1000 Projects later

Security Code Scans at SAP

OWASP
The Open Web Application Security Project

- [Ruediger Bachmann](#)
  is a Development Architect at SAP AG working, as member of the central code analyses team, in the areas application security and code analysis.

- *[Dr. Achim D. Brucker](#)*
  is a Senior Researcher in the Security & Trust Program of SAP Research as well as a member of the central code analysis team of SAP AG. His research interests includes static and dynamic security testing.

Why is SAP using Static Code Analysis?

Secure Development Lifecycle at SAP

Static Code Analysis at SAP

Challenges and Outlook

OWASP
The Open Web Application Security Project

- Started rollout in June 2010
- Centrally guided by a project team
  - Definition of Security Requirements
  - Establishment of Scan Infrastructure
- Support of the most important languages
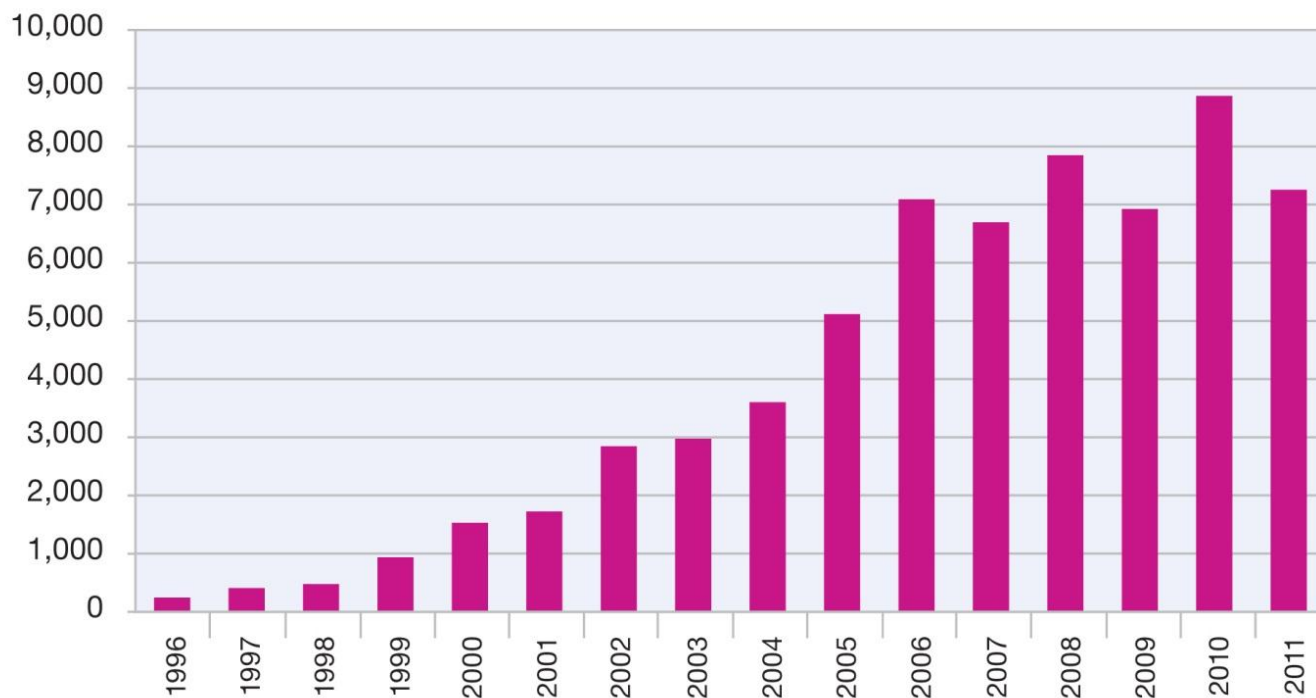- SAP development and third party code

OWASP
The Open Web Application Security Project

Vulnerability Disclosures Growth by Year
1996-2011

Year	Disclosures
1996	~200
1997	~400
1998	~450
1999	~950
2000	~1,550
2001	~1,700
2002	~2,850
2003	~2,950
2004	~3,600
2005	~5,100
2006	~7,100
2007	~6,700
2008	~7,850
2009	~6,900
2010	~8,850
2011	~7,250

Source: IBM X-Force® Research and Development

OWASP
The Open Web Application Security Project

Select Product:

SAP CRM

Code Metrics Overview
(ABAP Code Metrics only)

