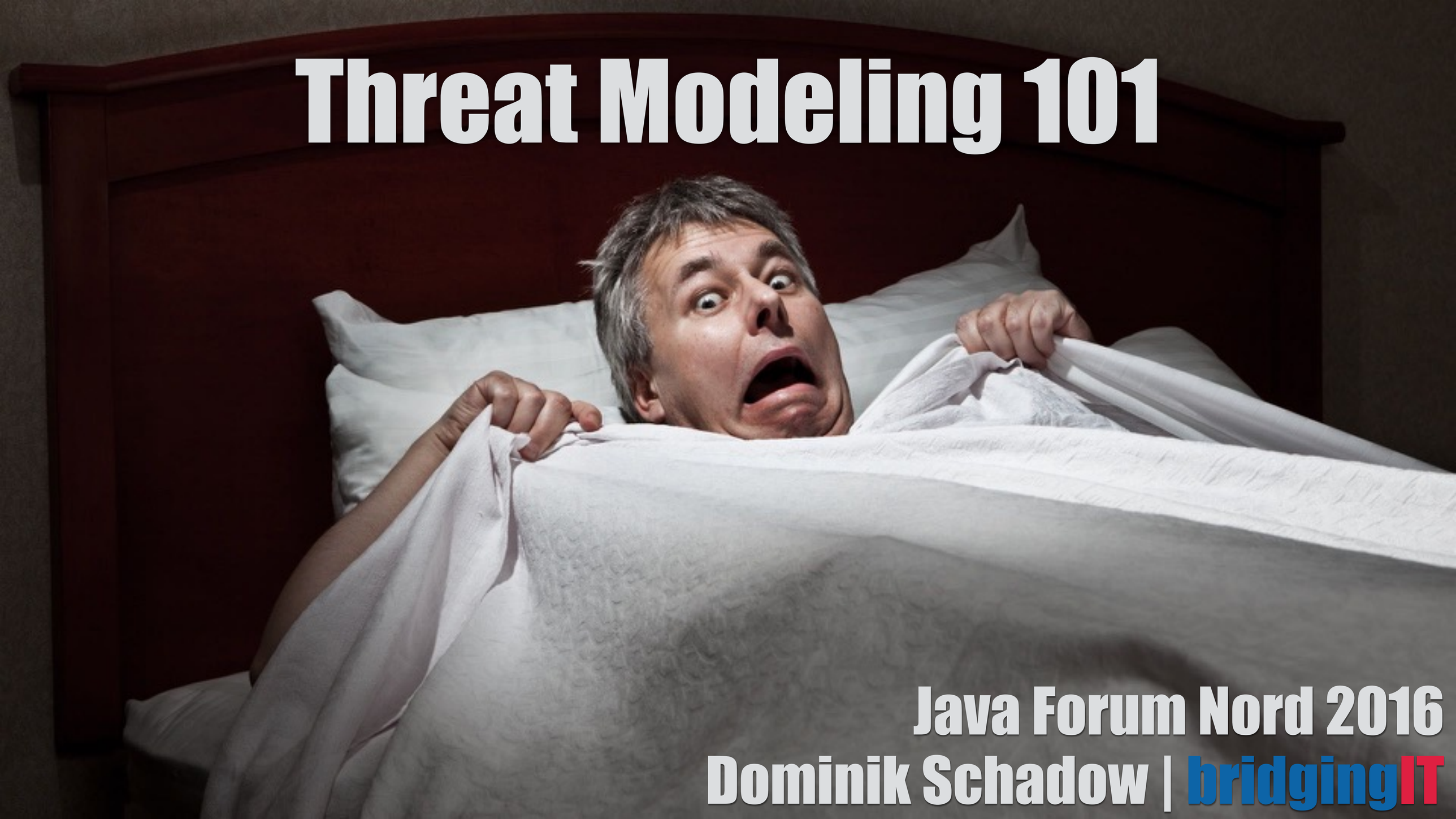
Threat Modeling 101



Java Forum Nord 2016

Dominik Schadow | bridgingIT

Threat

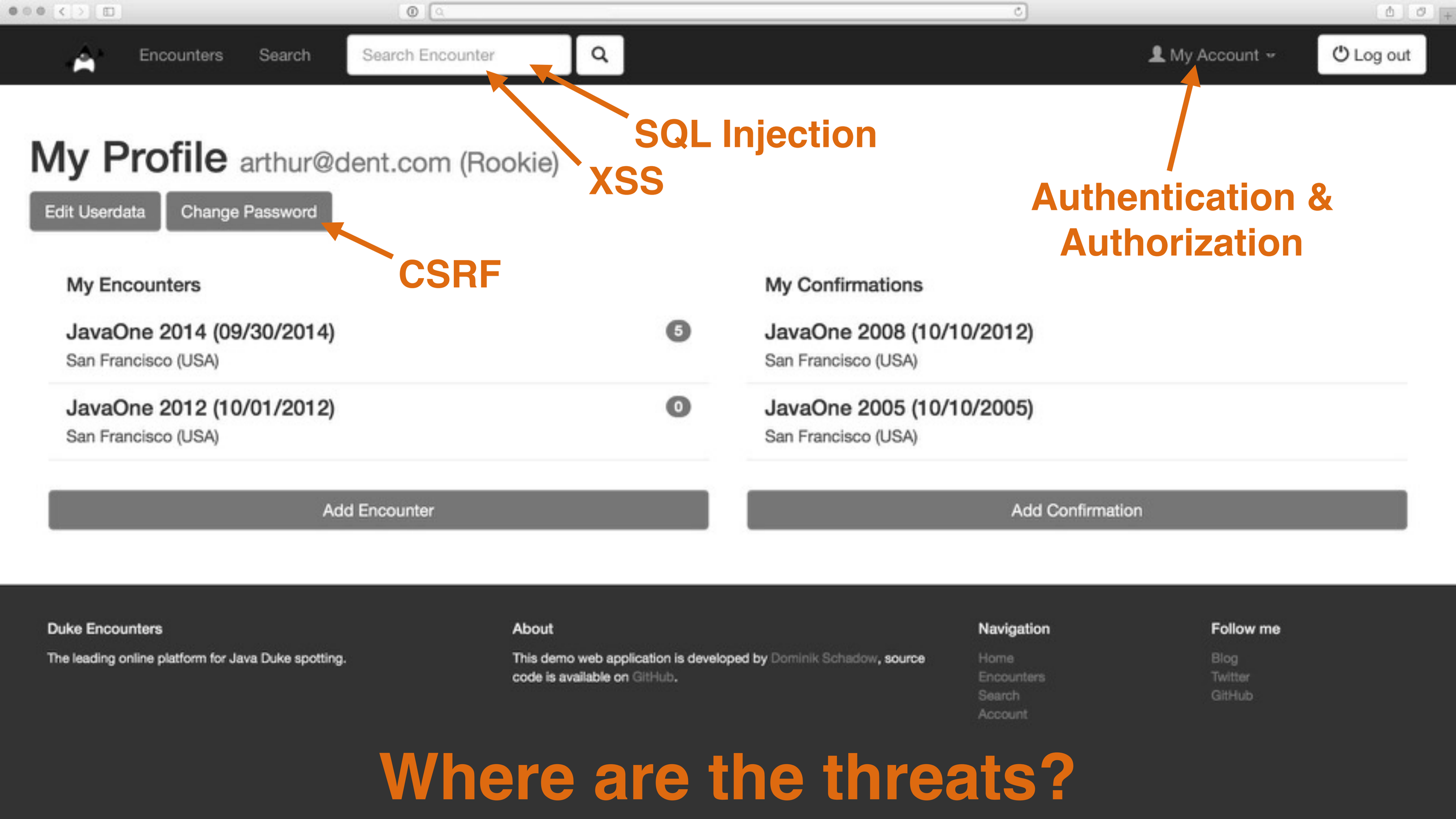
A source of damage or danger

Anything that can act against an asset (the threat target) resulting in a potential loss

Where are the threats?

Java web application

spring boot


Encounters
Search
Search Encounter
My Account
Log out
SQL Injection
XSS
Authentication &
Authorization
My Profile arthur@dent.com (Rookie)
Edit Userdata
Change Password
CSRF
My Encounters
JavaOne 2014 (09/30/2014)
5
San Francisco (USA)
JavaOne 2012 (10/01/2012)
0
San Francisco (USA)
My Confirmations
JavaOne 2008 (10/10/2012)
San Francisco (USA)
JavaOne 2005 (10/10/2005)
San Francisco (USA)
Add Encounter
Add Confirmation
Duke Encounters
The leading online platform for Java Duke spotting.
About
This demo web application is developed by Dominik Schadow, source code is available on GitHub.
Navigation
Home
Encounters
Search
Account
Follow me
Blog
Twitter
GitHub
Where are the threats?

Agenda

Threat Modeling **Basics**

Identifying Threats in Applications

Threat Modeling in **Action**

Threat Modeling Basics

Security flaws exist before code

Know and reduce attack surface with threat modeling

- Forget to authenticate a user
- Broken authorization
- Incomplete central user management system usage
- Missing auditing functionality

Different ways to threat model

There is no single perfect way

Focus on attackers: Can you really think like an attacker?
Focus on assets: What are your assets (valuables, qualities)? How do you link assets to threats?

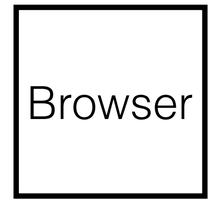
Follow the data

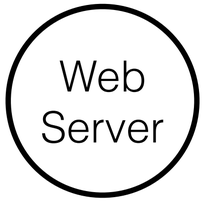
Threats tend to follow the data flow

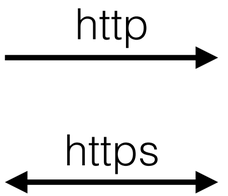
Start with external entities and follow the data flow through your application in a structured way and identify the real problems

Data Flow Diagrams

External Entity	People or code outside your control that interact with the application
Process	Code and components that handle data and the dev team controls
Data Store	Anything that stores data and does not modify it
Data Flow	Represents data movement within the application (including direction)



