Leveraging OWASP Projects and Tools in Your AppSec Program

John DiLeo (@gr4ybeard)

OWASP New Zealand and Datacom

February 2021

Thank You to Our Sponsors and Hosts!

OWASP
NEW ZEALAND
owasp.org.nz

THE UNIVERSITY OF AUCKLAND
Te Whare Wānanga o Tāmaki Makaurau
NEW ZEALAND

DATACOM

QUANTUM SECURITY

ATTACK™
DEFEND Group Company

aura INFORMATION SECURITY | POWERED BY KORDIA

INSOMNIA
SECURITY SPECIALISTS :: REST SECURED

myob

Pulse Security
www.pulsesecurity.co.nz

SECURE CODE WARRIOR

VOCUS

ZX SECURITY

flux

internetnz

LATERAL SECURITY

secInit security initiative

SEQA™
Information Security

Without them, this Conference couldn't happen

About Me

- Past lives
  - Simulation developer and system analyst
  - University lecturer - Maths, Comp Sci, IT, *et al.*
  - J2EE developer and architect
- Moved to Application Security in 2014
- Moved to New Zealand in late 2017
- OWASP Leadership
  - New Zealand Chapter
  - Author, Software Assurance Maturity Model (SAMM)
  - AppSec Curriculum Project

OWASP
Open Web Application Security Project

Where I Work and What I Do

Datacom's AppSec Division

- Team Lead in new consulting group
- External: Advise on Software Assurance
  - SAMM-based maturity assessments
  - Maturity improvement guidance
  - GRC, Training, Tooling, DevSecOps
- Internal: Improve Software Assurance maturity of our development teams – "eat our own dog food"

We're hiring…and we're here, so…

What You Can Expect to Hear

- My thoughts about Software Assurance
- Some information about the OWASP Software Assurance Maturity Model (SAMM)
- The names of *lots* of OWASP Projects
- A few thoughts on leveraging OWASP Projects

What You Shouldn’t Expect to Hear

- An in-depth treatment of SAMM
- Information about *every* OWASP project
  - There are 192 “active” OWASP projects*
  - I’ll mention only 30 or so by name
  - I’ll provide *some* details on fewer than 20

* Comprised of 19 Flagship, 20 Lab, 54 Incubator, and 99 “need website update” (12 Feb 2021)

OWASP
Open Web Application
Security Project

Reasons to Love OWASP Projects

- Developed and maintained by passionate volunteers…who happen to be experts
- Supportive community of users and contributors
  - OWASP Slack (https://owasp-slack.herokuapp.com/)
  - Project channels (e.g., #project-samm)
  - Topical channels (e.g., #threat-modeling)
- Open-source – Public repos on GitHub
- Project deliverables are FREE
  (as in ‘freedom’ *and* as in ‘free beer’)

OWASP
Open Web Application
Security Project

What Is Software Assurance?

Software Assurance is the "[l]evel of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at anytime during its lifecycle and that the software functions in the intended manner."

- [US] National Information Assurance (IA) Glossary, April 2010

And, by that you mean…?

- Attain and maintain high **stakeholder confidence** in successful delivery of the features you **intended** to deliver
- Prevent, detect, and remove **vulnerabilities**
- Ensure **reliability** and **resilience** of the production system

SO MUCH MORE than a few code reviews or 11th-hour penetration tests

Software Assurance Maturity Model Flagship Project

What is SAMM?

The Software Assurance Maturity Model (SAMM) is an open framework that provides an effective and measurable way for all types of organizations to analyze and improve their software security posture.

owaspsamm.org

Measurable
Defined maturity levels across business practices

Actionable
Clear pathways for improving maturity levels

Versatile
Technology, process, and organization agnostic

OWASP
Open Web Application Security Project

What is SAMM?

The resources provided by SAMM aid in:

- evaluating an organization’s existing software security practices;
- building a balanced software security assurance program in well-defined iterations;
- demonstrating concrete improvements to a security assurance program; and
- defining and measuring security-related activities throughout an organization.

SAMM v2.0 Structure

- Five Business Functions
- 15 Practice Areas
- 2 Activity Streams per Practice Area

Governance	Design	Implementation	Verification	Operations
Strategy & Metrics	Threat Assessment	Secure Build	Architecture Assessment	Incident Management
Policy & Compliance	Security Requirements	Secure Deployment	Requirements-driven Testing	Environment Management
Education & Guidance	Secure Architecture	Defect Management	Security Testing	Operational Management