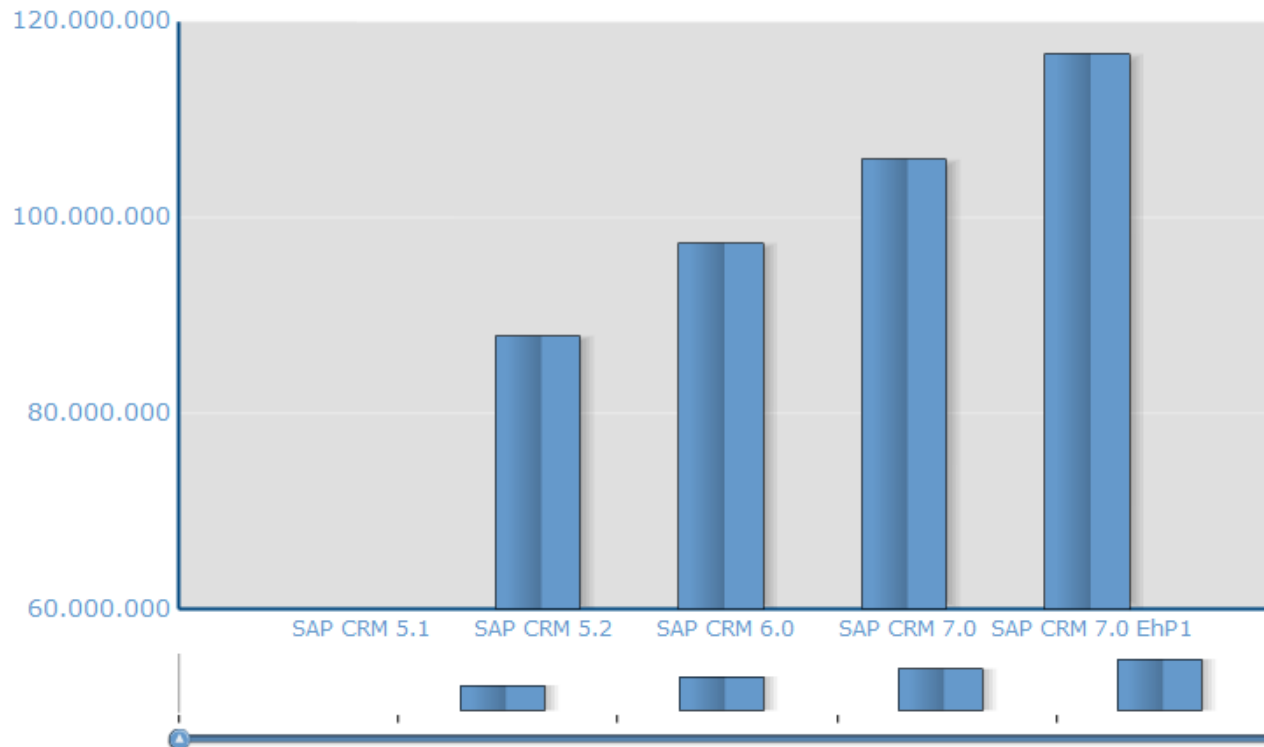**Change View:**

Chart | Table

Select Code Metric:

- Lines of Code
- Lines of Comment
- Number of Objects
- Number of Statements

120.000.000

100.000.000

80.000.000

60.000.000

SAP CRM 5.1   SAP CRM 5.2   SAP CRM 6.0   SAP CRM 7.0   SAP CRM 7.0 EhP1

OWASP
The Open Web Application Security Project

**Find Vulnerabilities
Using the Running Application**

**Find Vulnerabilities
Using the Source Code**

Manual Application
Penetration Testing

Manual Security
Code Review

Automated Application
Vulnerability Scanning

Automated Static
Code Analysis

OWASP
The Open Web Application Security Project

- Characteristics
  - Black box approach
  - Sends input to applications and analyses response
- Advantages
  - Provides concrete examples (attacks)
  - Analyze dataflows accross multiple components
- Disadvantages
  - Coverage unclear
  - Requires test system

OWASP
The Open Web Application Security Project

- Characteristics
  - White box approach
  - Analyses abstraction of the source (binary)
- Advantages
  - Explores all data paths / control flows
  - Can analyse single modules (unit test)
- Disadvantages
  - High false positive rate (not exploitable findings)
  - Does not consider application environment

Why is SAP using Static Code Analysis?