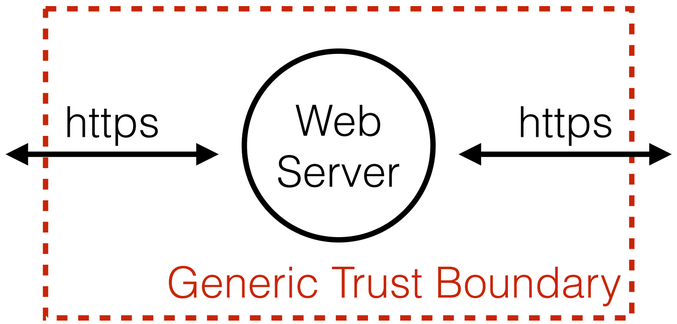


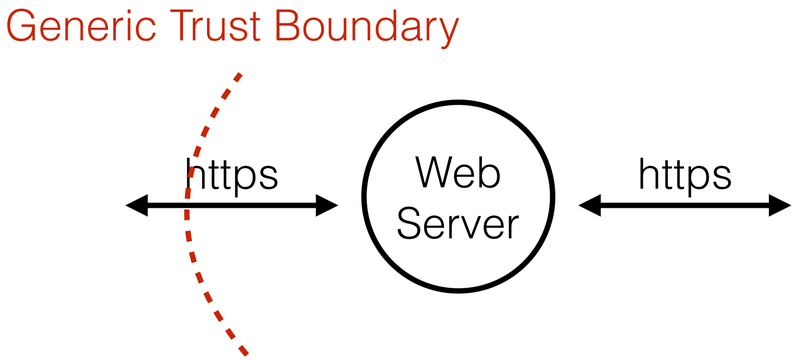


Trust Boundaries

Trust Boundary

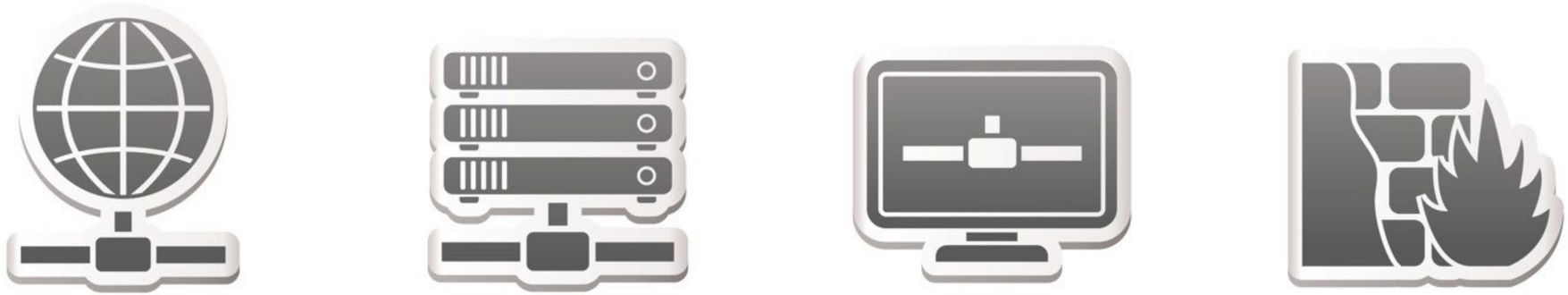
Represents the change of privilege levels as the data flows through the application (change in level of trust)