OWASP
Open Web Application
Security Project

SAMM Maturity Levels

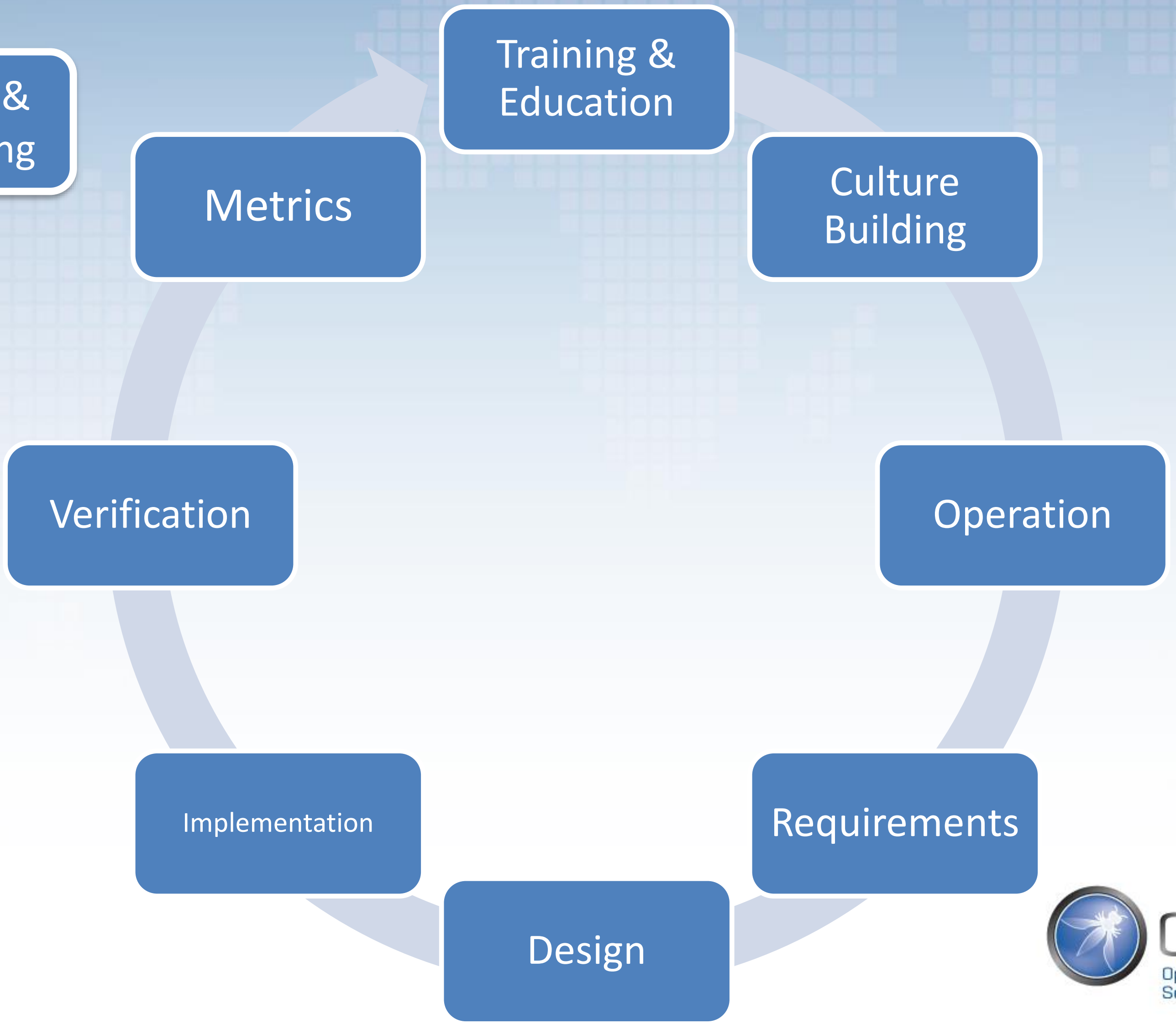
Within an Activity Stream, Activities represent progressive maturity levels:

- Level 0 – Practice unfulfilled
- Level 1 – *Ad hoc* / best-effort / inconsistent
- Level 2 – Defined / documented / standardised
- Level 3 – Measured and optimised

AppSec Program Elements

Ref: OWASP Integration Standards Project

Assessment & Road-mapping



Training & Education

Awareness Docs:

- **OWASP Top 10**
- Mobile Top 10
- API Top 10

Board Game:

- Snakes & Ladders

Training Platform:

- **SecureFlag**

Intentionally Vulnerable WebApps:

- **Juice Shop**
- Security Shepherd
- WebGoat
- PyGoat

OWASP
Open Web Application Security Project

OWASP Top 10 Flagship Project

- Standard awareness document for developers and web application security.
- Represents broad consensus about the most critical security risks to web apps
  - Current version: 2017
  - Next version: 2021

OWASP

OWASP Top 10 - 2017

The Ten Most Critical Web Application Security Risks

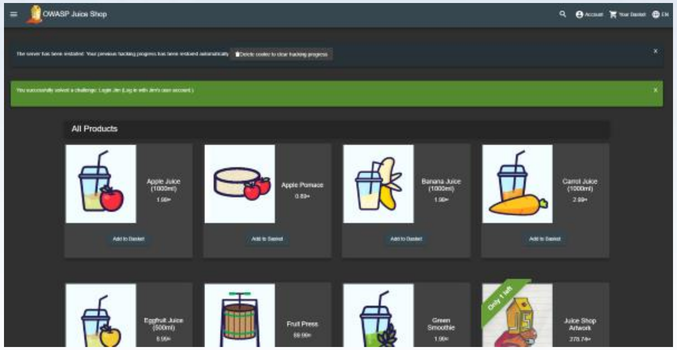
https://owasp.org

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

creative commons

Juice Shop
Flagship Project

- World’s most modern and sophisticated insecure web application!
- Exhibits vulnerabilities from the entire OWASP Top Ten, and lots more
- Useful for:
  - Security training
  - Awareness demos
  - Capture the Flag events (CTFs)
  - Target app for security tools


OWASP Juice Shop

Account Your Basket EN

The server has been restarted: Your previous hacking progress has been restored automatically. Delete cookie to clear hacking progress

You successfully solved a challenge: Login Jim (Log in with Jim's user account.)

All Products

Apple Juice (1000ml) 1.99¤ Add to Basket

Apple Pomace 0.89¤ Add to Basket

Banana Juice (1000ml) 1.99¤ Add to Basket

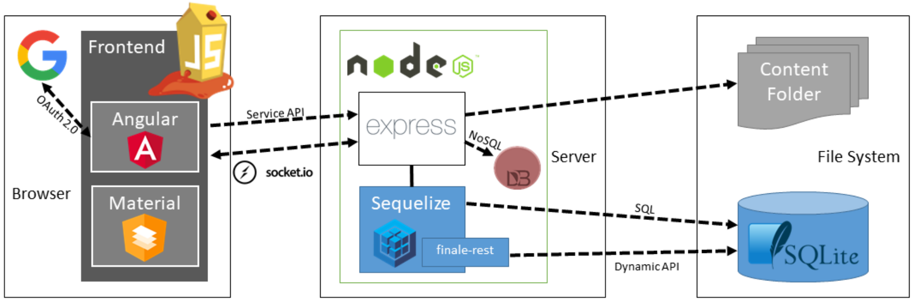
Carrot Juice (1000ml) 2.99¤ Add to Basket

