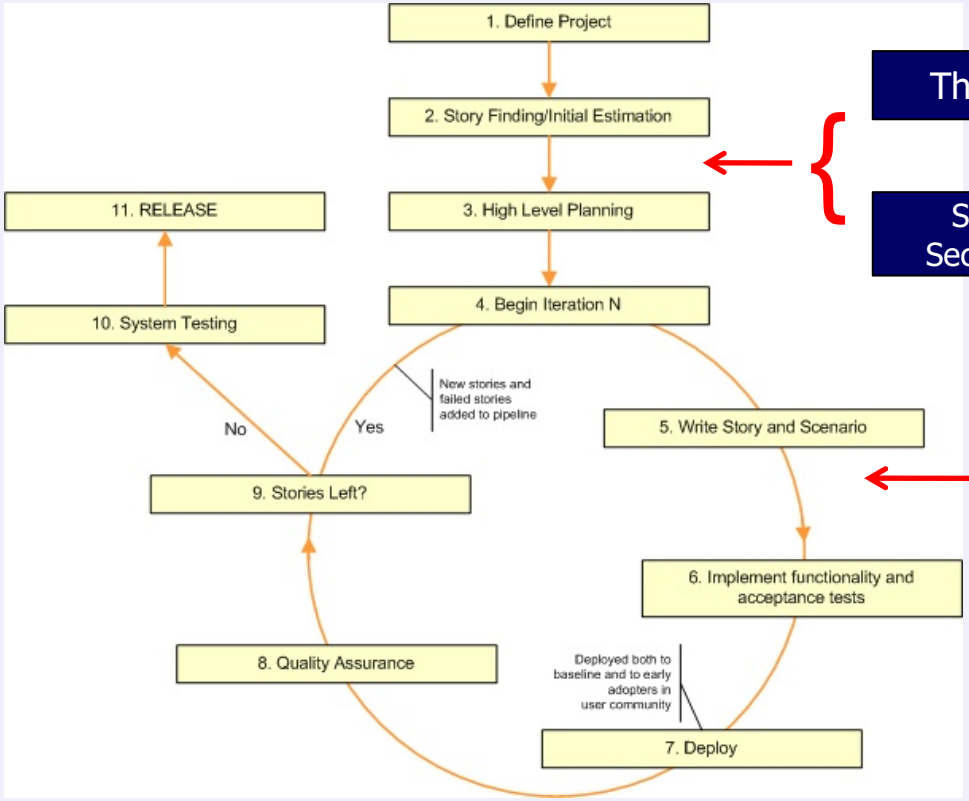
OWASP
The Open Web Application Security Project

Agile Sprint Planning - User Stories, Attacker Stories



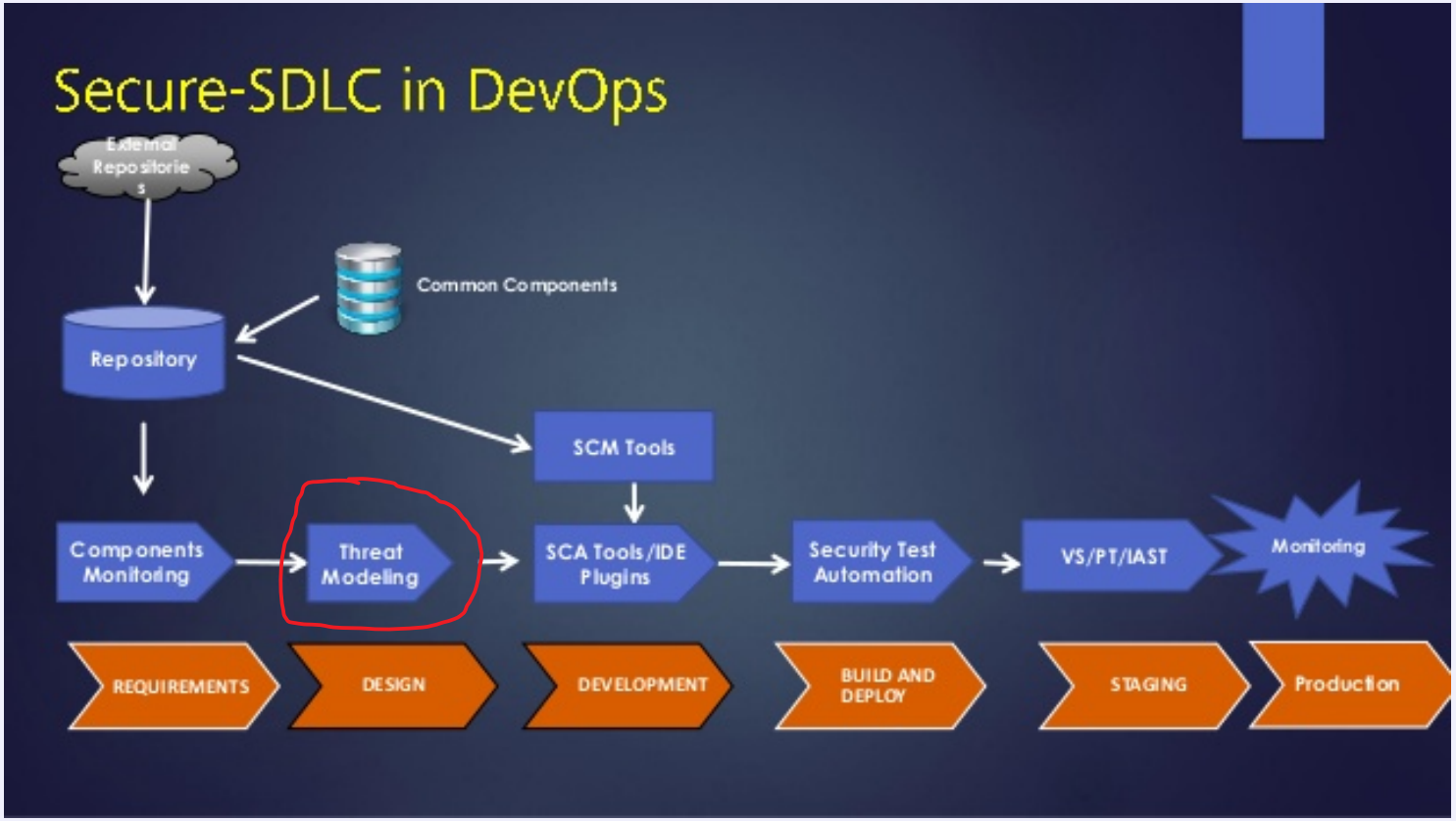
Agile Threat Modeling!

OWASP
The Open Web Application Security Project



Threat Modeling in DevOps

OWASP
The Open Web Application Security Project



When?

OWASP
The Open Web Application Security Project

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

OWASP
The Open Web Application Security Project

Domain Knowledge

Team

- Developers, QA, Architects, Project Managers, Business Stakeholders (not one person's job!)

Business / Technical Goals

- Threat modeling must support goals, not other way around

Meeting Date(s) / Time (s) / 1-2 hour focused sessions

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

Simple Tools

OWASP
The Open Web Application Security Project

Whiteboard

Visio (or equivalent) for diagraming

Word (or equivalent) or Excel (or equivalent) for documenting

Simple Threat Model – One Page

OWASP
The Open Web Application Security Project

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*

OWASP
The Open Web Application Security Project

5/17/2016 draw.io

Threat Model

Application Name: ...
Section: ...
JIRA Project: ....
Version: ....