Typical boundaries

Can be technical or organizational



Typical boundary locations

Follow the data, add boundary for new principal

Anonymous user

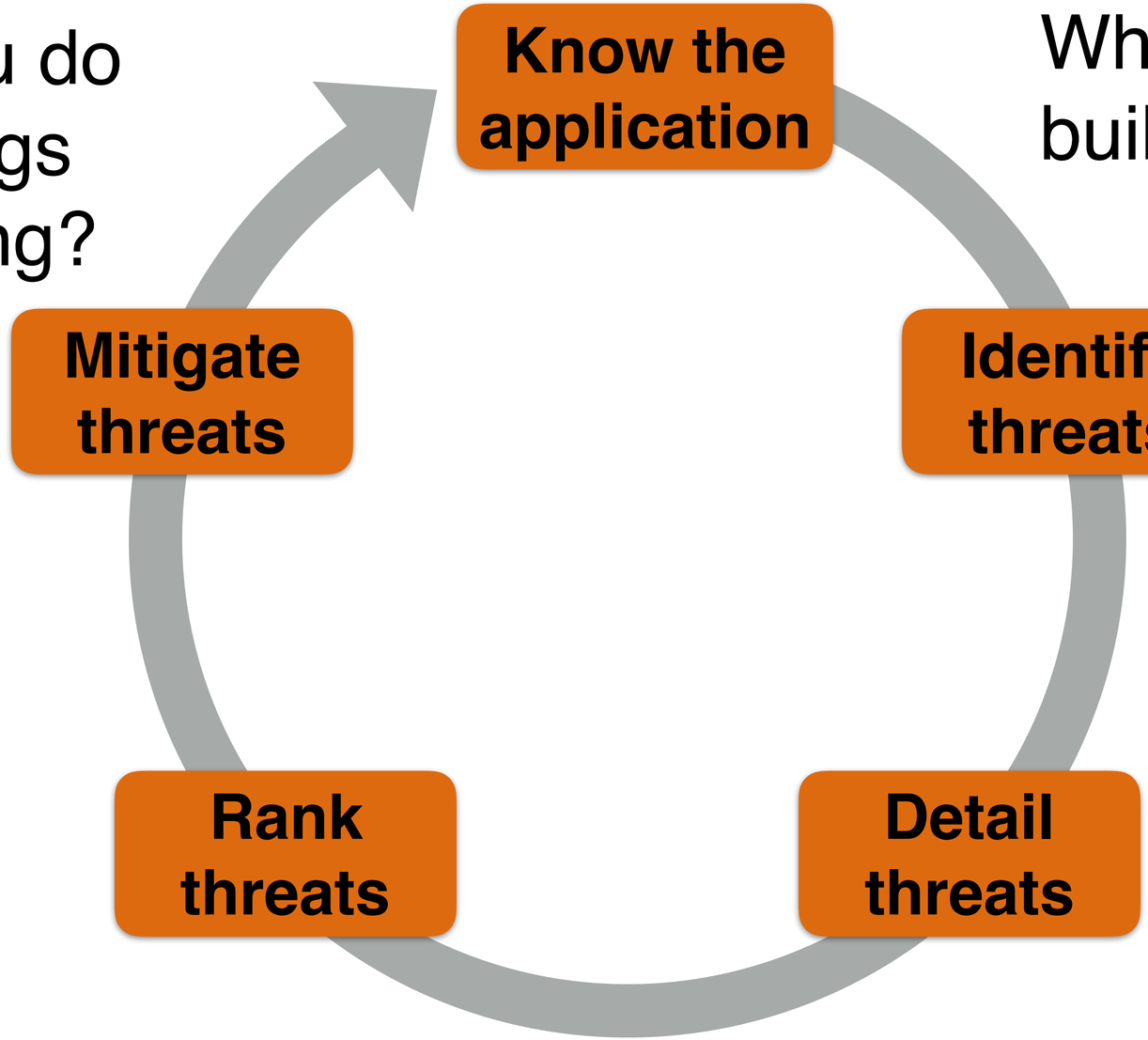
Tomcat user

MySQL user

Identifying Threats in Applications

Identifying threats in applications

What should you do about those things that can go wrong?



What are you building?

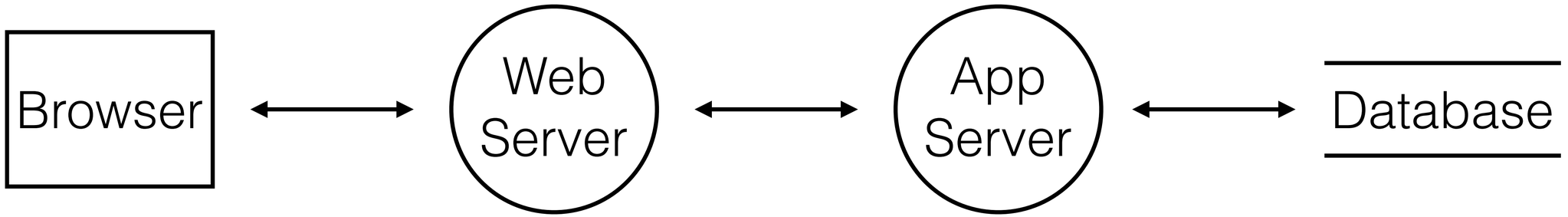
What can go wrong?

What are you building?

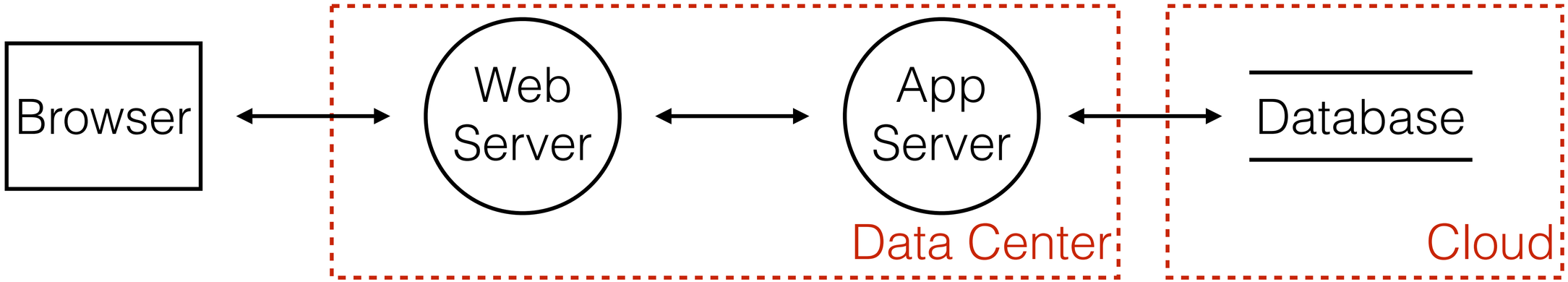
Focus on data flow

„*Sometimes*“ indicates alternatives: model all
No data sinks: show the consumers
Data does not move by itself: draw the process moving it

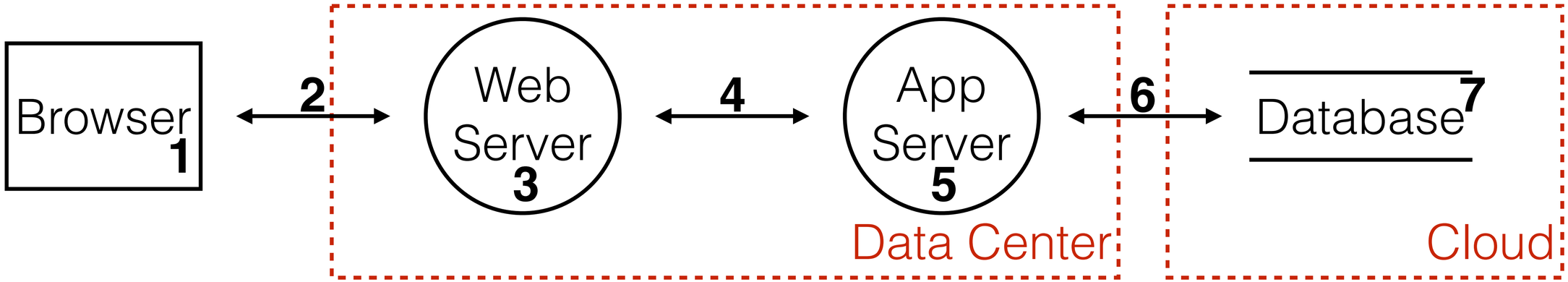
Follow the data



Add trust boundaries



Identify each element



What can go wrong?