Secure Development Lifecycle at SAP

Static Code Analysis at SAP

Challenges and Outlook

OWASP
The Open Web Application Security Project

- Education:

  The prerequisite for achieving a high security quality

- Security awareness:

  Reducing the number of "built-in" security problems

- Trained persons:

  Analyze and fix vulnerabilities much more efficiently

- Trainings:

  Secure Programming, Build & Scan, Auditing, ...

OWASP
The Open Web Application Security Project

- Structure the investment of time and resources
  - – to safeguard a high level of security
  - – to ensure security standards across all areas

- Security requirements
  - – are taken into account and
  - – are implemented
  in all phases of product development

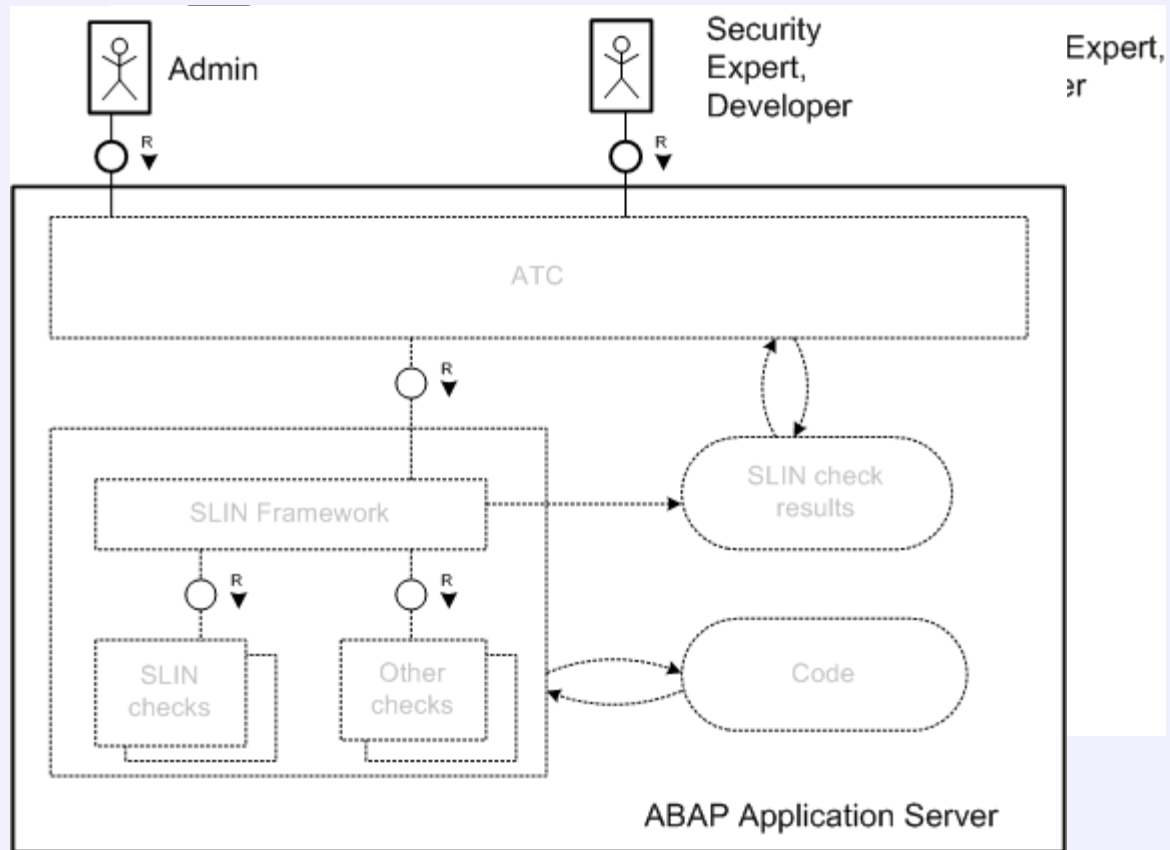**OWASP**
The Open Web Application Security Project