This Threat Model represents
....

DFD (Data Flow Diagram)

Entrypoints

URL , Port, Service
....

Assets

Data
....

Threats

STRIDE	Description	JIRA #
....	....	...

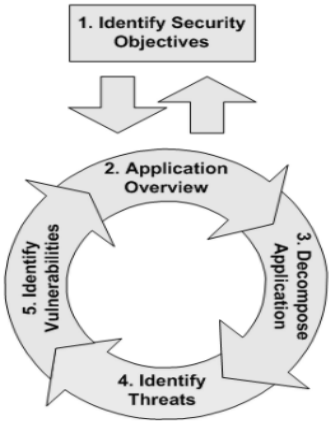
chrome-extension://pebppomjfocnoigkeepgbmcifnnlndla/index.html 1/1

* https://github.com/DinisCruz/Security-Research/blob/master/pdfs/Threat-Modeling/Template/Threat-Model-Template-v0.1.pdf

Simple Threat Model – Concepts*

OWASP
The Open Web Application Security Project

Threat Model Concepts



Vocabulary

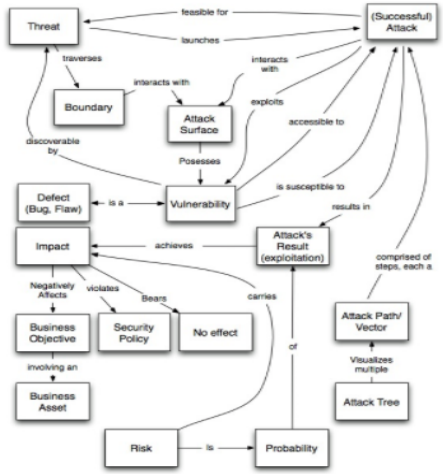


Image from https://www.cigital.com/blog/threat-modeling-vocabulary/

DFD Elements

External Entity	The external entity shape is used to represent any entity outside the application that interacts with the application via an entry point
Process	Represents a task that handles data within the application. The task may process the data or perform an action based on the
Multiple Process	Used to present a collection of subprocesses. The multiple process can be broken down into its subprocesses in another DFD.
Data Store	Represents locations where data is stored
Data Flow	Represents data movement within the application. The direction of the data movement is represented by the arrow.
Privilege Boundary	Represent the change of privilege levels as the data flows through the application.

Data Classification

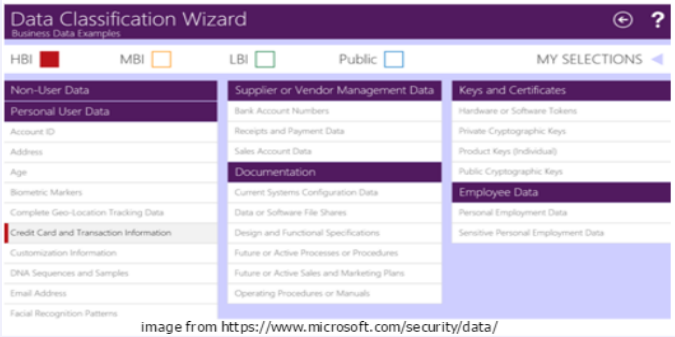


image from https://www.microsoft.com/security/data/

STRIDE

Threat	Description	Breaks
Spoofing	Pretending to be somebody else	Authentication
Tampering	Modifying data that should not be modifiable	Integrity
Repudiation	Claiming someone didn't do something	Non-Repudiation
Information Disclosure	Exposing information	Confidentiality
Denial of Service	Preventing a system from providing service	Availability
Elevation of Privilege	Doing things that one isn't supposed to do	Authorization

* https://github.com/DinisCruz/Security-Research/blob/master/pdfs/Threat-Modeling/Concepts/Threat%20Model%20Concepts-v0.2.pdf

Threat Model Sample Worksheet

OWASP
The Open Web Application Security Project

	A	B	C	D	E	F	G
1	**Threat Model Worksheet**						
2							
3	**ID**	**Risk Level (H, M, L)**	**Threat**	**Description / Impact**	**Countermeasures**	**Compenents Affected**	**Follow Up Plan**
4							
5							

Other Tools

OWASP
The Open Web Application Security Project

Microsoft Threat Modeling Tool 2016 (also 2017 preview available)

ThreatModeler – Web Based (in-house) Tool

ThreadFix

IriusRisk Software Risk Manager

OWASP Threat Dragon (new in 2017)

Review Security Principles*

OWASP
The Open Web Application Security Project

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

(*"Thirteen principles to ensure enterprise system security" by Gary McGraw, 2013, http://cs.brown.edu/courses/cs180/static/files/lectures/readings/lecture12/thirteen_principles.pdf)

Review Security Principles*

OWASP
The Open Web Application Security Project

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

(*"Thirteen principles to ensure enterprise system security" by Gary McGraw, 2013, http://cs.brown.edu/courses/cs180/static/files/lectures/readings/lecture12/thirteen_principles.pdf)

OWASP
The Open Web Application Security Project

Take a look at:

http://www.computer.org/cms/CYBSI/docs/Top-10-Flaws.pdf

Bugs vs Flaws

OWASP
The Open Web Application Security Project

Bug – an implementation-level software problem

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

Secure Design Flaws

OWASP
The Open Web Application Security Project

1. Incorrect trust assumptions
2. Broken authentication mechanisms that can be bypassed or tampered with
3. Neglecting to authorize after authentication
4. Lack of strict separation between data and control instructions, and as a result processing control instructions received from an untrusted source
5. Not explicitly validating all data
6. Misuse of cryptography
7. Failure to identify sensitive data and how they should be handled
8. Failure to consider the users
9. Misunderstanding how integrating external components change an attack surface
10. Brittleness in the face of future changes made to objects and actors