Eggfruit Juice (500ml) 8.99¤

Fruit Press 89.99¤

Green Smoothie 1.99¤

Only 1 left Juice Shop Artwork 278.74¤



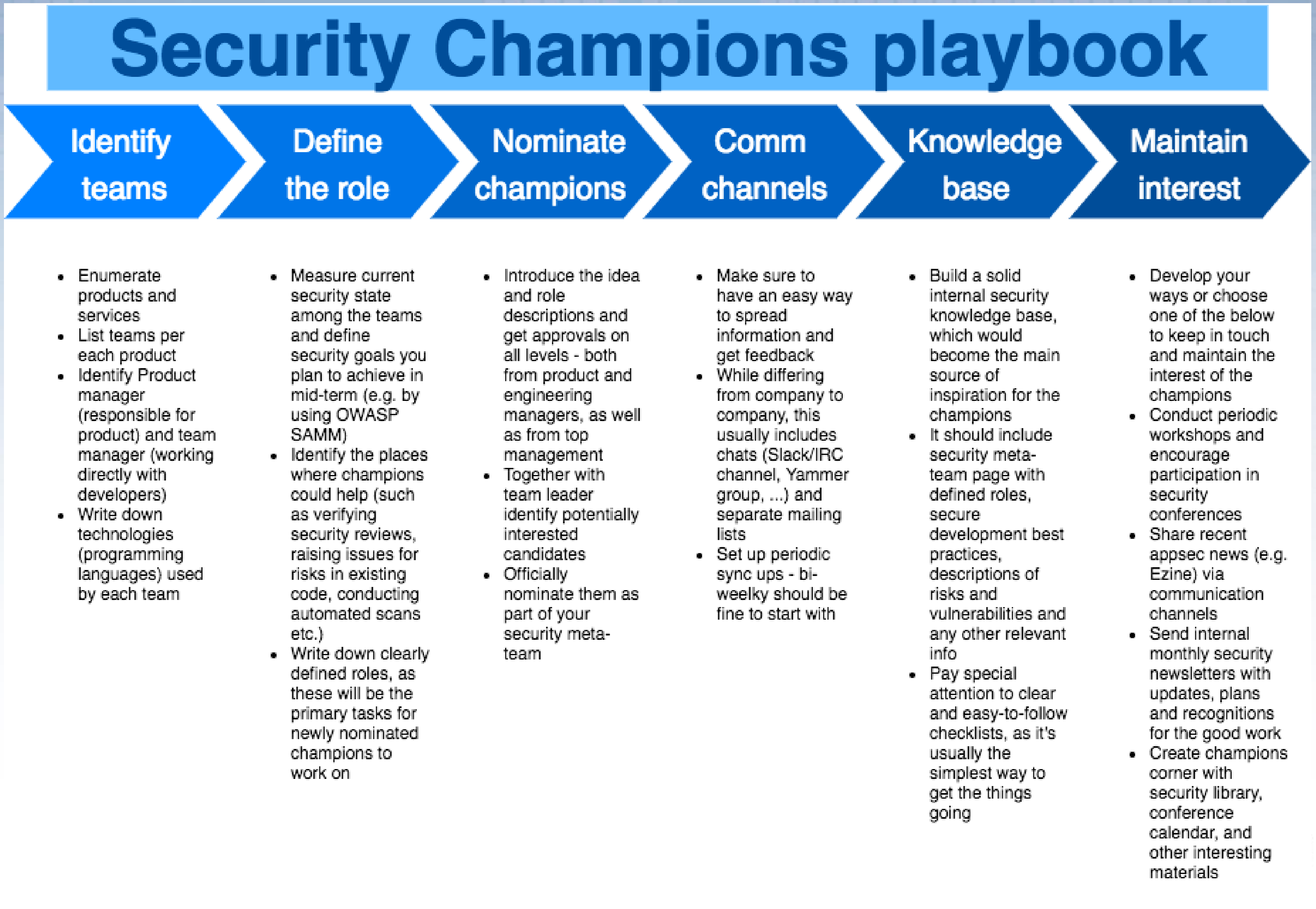
SecureFlag Open Platform Incubator Project

“The SecureFlag Open Platform is an open-source training platform created for developers to learn and practice modern secure coding techniques through hands-on exercises.”

- Built-in courses
- Real-world challenges – verified with *functional tests*
- Containerised labs
- Deployed in multiple AWS regions

OWASP’s instance for members:
https://secureflag.owasp.org

Culture Building



Security Champions playbook

Identify teams

- Enumerate products and services
- List teams per each product
- Identify Product manager (responsible for product) and team manager (working directly with developers)
- Write down technologies (programming languages) used by each team

Define the role

- Measure current security state among the teams and define security goals you plan to achieve in mid-term (e.g. by using OWASP SAMM)
- Identify the places where champions could help (such as verifying security reviews, raising issues for risks in existing code, conducting automated scans etc.)
- Write down clearly defined roles, as these will be the primary tasks for newly nominated champions to work on

Nominate champions

- Introduce the idea and role descriptions and get approvals on all levels - both from product and engineering managers, as well as from top management
- Together with team leader identify potentially interested candidates
- Officially nominate them as part of your security meta-team

Comm channels

- Make sure to have an easy way to spread information and get feedback
- While differing from company to company, this usually includes chats (Slack/IRC channel, Yammer group, ...) and separate mailing lists
- Set up periodic sync ups - bi-weekly should be fine to start with

Knowledge base

- Build a solid internal security knowledge base, which would become the main source of inspiration for the champions
- It should include security meta-team page with defined roles, secure development best practices, descriptions of risks and vulnerabilities and any other relevant info
- Pay special attention to clear and easy-to-follow checklists, as it's usually the simplest way to get the things going