Start with data crossing trust boundaries

Brainstorm meetings with technology experts
Elevation of Privilege game
STRIDE

STRIDE

STRIDE is the opposite of a property you want

Spoofing, **T**ampering, **R**epudiation, **I**nformation Disclosure, **D**enial of Service, **E**levation of Privilege

STRIDE

Spoofing — Pretending to be something or somebody else
Violated property: Authentication
Standard defenses: Passwords, multi-factor authN

Tampering — Modifying something on disk, network or memory
Violated property: Integrity
Standard defenses: Digital signatures, hashes

Repudiation — Claiming that someone didn't do something
Violated property: Non-Repudiation
Standard defenses: Logging, auditing, timestamps

STRIDE

Information Disclosure
Providing information to someone not authorized
Violated property: Confidentiality
Standard defenses: Encryption, authorization

Denial of Service
Absorbing resources needed to provide service
Violated property: Availability
Standard defenses: Filtering, quotas

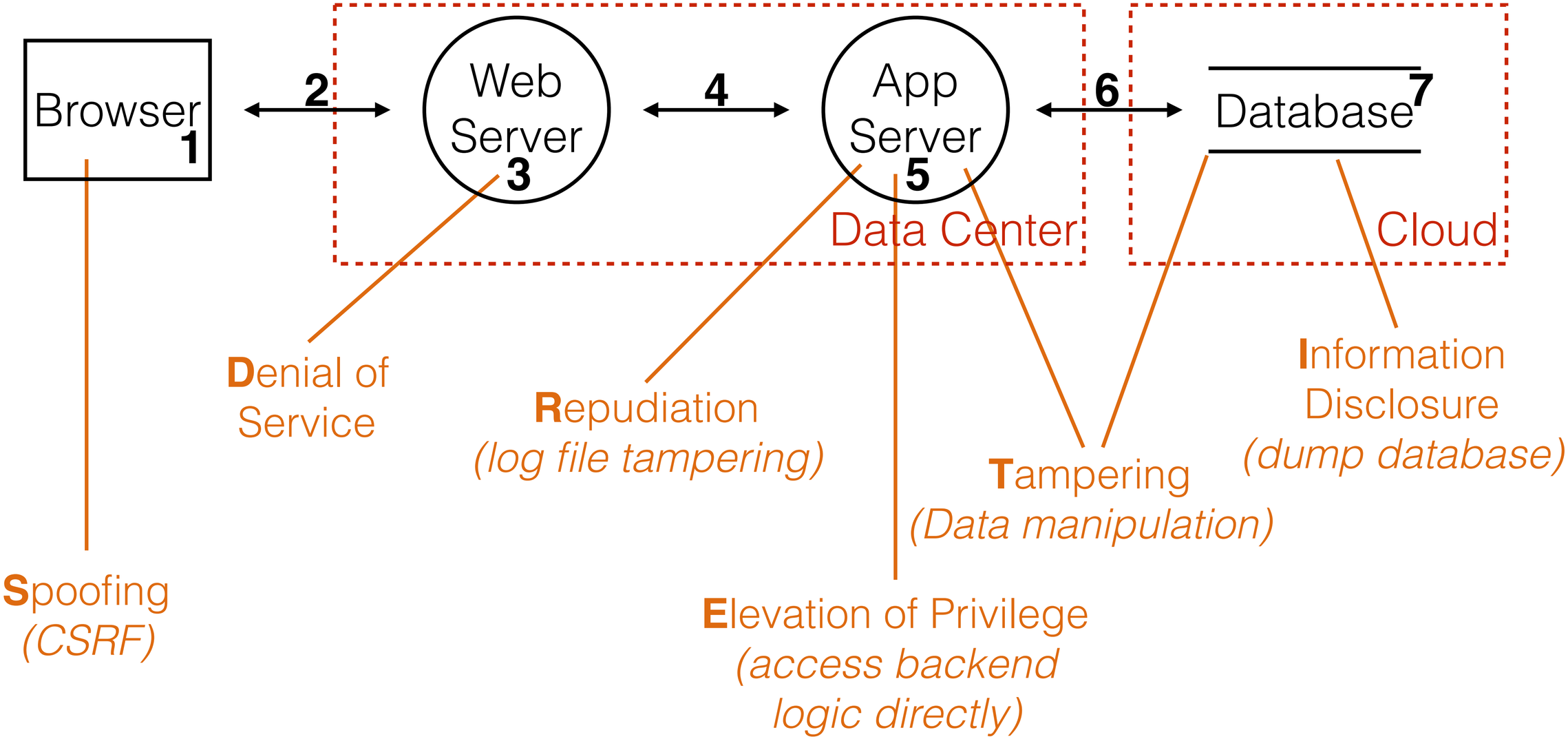
Elevation of Privilege
Doing something someone is not authorized to do
Violated property: Authorization
Standard defenses: Input validation, least privilege