Secure Design Recommendations

OWASP
The Open Web Application Security Project

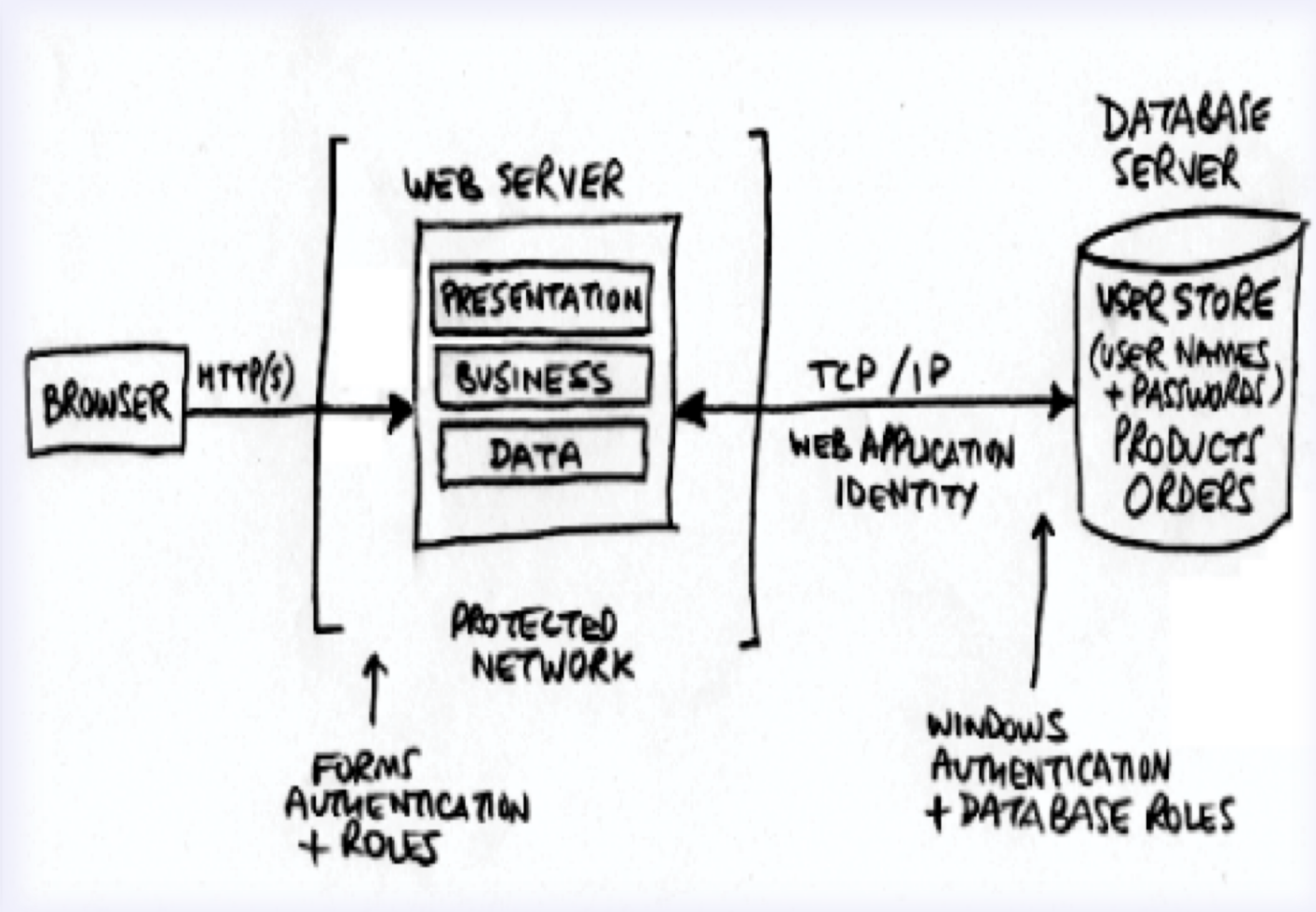
1. Earn or give, but never assume, trust
2. Use authentication mechanism that cannot be bypassed or tampered with
3. Authorize after you authenticate
4. Strictly separate data and control instructions, and never process control instructions received from untrusted sources
5. Define an approach that ensures all data are explicitly validated
6. Use cryptography correctly
7. Identify sensitive data and how they should be handled
8. Always consider the users
9. Understand how integrating external components changes your attack surface
10. Be flexible when considering future changes to objects and actors

Threat Modeling Process

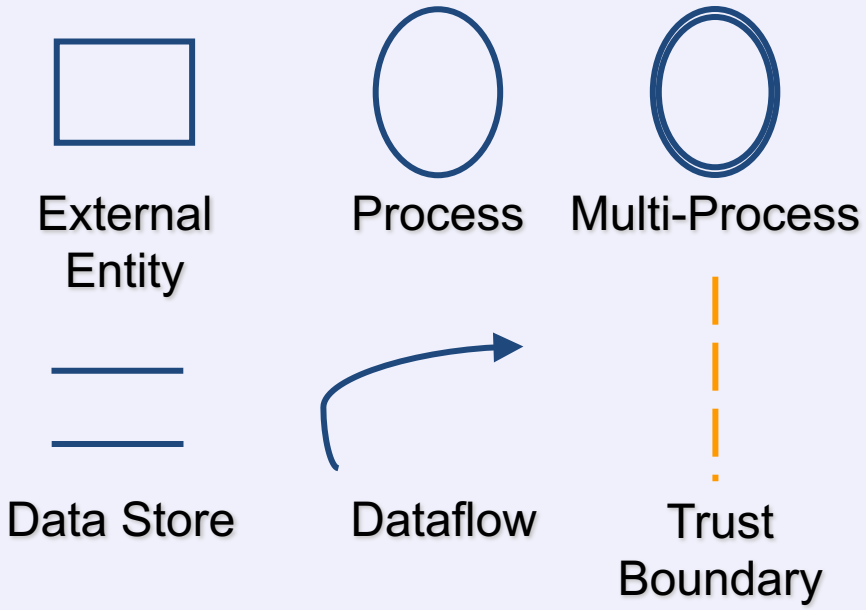
OWASP
The Open Web Application Security Project

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



DFD – Data Flow Diagrams (MS SDL)