- Developer
  - fixes software security issues

- Security Expert
  - review scan results, decides on fixes

- Build Master
  - scans the source code, manages results

- Scrum Master
  - requests scan, assigns vulnerabilities to developers

OWASP
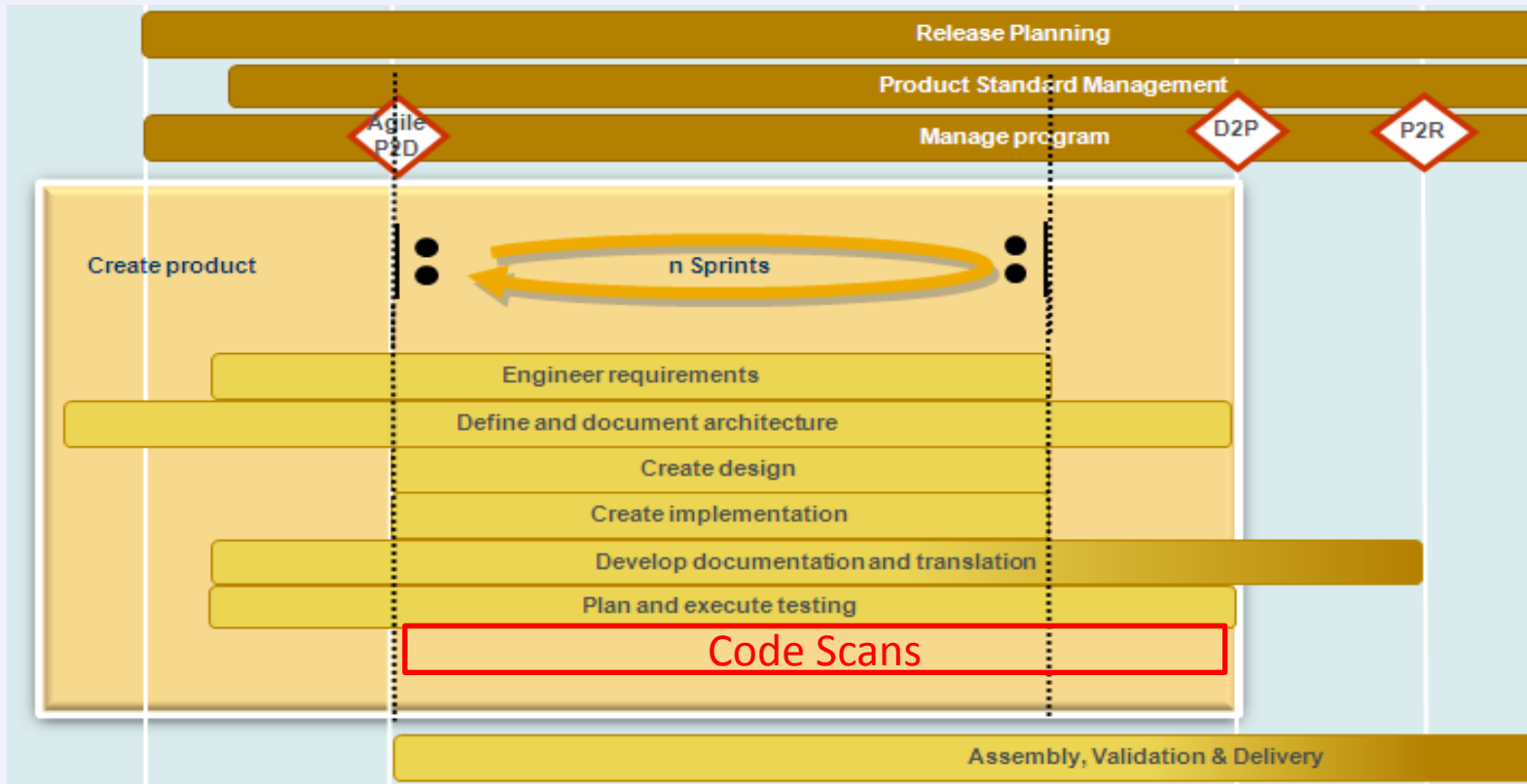The Open Web Application Security Project

OWASP
The Open Web Application Security Project

		Release Planning		
		Product Standard Management		
Agile P2D		Manage program	D2P	P2R

Create product — n Sprints

Engineer requirements
Define and document architecture
Create design
Create implementation
Develop documentation and translation
Plan and execute testing

Code Scans

Assembly, Validation & Delivery

For passing D2P Q-gate, evidence has to be provided that the source code has been scanned and exploitables have been fixed.

P2D: Planning to Development. / D2P: Development to Production. /

P2R: Production to Ramp-up (gradual roll-out to customers).