Maintain interest

- Develop your ways or choose one of the below to keep in touch and maintain the interest of the champions
- Conduct periodic workshops and encourage participation in security conferences
- Share recent appsec news (e.g. Ezine) via communication channels
- Send internal monthly security newsletters with updates, plans and recognitions for the good work
- Create champions corner with security library, conference calendar, and other interesting materials

Operation-ModSecurity Core Rule Set Flagship Project

- Set of generic attack detection rules for use with ModSecurity or compatible web application firewalls
- Sims to protect web applications from a wide range of attacks, including the OWASP Top Ten, with a minimum of false alerts.
- Provides protection against many common attack categories

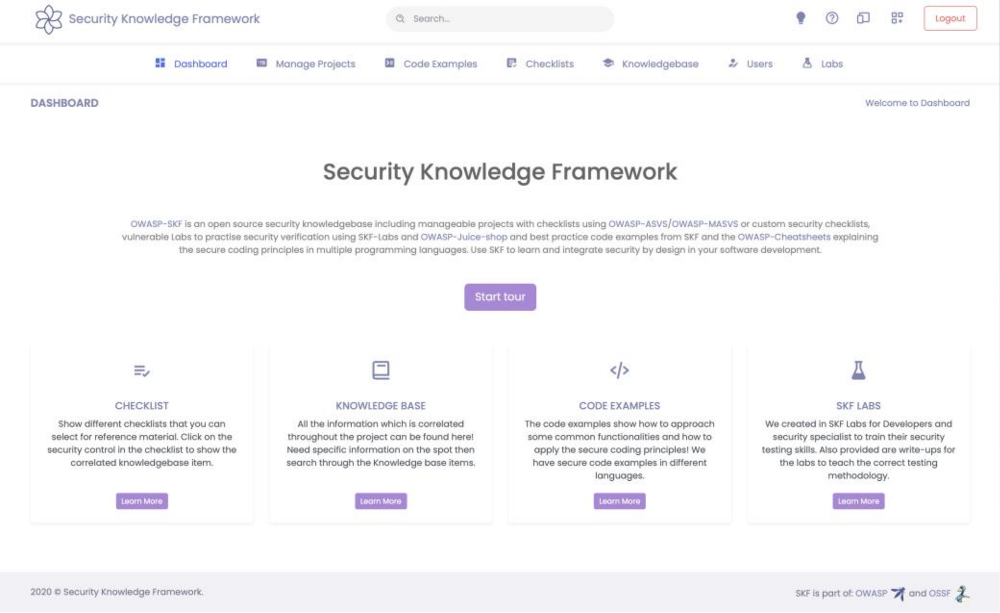
OWASP
ModSecurity
Core Rule Set
THE 1ST LINE OF DEFENSE

Requirements

- **Security Knowledge Framework (SKF)**
- **SecurityRAT**
- Application Security Verification Standard (ASVS)
- Mobile Application Security Verification Standard (MASVS)

Security Knowledge Framework (SKF) Flagship Project

- Open-source Python-Flask web application
- Uses ASVS to train you and your team in writing secure code, ***by design***



Security Knowledge Framework

OWASP-SKF is an open source security knowledgebase including manageable projects with checklists using OWASP-ASVS/OWASP-MASVS or custom security checklists, vulnerable Labs to practise security verification using SKF-Labs and OWASP-Juice-shop and best practice code examples from SKF and the OWASP-Cheatsheets explaining the secure coding principles in multiple programming languages. Use SKF to learn and integrate security by design in your software development.

CHECKLIST

Show different checklists that you can select for reference material. Click on the security control in the checklist to show the correlated knowledgebase item.

KNOWLEDGE BASE

All the information which is correlated throughout the project can be found here! Need specific information on the spot then search through the Knowledge base items.

CODE EXAMPLES

The code examples show how to approach some common functionalities and how to apply the secure coding principles! We have secure code examples in different languages.

SKF LABS

We created in SKF Labs for Developers and security specialist to train their security testing skills. Also provided are write-ups for the labs to teach the correct testing methodology.

2020 © Security Knowledge Framework.

SKF is part of: OWASP and OSSF

https://demo.securityknowledgeframework.org/

SecurityRAT Incubator Project

Security Requirement Automation Tool (SecurityRAT) focuses on automating the generation and management of security requirements

1. You specify the type of software artifact
2. SecurityRAT tells you which requirements you should fulfill
3. You decide how to handle those desired requirements
4. You persist the artifact state in an issue tracker and create tickets for the requirements where an explicit action is necessary
5. You document relevant changes in requirement compliance whenever appropriate.