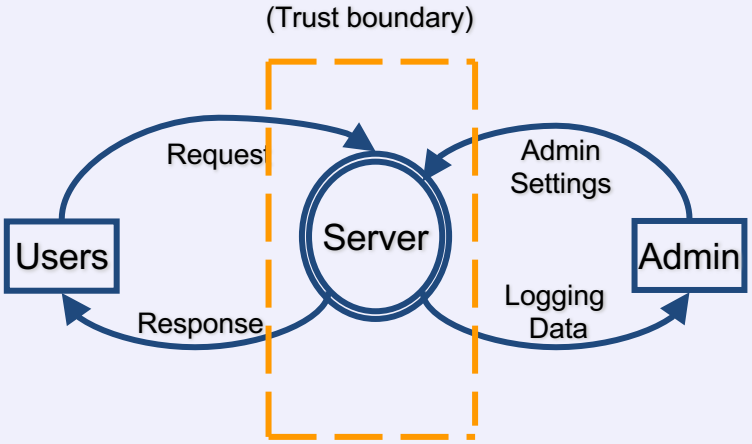


Understand the System

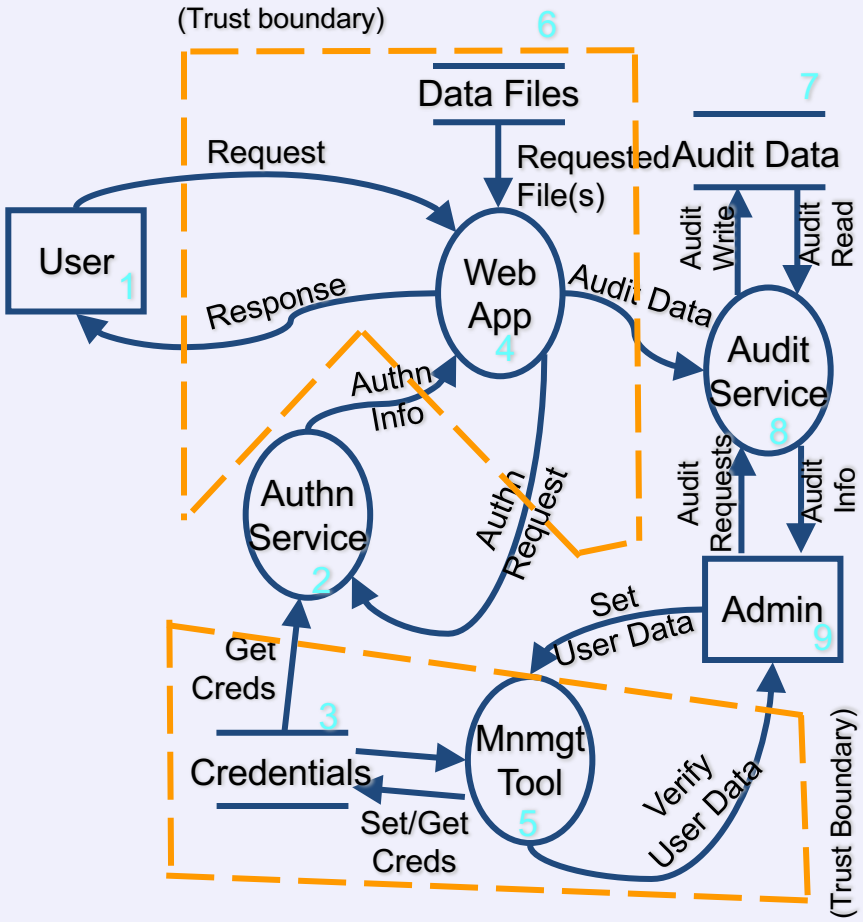
OWASP
The Open Web Application Security Project

Understand logical and component architecture of system

Understand every communication flow and valuable data moved and stored

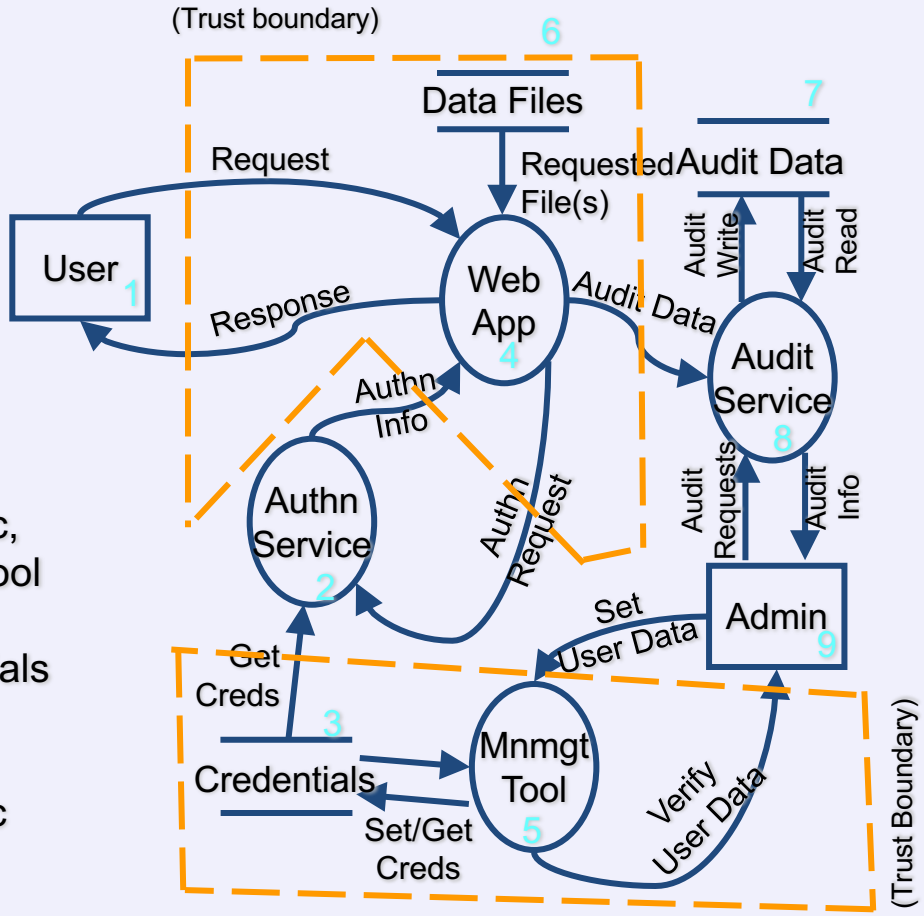


Understand the system



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
Admin <-> Audit Svc



Your threat model now consists of …

OWASP
The Open Web Application Security Project

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

OWASP
The Open Web Application Security Project

Lab 1

Review Rare Books R Us
Draw a Data Flow Diagram (DFD)

Identify threats

OWASP
The Open Web Application Security Project

Most important part of threat modeling (and most difficult)

Many ways – determine what works best for your team

Identify threats

OWASP
The Open Web Application Security Project

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

OWASP
The Open Web Application Security Project

No one would ever do that!

Why / who would ever do that?

Identity threats - Games

OWASP
The Open Web Application Security Project

OWASP Cornucopia

Suits:

Data validation and encoding

Authentication

Session Management

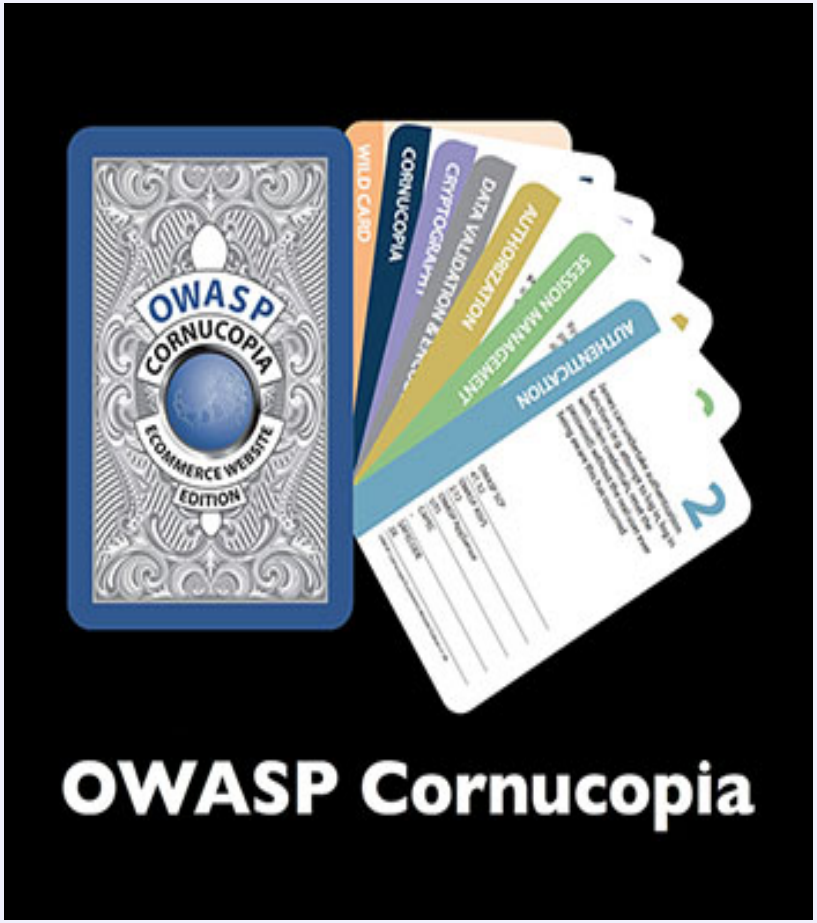
Authorization

Cryptography

Cornucopia

13 cards per suit, 2 Jokers

Play a round, highest value wins



STRIDE Framework - Threat List

OWASP
The Open Web Application Security Project

Threat	Examples
Spoofing	Pretending to be someone else
Tampering	Modifying data that should not be modifiable
Repudiation	Claiming someone didn't do something
Information disclosure	Exposing information
Denial of service	Preventing a system from providing service
Elevation of privilege	Doing things that one isn't suppose to do