OWASP
The Open Web Application Security Project

- Third party code
  - Open Source libaries and frameworks
  - Freeware
  - other third party components
- Different approaches
  - SAST analysis by SAP
  - Certificate from vendor
  - SLA with vendor

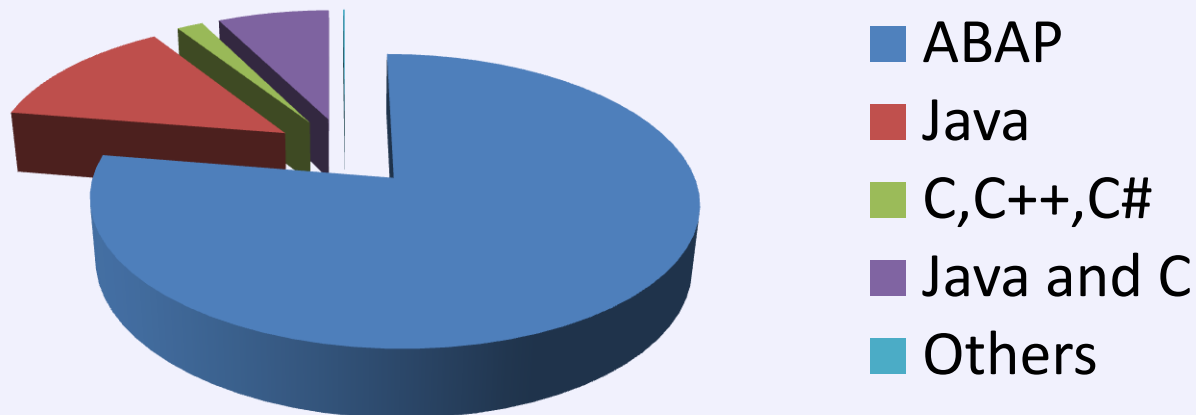Why is SAP using Static Code Analysis?

Secure Development Lifecycle at SAP

Static Code Analysis at SAP

Challenges and Outlook

- Over 2000 developers are using SAST tools
- Over 500 MLOC scanned

- ABAP
- Java
- C,C++,C#
- Java and C
- Others

Statistics Jan 2012

OWASP
The Open Web Application Security Project

Language	Scan Application
ABAP	SAP
C/C++	Coverity
Others	HP/Fortify

OWASP
The Open Web Application Security Project

- SAP on Corporate Security Requirements
  - SAP Applications shall be free of backdoors
  - SQL injection vulnerabilities shall be avoided
  - Cross-Site Scripting vulnerabilities shall be prevented
  - Directory traversal vulnerabilities shall be prevented
  - The system shall be protected against buffer overflow vulnerabilities
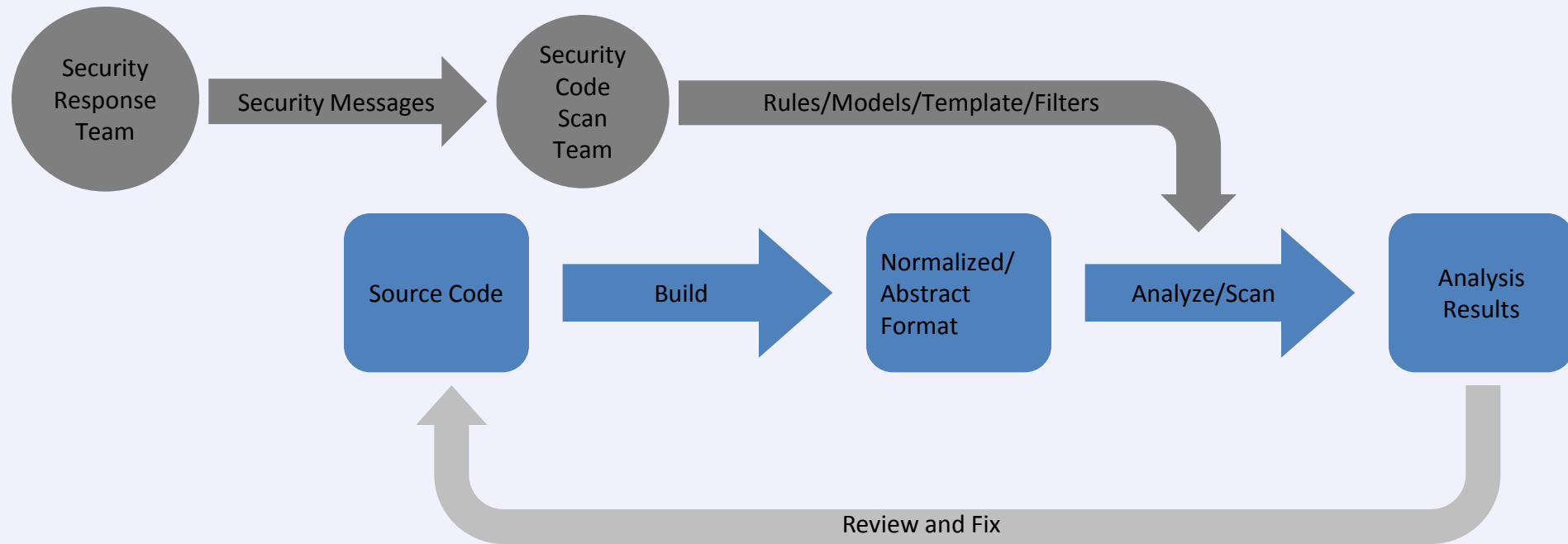- OWASP Top 10
- CWE/SANS Top 25 2011
- CVE