Demo instance at https://securityrat.org

OWASP
Open Web Application Security Project

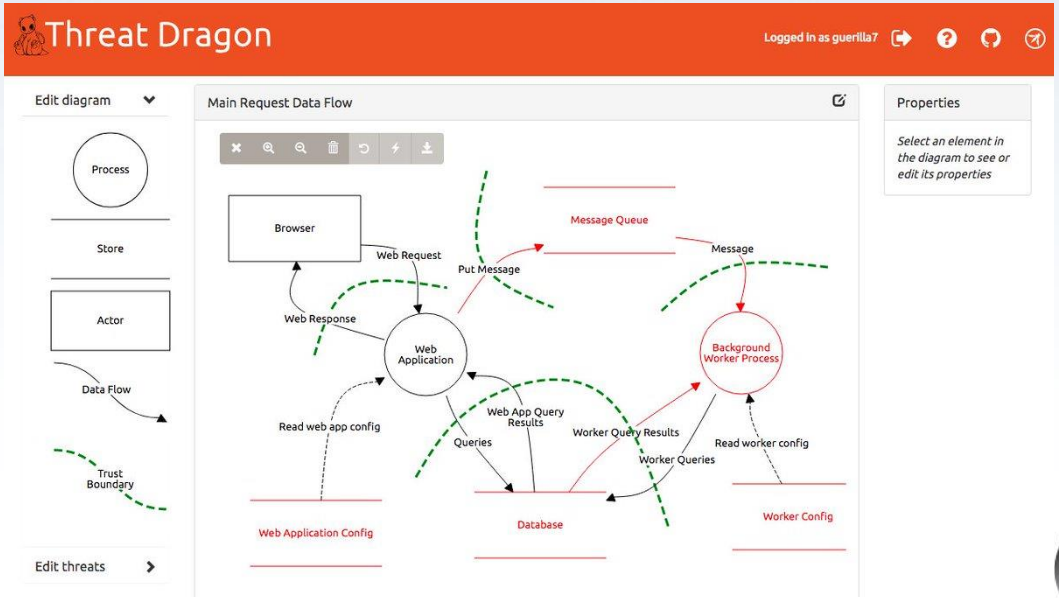
Design

Threat Modelling

- **Threat Dragon**
- **PyTM**
- **Cornucopia**
- Threat Modeling Playbook – See Seba’s talk

Threat Dragon Incubator Project

- Open-source threat model diagram creation tool
- Runs as desktop app or web app

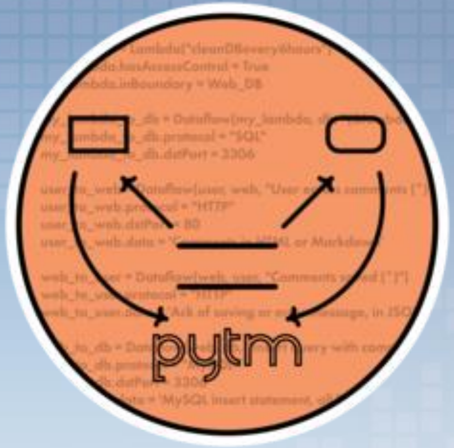


Select an element in the diagram to see or edit its properties

OWASP
Open Web Application Security Project

PyTM

Incubator Project



- A ‘Pythonic’ framework for threat modeling
- Define your system *in Python*, using the elements and properties described in the pytm framework.
- Can generate Data Flow Diagram (DFD) or Sequence Diagram views of system and threats

Cornucopia
Lab Project

- Card game to support secure coding design
- Similar design to *Elevation of Privilege (EoP)*
- Based on Secure Code Practices (SCP) – Quick Reference Guide
- Six suits:
  - Data validation and encoding
  - Authentication
  - Session management
  - Authorization
  - Cryptography
  - Cornucopia
- Cards available through OWASP, or download and print locally

OWASP
Open Web Application
Security Project

Implementation

- Documentation
  - **Top 10 Proactive Controls**
  - Go Secure Coding Practices (SCP) Guide
  - Cheat Sheet Series
- Software Composition Analysis (SCA)
  - Dependency-Check
  - **Dependency-Track**
- Libraries
  - Enhanced Security API (ESAPI)
  - CSRFGuard

Top 10 Proactive Controls Lab Project

Describes the most important control and control categories that **every architect and developer** should absolutely, 100% include in every project

- C1: Define Security Requirements
- C2: Leverage Security Frameworks and Libraries
- C3: Secure Database Access
- C4: Encode and Escape Data
- C5: Validate All Inputs
- C6: Implement Digital Identity
- C7: Enforce Access Controls
- C8: Protect Data Everywhere
- C9: Implement Security Logging and Monitoring
- C10: Handle All Errors and Exceptions

OWASP
Open Web Application
Security Project

OWASP PRO Active CONTROLS

FOR DEVELOPERS

2018 v 3.0

10 Critical Security Areas That Software Developers Must Be Aware Of

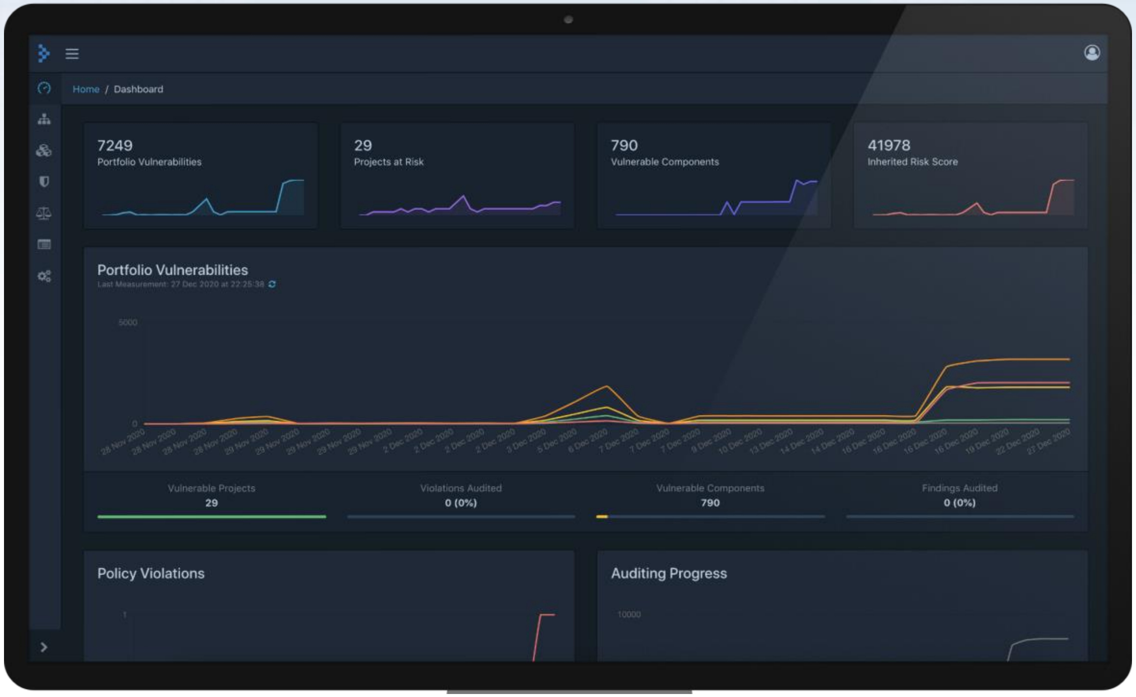
PROJECT LEADERS

KATY ANTON
JIM MANICO
JIM BIRD

OWASP
Open Web Application
Security Project

Dependency-Track Flagship Project

- Intelligent Supply Chain Component Analysis platform
- Leverages capabilities of Software Bill of Materials (SBOM)