STRIDE Framework – Data Flow

OWASP
The Open Web Application Security Project

Threat	Property we want
Spoofing	Authentication
Tampering	Integrity
Repudiation	Non-repudiation
Information Disclosure	Confidentiality
Denial of Service	Availability
Elevation of Privilege	Authorization

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

OWASP
The Open Web Application Security Project

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

Identify Threats – Functional

OWASP
The Open Web Application Security Project

Input and data validation

Authentication

Authorization

Configuration management

Sensitive data / privacy concerns

Identify Threats – Functional

OWASP
The Open Web Application Security Project

Session management

Cryptography

Parameter manipulation

Exception management

Auditing and logging

Identity Threats - Ask Questions

OWASP
The Open Web Application Security Project

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

OWASP
The Open Web Application Security Project

How is authentication handled between callers and services?

What about authorization?

Are we sending data in the open?

Are we using cryptography properly?

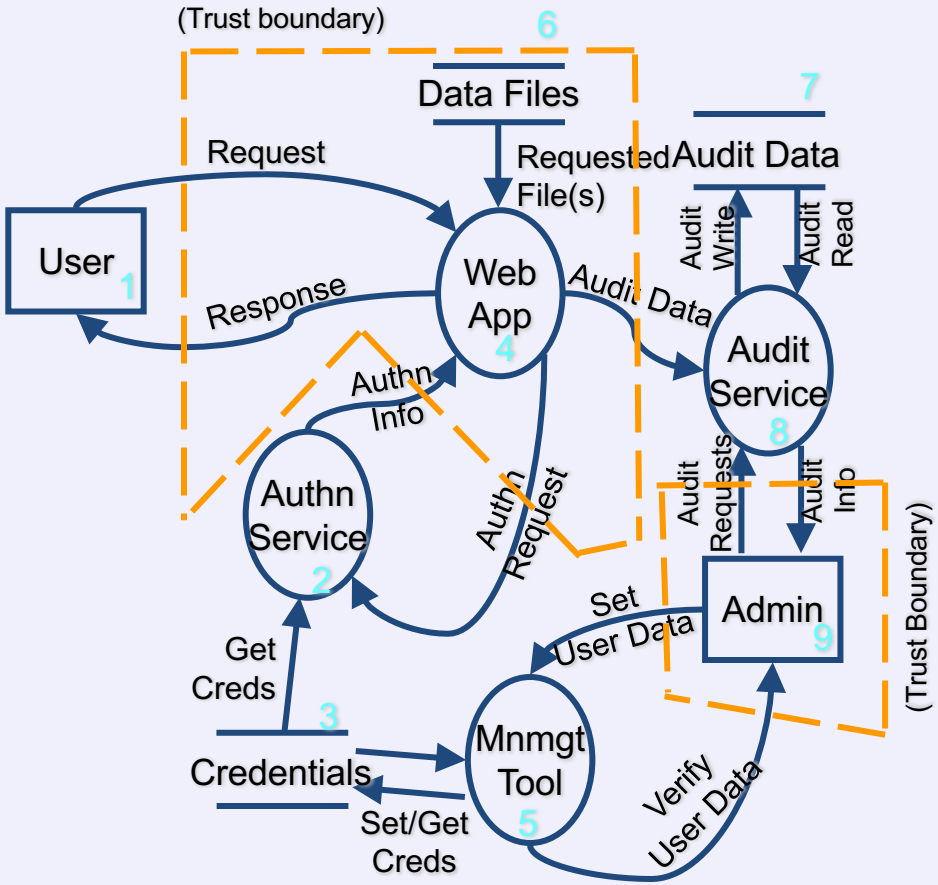
Is there logging? What is stored?

Etc.

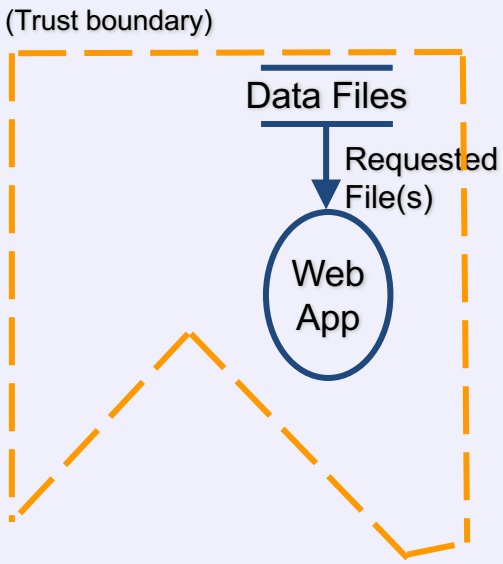
Is there anything keeping you up at night worrying about this system?

Scenario – Configuration Management

OWASP
The Open Web Application Security Project



Scenario – Configuration Management



Data Files such as configuration files

Scenario – Configuration Management

OWASP
The Open Web Application Security Project

System: Web application uses configuration files

Security principles:

- Be reluctant to trust, Assume secrets not safe

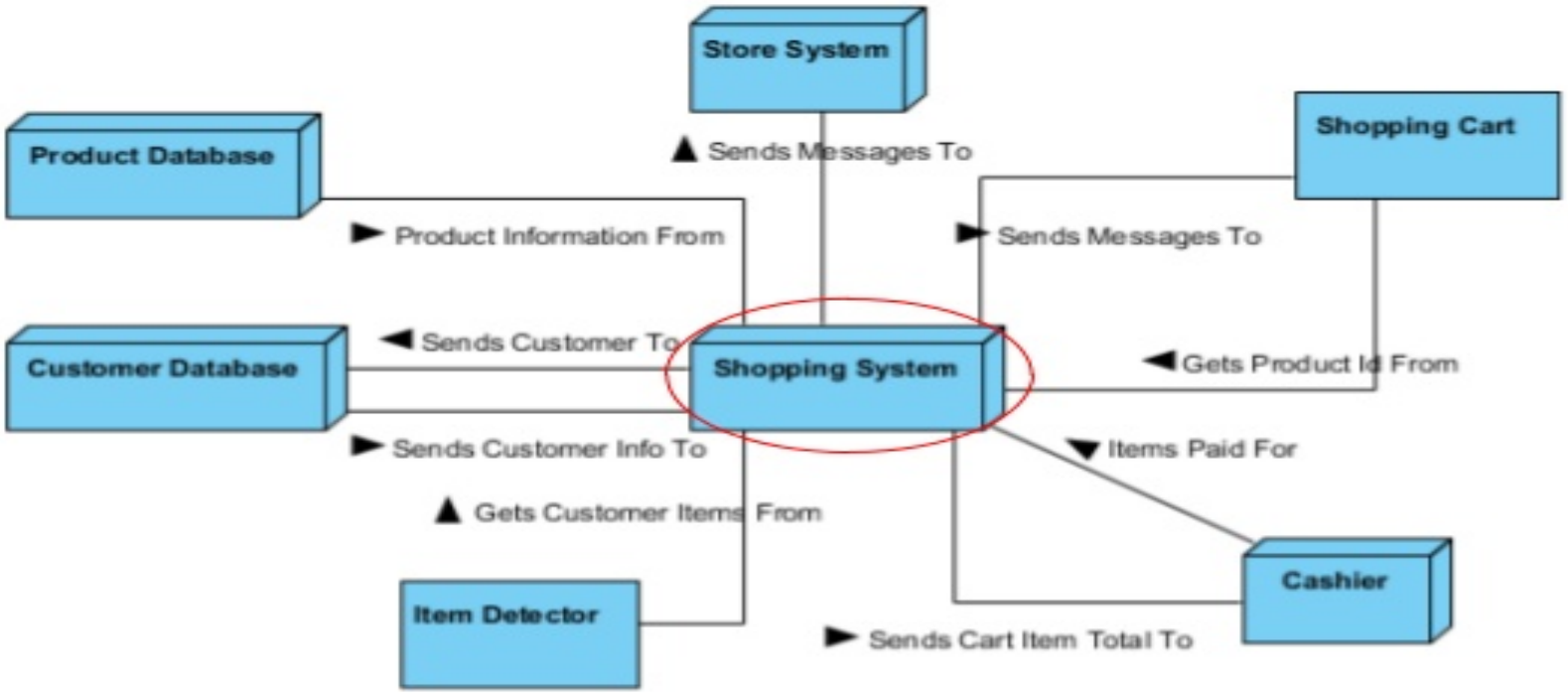
Questions:

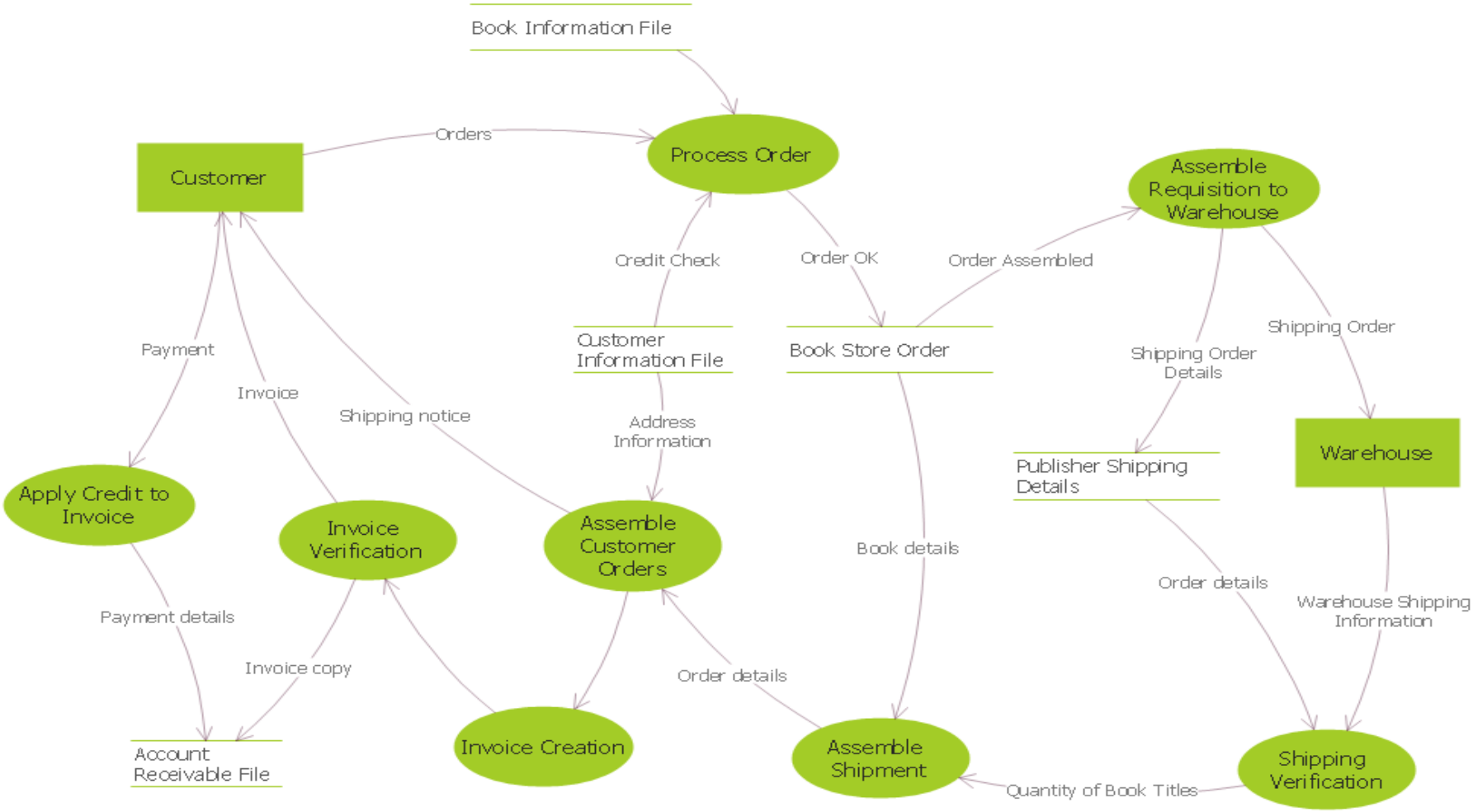
- How does the app use the configuration files?
- What validation is applied? Implied trust?