www.my-web-application.com?**admin=false**

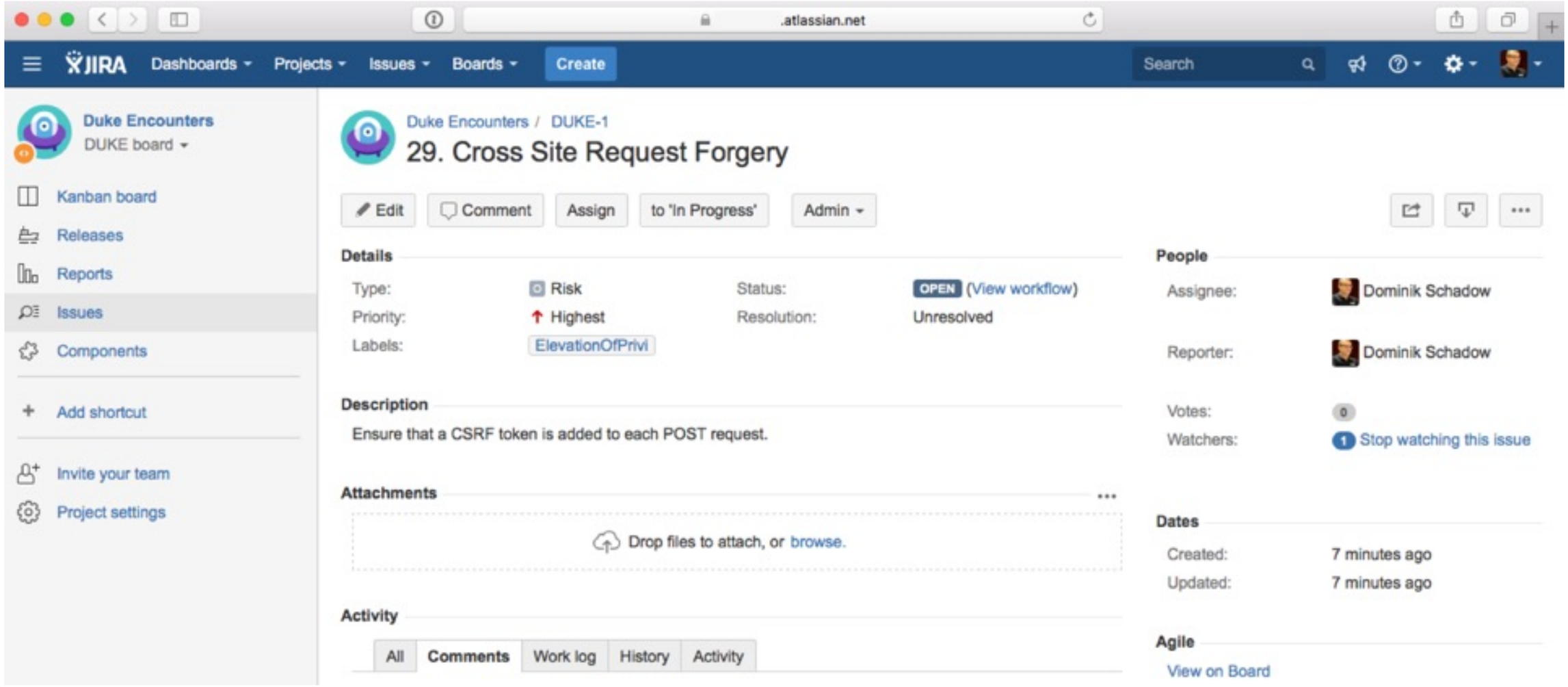
www.my-web-application.com?**admin=true**

Elevation of Privilege

Add threats


Browser
1
2
Web
Server
3
4
App
Server
5
6
Database
7
Data Center
Cloud
Spoofing
(CSRF)
Denial of
Service
Repudiation
(log file tampering)
Elevation of Privilege
(access backend
logic directly)
Tampering
(Data manipulation)
Information
Disclosure
(dump database)

Add all risks to bug tracking



Addressing each threat

Decide for each threat how to handle it

Mitigate **Eliminate** **Transfer** **Accept**

Mitigate it

Preferred solution

Do something to make it harder to take advantage of a threat (like adding Spring Security AND configuring it)

Eliminate it

Most secure solution

Results in feature elimination most of the time (like removing admin functionality from the Internet facing application)

Transfer it

Team solution

Someone/ something else handles the risk, depending who can easily fix the problem (like operations adding a web application firewall)

Accept it

Last resort solution

Stop worrying about it and live with the risk (like someone stealing your servers' hard disk)

Threat Target	Mitigation Strategy	Mitigation Technique	Priority	Issue ID
Repudiating actions	Log	Logging all security relevant actions in an audit log	2	1001
Spoofing a user	Identification and authentication	Password policy, token, password reset process	1	1002
Network flooding	Elastic cloud	Dynamic cloud resources (servers and databases) to provide service	3	1006
Tampering network packets	Cryptography	HTTPS/TLS	1	1007

Is it complete?