Home / Dashboard

7249 Portfolio Vulnerabilities

29 Projects at Risk

790 Vulnerable Components

41978 Inherited Risk Score

Portfolio Vulnerabilities

Last Measurement: 27 Dec 2020 at 22:25:38

Vulnerable Projects 29

Violations Audited 0 (0%)

Vulnerable Components 790

Findings Audited 0 (0%)

Policy Violations

Auditing Progress

OWASP
Open Web Application
Security Project

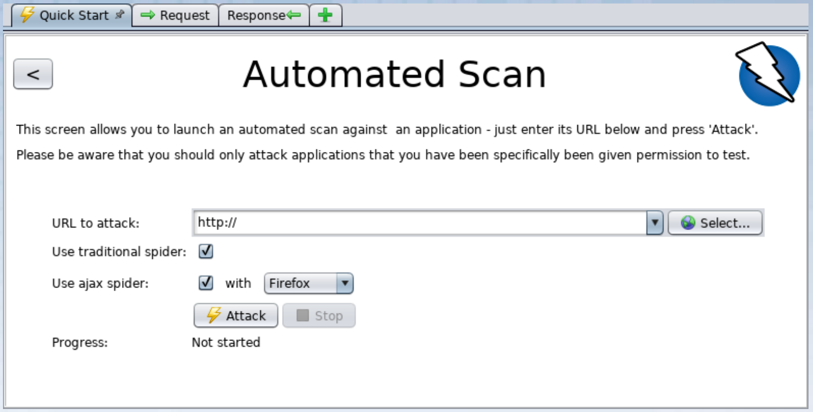
Verification

- Documentation
  - Web Security Testing Guide
  - Mobile Security Testing Guide
- Tools
  - **Zed Attack Proxy (ZAP)**
  - Attack Surface Detector
  - **Amass**
  - **Code Pulse**
  - Offensive Web Testing Framework (OWTF)
  - Nettacker
  - **DefectDojo –** Check out Rohit's talk tomorrow
- Frameworks
  - Glue
  - Dracon

OWASP
Open Web Application
Security Project

Zed Attack Proxy (ZAP) Flagship Project

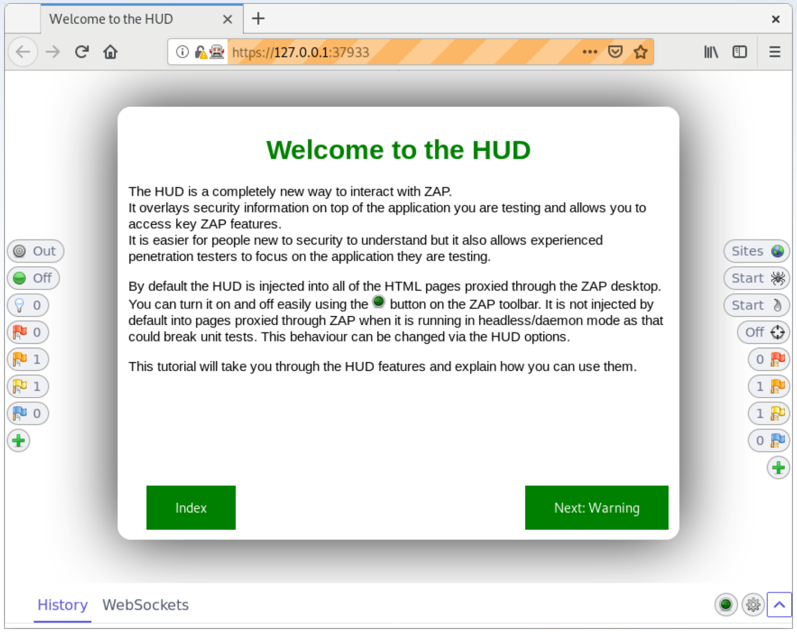
- World’s most widely used web app scanner
- Free and open-source
- Passive scanning
- Automated active scanning
- Manual exploring
- ZAP Heads-Up Display puts functionality directly in your browser



Automated Scan

This screen allows you to launch an automated scan against an application - just enter its URL below and press 'Attack'.
Please be aware that you should only attack applications that you have been specifically been given permission to test.

URL to attack: http://   Select...
Use traditional spider: ☑
Use ajax spider: ☑ with Firefox
Attack   Stop
Progress: Not started



Welcome to the HUD

The HUD is a completely new way to interact with ZAP.
It overlays security information on top of the application you are testing and allows you to access key ZAP features.
It is easier for people new to security to understand but it also allows experienced penetration testers to focus on the application they are testing.

By default the HUD is injected into all of the HTML pages proxied through the ZAP desktop. You can turn it on and off easily using the ● button on the ZAP toolbar. It is not injected by default into pages proxied through ZAP when it is running in headless/daemon mode as that could break unit tests. This behaviour can be changed via the HUD options.