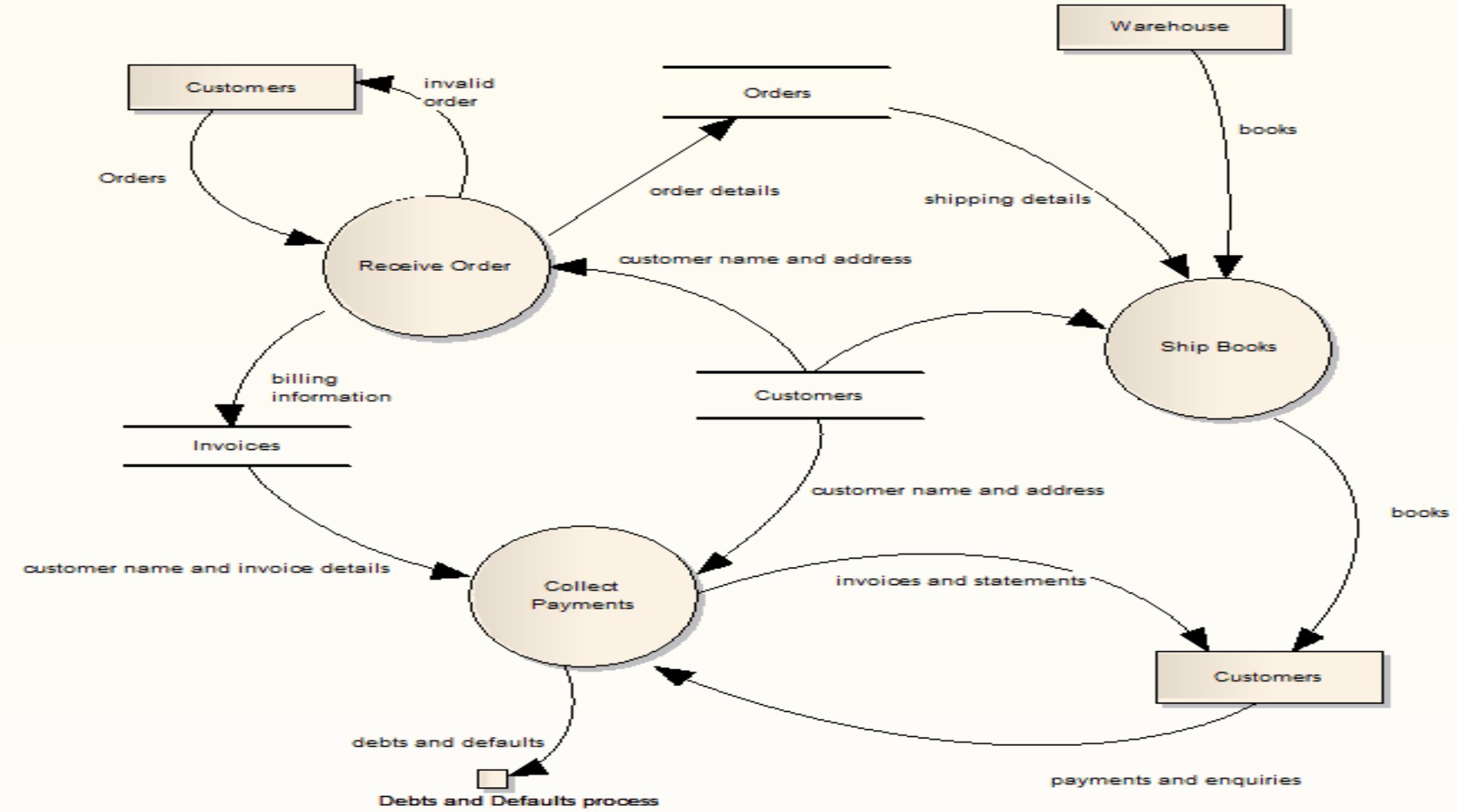
Possible controls/mitigation:

- Set permissions on configuration files.
- Validate all data input from files. Use fuzz testing to insure input validation.

Architecture Diagram




Book Information File
Orders
Customer
Process Order
Assemble Requisition to Warehouse
Credit Check
Order OK
Order Assembled
Customer Information File
Book Store Order
Shipping Order
Shipping Order Details
Payment
Invoice
Shipping notice
Address Information
Warehouse
Publisher Shipping Details
Apply Credit to Invoice
Invoice Verification
Assemble Customer Orders
Book details
Order details
Warehouse Shipping Information
Payment details
Invoice copy
Order details
Account Receivable File
Invoice Creation
Assemble Shipment
Shipping Verification
Quantity of Book Titles


Warehouse
Customers
invalid order
Orders
books
Orders
order details
shipping details
Receive Order
customer name and address
Ship Books
billing information
Customers
Invoices
customer name and address
books
customer name and invoice details
Collect Payments
invoices and statements
Customers
debts and defaults
payments and enquiries
Debts and Defaults process

Your threat model now consists of …

OWASP
The Open Web Application Security Project

1. Diagram / understanding of your system and the data flows
2. Identify threats through answers to questions

OWASP
The Open Web Application Security Project

Lab 2

Review Rare Books R Us DFD
Create threats

Authentication:

All connections go through an appropriate and adequate form of authentication

No bypassing

Credentials never traverse wire in clear text

OWASP
The Open Web Application Security Project

Authorization:

Authorization mechanisms in place

Authorization after Authentication

Least privilege stance in operation

Checked on every request

OWASP
The Open Web Application Security Project

Data/Input Validation:

All input properly validated

Proper length checks on all input

Well formed data

Golden Rule: All external input, no matter what it is, is examined and validated

OWASP
The Open Web Application Security Project

Error Handling / Information leakage:

Ensure all method/function calls that return a value have proper error handling and return value checking

Exceptions and error conditions are properly handled

Application fails in a secure manner

OWASP
The Open Web Application Security Project

Logging / Auditing:

No sensitive information is logged

Successful and unsuccessful authentication logged

Application errors logged

Cryptography:

No sensitive data transmitted in the clear

Application implements known good cryptographic methods

OWASP
The Open Web Application Security Project

Session Management:

Distinguish unauthenticated and authenticated sessions

Complex / strong Session ID

Session storage and tracking

Invalid Session ID handling

Multithreaded / multi-user sessions

Determine mitigations and risks

OWASP
The Open Web Application Security Project

Mitigation Options:

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

What is the risk associated with the threat?

OWASP
The Open Web Application Security Project

Risk Management

- FAIR (Factor Analysis of Information Risk) – Jack Jones, Jack Freund
- CVSS (Common Vulnerability Scoring System)
- Generic Risk Rating (High, Medium, Low)

Risk Rating

OWASP
The Open Web Application Security Project

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

OWASP
The Open Web Application Security Project

Risk Rating	Description
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

OWASP
The Open Web Application Security Project

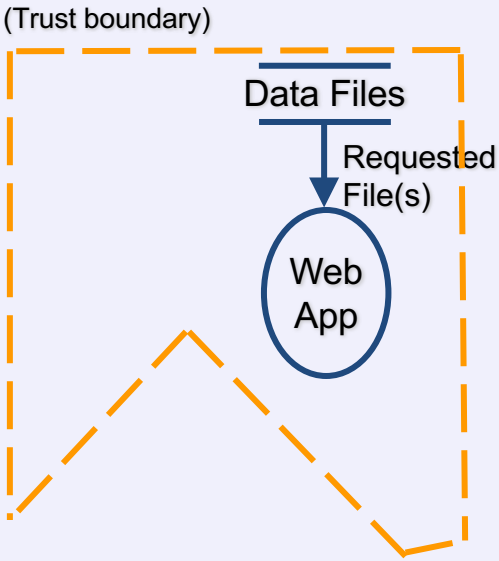
Risk Rating	Description
High	• Administrator-level access (for arbitrary code execution through privilege escalation for instance) or disclosure of sensitive information • Depending on the criticality of the system, some denial-of-service issues are considered high impact • All or significant number 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

OWASP
The Open Web Application Security Project

ID - Risk	3 - Medium
Threat	Lack of CSRF protection allows attackers to submit commands on behalf of users
Description/Impact	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
Countermeasures	Per transaction codes (nonce), thresholds, event visibility
Components Affected	CO-3

Scenario – Configuration Management



Data Files such as configuration files

Scenario – Configuration Management

OWASP
The Open Web Application Security Project

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 …

OWASP
The Open Web Application Security Project

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

OWASP
The Open Web Application Security Project

Lab 3

Review Rare Books R Us DFD

Determine mitigations and risks

Follow through

OWASP
The Open Web Application Security Project

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 …

OWASP
The Open Web Application Security Project

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!