Let someone introduce the application by following the data flow

Watch out for phrases like *„Sometimes we have to do … instead of … here“* or *„A lot of things are happening here which are not completely listed…“*

Breadth before depth

Criteria exist to show you are NOT done, but none to show you are done

Easy

One threat of each STRIDE type

Harder

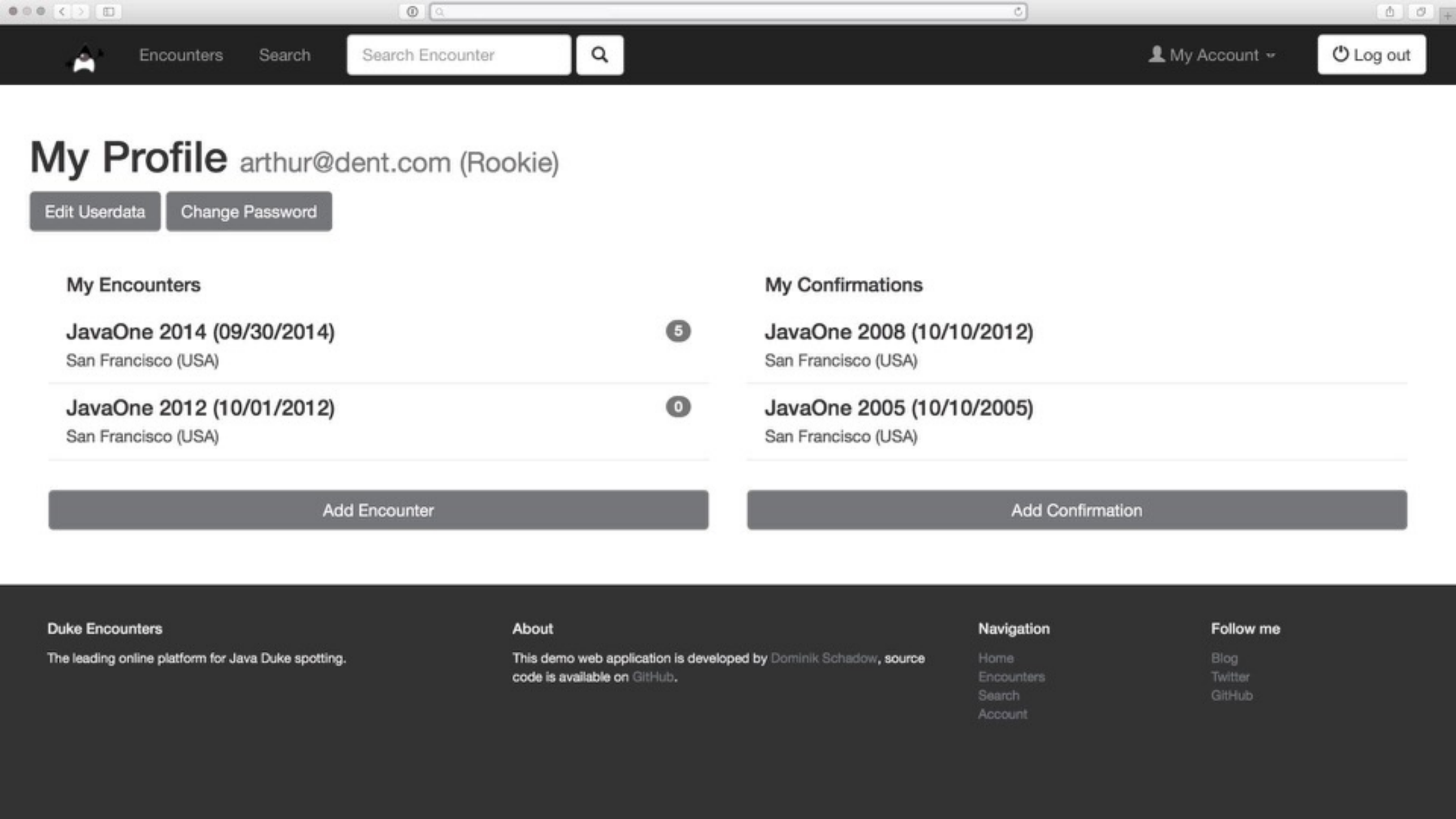
One threat per diagram element

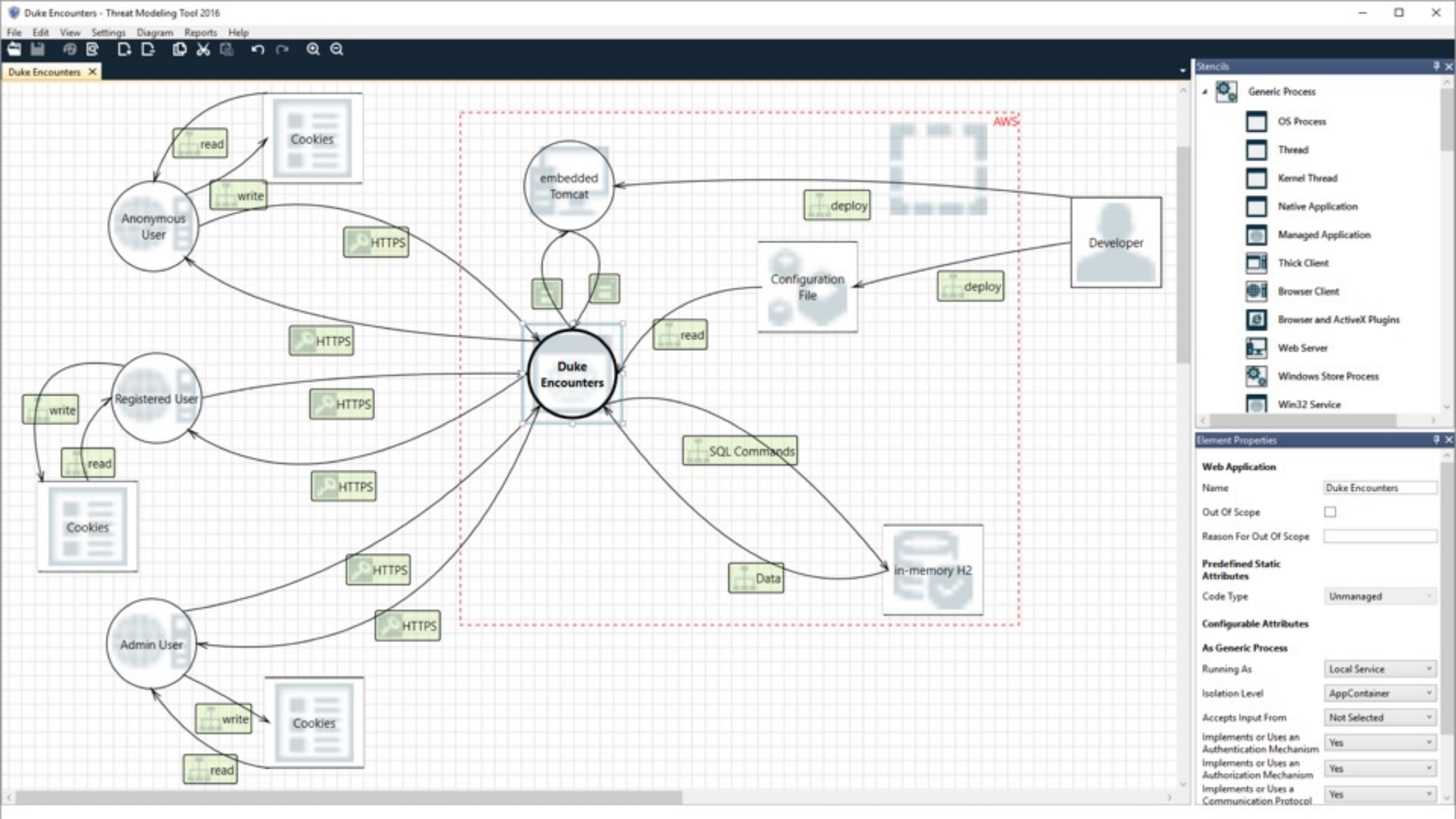
Threat Modeling in **Action**

A threat model is a living document

Version models in the repo

Check and update them every time the application changes and regularly from time to time


Encounters
Search
Search Encounter
My Account
Log out
My Profile arthur@dent.com (Rookie)
Edit Userdata
Change Password
My Encounters
JavaOne 2014 (09/30/2014)
5
San Francisco (USA)
JavaOne 2012 (10/01/2012)
0
San Francisco (USA)
My Confirmations
JavaOne 2008 (10/10/2012)
San Francisco (USA)
JavaOne 2005 (10/10/2005)
San Francisco (USA)
Add Encounter
Add Confirmation
Duke Encounters
The leading online platform for Java Duke spotting.
About
This demo web application is developed by Dominik Schadow, source code is available on GitHub.
Navigation
Home
Encounters
Search
Account
Follow me
Blog
Twitter
GitHub

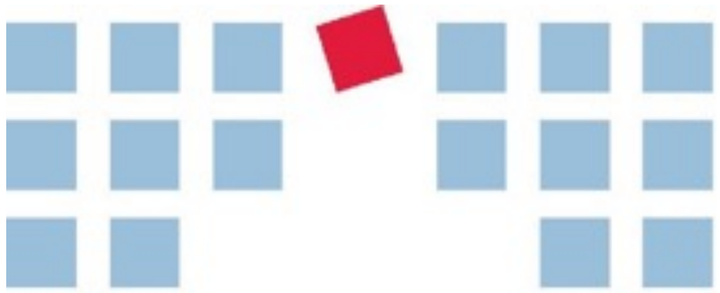

Duke Encounters - Threat Modeling Tool 2016
File Edit View Settings Diagram Reports Help
Duke Encounters
Stencils
Generic Process