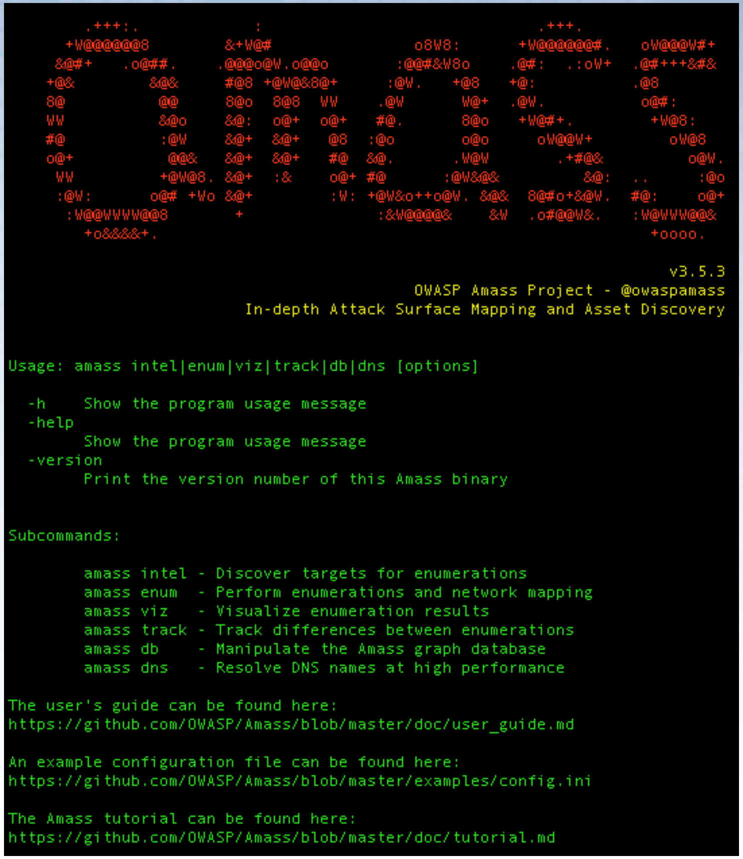
This tutorial will take you through the HUD features and explain how you can use them.

Index   Next: Warning

Amass
Flagship Project

Our Goal - In-depth DNS Enumeration, Attack Surface Mapping and External Asset Discovery!

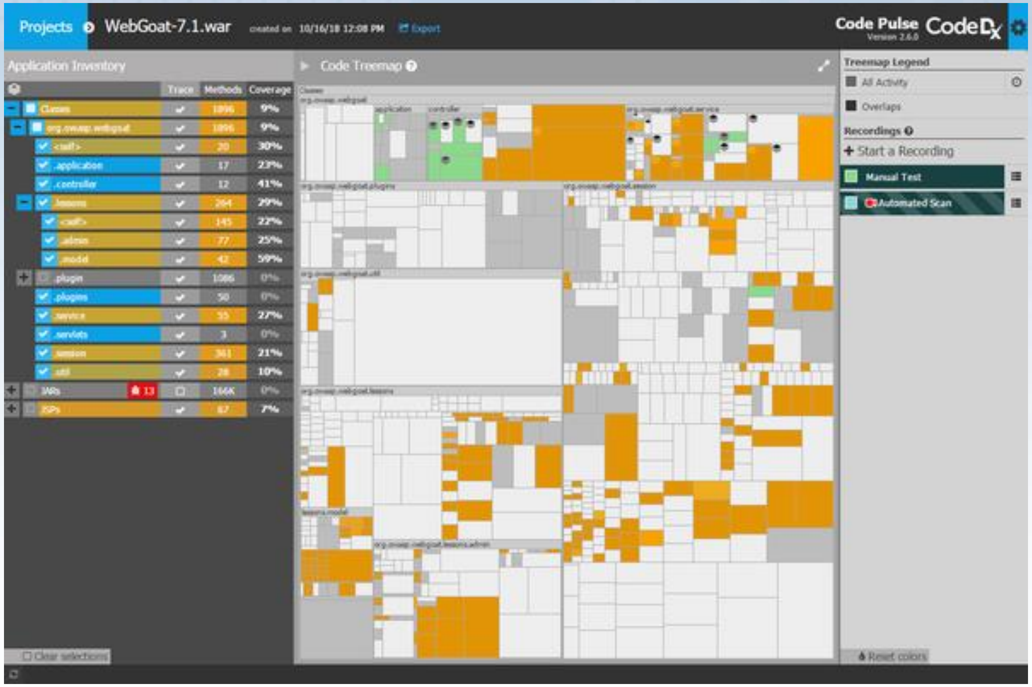
- Mapping of network attack surfaces
- External asset discovery
- Open-source information gathering and active reconnaissance techniques



```
        .+++:.             :                            .+++.
      +W@@@@@@8        &+W@#               o8W8:      +W@@@@@@#.   oW@@@W#+
     &@#+   .o@##.    .@@@o@W.o@@o       :@@#&W8o    .@#: .:oW+   .@#+++&#&
    +@&        &@&     #@8  +@W@&8@+    :@W.   +@8   +@:          .@8
    8@          @@     8@o  8@8  WW    .@W      W@+  .@W.           o@#:
    WW          &@o    &@:  o@+  o@+   #@.      8@o   +W@#+.         +W@8:
    #@          :@W    &@+  &@+   @8  :@o       o@o     oW@@W+         oW@8
    o@+          @@&   &@+  &@+   #@  &@.      .W@W       .+#@&          o@W.
     WW         +@W@8. &@+  :&    o@+ #@      :@W&@&         &@:  ..     :@o
     :@W:      o@# +Wo &@+        :W: +@W&o++o@W. &@&  8@#o+&@W.  #@:    o@+
      :W@@WWWW@@8       +              :&W@@@@&    &W  .o#@@W&.   :W@WWW@@&
        +o&&&&+.                                                    +oooo.

                                                                      v3.5.3
                                           OWASP Amass Project - @owaspamass
                         In-depth Attack Surface Mapping and Asset Discovery


Usage: amass intel|enum|viz|track|db|dns [options]

  -h    Show the program usage message
  -help
        Show the program usage message
  -version
        Print the version number of this Amass binary


Subcommands:

        amass intel - Discover targets for enumerations
        amass enum  - Perform enumerations and network mapping
        amass viz   - Visualize enumeration results
        amass track - Track differences between enumerations
        amass db    - Manipulate the Amass graph database
        amass dns   - Resolve DNS names at high performance

The user's guide can be found here:
https://github.com/OWASP/Amass/blob/master/doc/user_guide.md

An example configuration file can be found here:
https://github.com/OWASP/Amass/blob/master/examples/config.ini

The Amass tutorial can be found here:
https://github.com/OWASP/Amass/blob/master/doc/tutorial.md
```