OWASP Threat Dragon

OWASP
The Open Web Application Security Project

OWASP Incubator Project – early 2017

Project Leader: Mike Goodwin (@theblacklabguy)

OWASP Threat Dragon

https://www.owasp.org/index.php/OWASP_Threat_Dragon

Links – Source, Threat Dragon Live, Docs

https://github.com/mike-goodwin/owasp-threat-dragon

https://threatdragon.org/

http://docs.threatdragon.org

Featured on AppSecPodcast (S02E06) 6/27/2017

https://www.appsecpodcast.org/2017/06/27/the-girl-with-the-owasp-threat-dragon-tattoo-s02e06/

OWASP Threat Dragon

OWASP
The Open Web Application Security Project

Free, open-source, threat modelling web application for teams implementing the STRIDE approach

Key Areas:

Great UX - using Threat Dragon should be simple, engaging and fun

A powerful threat/mitigation rule engine - lowers barrier to entry for teams and allows non-specialists to contribute

Integration points with other development lifecycle tools - ensures models slot easily into development lifecycle and remain relevant as the project evolves

OWASP
The Open Web Application Security Project

Uses Angular JS, Node JS

Web-based app integrates with GitHub account – saves models to GitHub

Desktop Installation (Using Electron - Windows / Mac versions) – saves models to file system

OWASP Threat Dragon

OWASP
The Open Web Application Security Project

It is still in the “alpha” stage, so still “rough”

Take if for a spin – give feedback

Needs:

- Threat Modeling practitioners use it / test it / give feedback
- Developers help / join dev team

OWASP
The Open Web Application Security Project

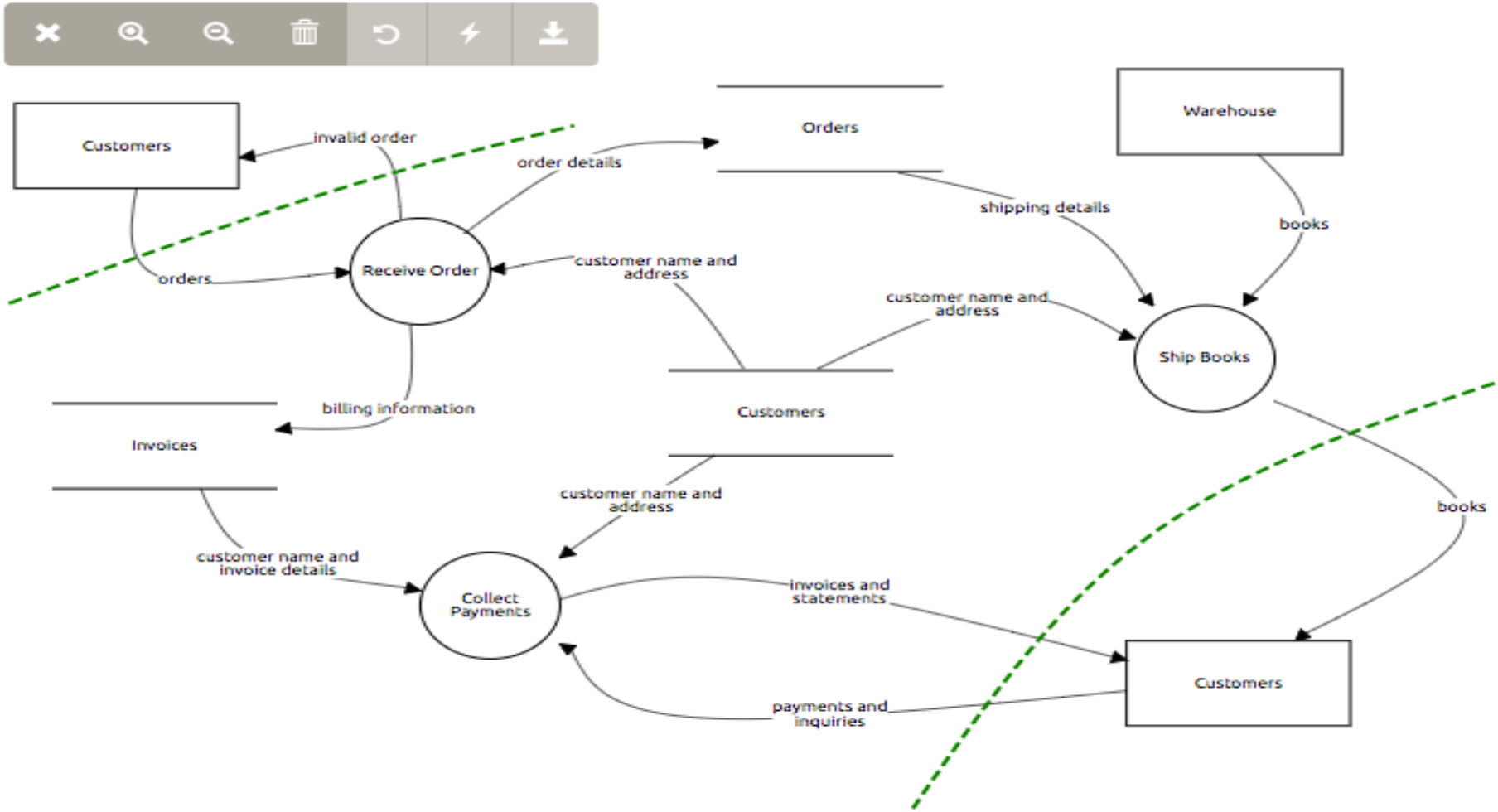
Lab 4

OWASP Threat Dragon

Threat Dragon – RBRU TM

OWASP
The Open Web Application Security Project

RBRU TM



Your challenge

OWASP
The Open Web Application Security Project

Use threat modeling for:

- secure design before new features
- driving your testing and other review activities
- understanding bigger picture

Resources - Books

OWASP
The Open Web Application Security Project

Threat Modeling: Designing for Security

Adam Shostack

Securing Systems: Applied Architecture and Threat Models

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

OWASP
The Open Web Application Security Project

Microsoft Threat Modeling Tool 2016

http://www.microsoft.com/en-us/download/details.aspx?id=49168

https://blogs.msdn.microsoft.com/secdevblog/2017/04/21/whats-new-with-microsoft-threat-modeling-tool-preview/ (2017 Preview)

ThreatModeler – Web Based (in-house) Tool

http://myappsecurity.com

ThreadFix

http://www.denimgroup.com/blog/denim_group/2016/03/threadfix-in-action-tracking-threats-and-threat-models.html

IriusRisk Software Risk Manager

https://iriusrisk.continuumsecurity.net

OWASP Threat Dragon

https://www.owasp.org/index.php/OWASP_Threat_Dragon

Resources - Tools

OWASP
The Open Web Application Security Project

Attack Trees – Bruce Schneier on Security

https://www.schneier.com/attacktrees.pdf

Elevation of Privilege (EoP) Game

http://www.microsoft.com/en-us/download/details.aspx?id=20303

OWASP Cornucopia

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

OWASP Application Security Verification Standard (ASVS)

https://www.owasp.org/index.php/Category:OWASP_Application_Security_Verification_Standard_Project

OWASP Proactive Controls 2016

https://www.owasp.org/index.php/OWASP_Proactive_Controls

Questions?

OWASP
The Open Web Application Security Project

Contacts

Web Site:
https://roberthurlbut.com

Twitter: @RobertHurlbut

ROBERT HURLBUT CONSULTING SERVICES

securedevelop.NET