OS Process
Thread
Kernel Thread
Native Application
Managed Application
Thick Client
Browser Client
Browser and ActiveX Plugins
Web Server
Windows Store Process
Win32 Service
Element Properties
Web Application
Name
Duke Encounters
Out Of Scope
Reason For Out Of Scope
Predefined Static Attributes
Code Type
Unmanaged
Configurable Attributes
As Generic Process
Running As
Local Service
Isolation Level
AppContainer
Accepts Input From
Not Selected
Implements or Uses an Authentication Mechanism
Yes
Implements or Uses an Authorization Mechanism
Yes
Implements or Uses a Communication Protocol
Yes
AWS
Duke Encounters
embedded Tomcat
Configuration File
Developer
deploy
deploy
read
Anonymous User
read
write
HTTPS
HTTPS
Registered User
write
read
HTTPS
HTTPS
Admin User
HTTPS
HTTPS
write
read
Cookies
Cookies
SQL Commands
Data
in-memory H2

Demo

Summary

Threat model early, threat model often

Address and document every identified threat



bridging IT

Marienstraße 17
70178 Stuttgart

dominik.schadow@bridging-it.de
www.bridging-it.de

Blog blog.dominikschadow.de
Twitter @dschadow

Application Threat Modeling
www.owasp.org/index.php/Application_Threat_Modeling

Microsoft Threat Modeling Tool
www.microsoft.com/en-us/sdl/adopt/threatmodeling.aspx

SecDevOps Risk Workflow
leanpub.com/secdevops

Threat Modeling: Designing for Security (Adam Shostack)
eu.wiley.com/WileyCDA/WileyTitle/productCd-1118809998.html

Pictures
www.dreamstime.com