The Open Web Application Security Project

- Collect feedback from the
  - Product Security Response Team
  - Development Teams
- Develop rules/models to improve the scans
- Continuously improve the infrastructure
- Continuously improve the rollout process

OWASP
The Open Web Application Security Project

Security Response Team → Security Messages → Security Code Scan Team

Rules/Models/Template/Filters

Source Code → Build → Normalized/Abstract Format → Analyze/Scan → Analysis Results

Review and Fix

•Further input channels: Development teams, internal research, scan reviews, code reviews

OWASP
The Open Web Application Security Project

- Scans have to be obligatory
  but not introduced 'brute force'

- Establish Secure Development Life Cycle
  make scans a natural part of development

- Plan carefully

  – Do not start with scans right before Dev. Close

  – Do it regularly (nightly)

- Do not introduce changes during development

Why is SAP using Static Code Analysis?

Secure Development Lifecycle at SAP

Static Code Analysis at SAP

Challenges and Outlook

OWASP
The Open Web Application Security Project

C, C++, CVS, Perforce, ABAP Repository, Git, Ant, Maven, Make, Japro, Mapro, Hudson, Jenkins MS Foundation Server, Mobile, Android, iOS, Objective-C, Java, Local Build, Central Build, Different Development Cycles, Development on/for different OS like Linux Windows, Unicode, Non-Unicode, Developer IDE Support (Plugins for Eclipse, Visual Studio), Silverlight, BSP, Flash, TCL/TK, Erlang, Perl, JSP, TagLib, OSS, HTML5/JavaScript, Convenient Access to Central Scan Result Repositories, Provide a Secure Code Scan Infrastructure, Convince everybody that code scanning helps to improve software, Open Source, Third Party Software, Merge Threat Modeling with Code Scans, Q-Gate-Reporting, Establish a stable plan-scan-check-correct workflow, …

OWASP
The Open Web Application Security Project

- Assume the following index.html:

```
<TITLE>Welcome!</TITLE>
Hi
  <SCRIPT>
     var pos=document.URL.indexOf("name=")+5;
     document.write(document.URL.substring(pos,document.URL.length));
  </SCRIPT>
Welcome to our system
```

 and a call

```
index.html?name=<script>alert(document.cookie)</script>
```

- resulting in a DOM-based XSS attack

- DOM implementations are Browser specific

- ## A simple script statement

```
<script language="javascript">
    document.write("<script>src='other.js'></script>");
</script>
```

- ## Dynamically creating script tags

```
<script>
  var oHead = document.getElementsByTagName('HEAD').item(0);
  var oScript= document.createElement("script");
  oScript.type = "text/javascript";
  oScript.src="other.js";
  oHead.appendChild( oScript);
</script>
```

- ## Or using eval() directly (not shown here)

OWASP
The Open Web Application Security Project

- Combining the comlexity of two worlds

```
var entry=JSON.parse(data);
query = "insert into \"FOO(".NAME")\";
var conn = $.db.getConnection();
conn.execute(query);
```

- SAST works very well for
  - "traditional" programming languages
  - Analyzing data paths within one technology
- Many new development uses JavaScript
  - HTML5/JavaScript UIs
  - Server-side JavaScript
- JavaScript
  - Untyped
  - Dynamic programming model

OWASP
The Open Web Application Security Project

You cannot pay people well enough, to do proper code audits.

I tried.

(Yaron Minsky, Jane Street Capital)

Thank you!