Code Pulse
Lab Project

- Provides insight into the real-time code coverage of black box testing activities
- Cross-platform desktop application
- Agent-based runtime monitoring



Projects WebGoat-7.1.war created on 10/16/18 12:08 PM Export

Code Pulse Version 2.6.0 CodeDx

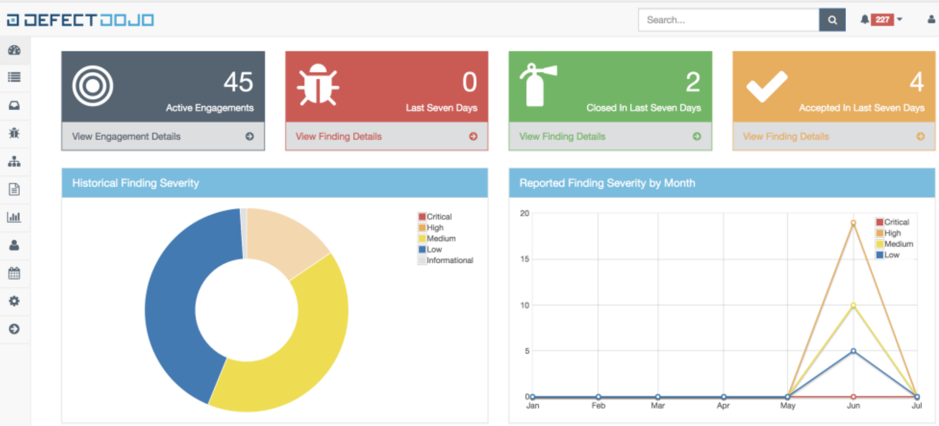
Application Inventory

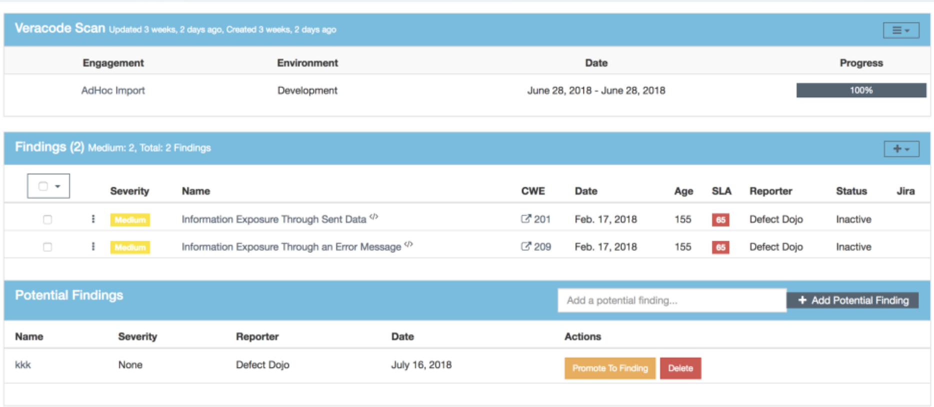
	Trace	Methods	Coverage
Classes	✓	1896	9%
org.owasp.webgoat	✓	1896	9%
<self>	✓	20	30%
.application	✓	17	23%
.controller	✓	12	41%
.lessons	✓	264	29%
<self>	✓	145	22%
.admin	✓	77	25%
.model	✓	42	59%
.plugin	✓	1086	0%
.plugins	✓	50	0%
.service	✓	55	27%
.servlets	✓	3	0%
.session	✓	361	21%
.util	✓	28	10%
JARs 13	□	166K	0%
JSPs	✓	67	7%

OWASP
Open Web Application Security Project

DefectDojo
Flagship Project

- Open-source vulnerability management tool
- Streamlines the testing process
  - Templating
  - Report generation
  - Metrics





Veracode Scan Updated 3 weeks, 2 days ago, Created 3 weeks, 2 days ago

Engagement	Environment	Date	Progress
AdHoc Import	Development	June 28, 2018 - June 28, 2018	100%

Findings (2) Medium: 2, Total: 2 Findings

□		Severity	Name	CWE	Date	Age	SLA	Reporter	Status	Jira
□	⋮	Medium	Information Exposure Through Sent Data </>	201	Feb. 17, 2018	155	65	Defect Dojo	Inactive	
□	⋮	Medium	Information Exposure Through an Error Message </>	209	Feb. 17, 2018	155	65	Defect Dojo	Inactive	

Potential Findings Add a potential finding... + Add Potential Finding

Name	Severity	Reporter	Date	Actions
kkk	None	Defect Dojo	July 16, 2018	Promote To Finding Delete

OWASP
Open Web Application Security Project

Some Closing Thoughts

- Don’t oversell – “free” tools aren’t *really* free
  - Be honest and realistic about total cost of ownership: instance charges, admin hours, etc.
- Use the right tool for your use case
  - When the OWASP tool isn’t the right one, it can still provide a cost-effective proof-of-concept
- Don’t be too proud to ask for help
  - OWASP community
  - NZ AppSec community
  - External consultants (like me)

Resources

- OWASP Integration Standards Project: https://owasp.org/www-project-integration-standards/
- OWASP SAMM: https://owaspsamm.org/
- OWASP: https://owasp.org
- Security Champions Playbook: https://github.com/c0rdis/security-champions-playbook
- Join the OWASP Slack: https://owasp-slack.herokuapp.com/

OWASP
Open Web Application
Security Project

Questions?

Thank You!

Want to chat some more?

Looking for help?

Reach out!

- OWASP: *john.dileo@owasp.org*
- Day job: *john.dileo@datacom.co.nz*
- Twitter: *@gr4ybeard*
- LinkedIn: *@john-dileo*