1000 Projects later
Security Code Scans at SAP

About Us

• **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.

Agenda

Why is SAP using Static Code Analysis?
Secure Development Lifecycle at SAP
Static Code Analysis at SAP
Challenges and Outlook

Overview

• 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
**Dynamic Security Testing**

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

- 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

Agenda

Why is SAP using Static Code Analysis?
Secure Development Lifecycle at SAP
Static Code Analysis at SAP
Challenges and Outlook

First Step: Security Training

- 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, ...

Secure Development Lifecycle (SDLC) at SAP

- 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
The Different Roles

- **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

---

SAP Secure Software
Development Life Cycle

- 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).

---

Third Party Code

- **Third party code**
  - Open Source libraries 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

Code Scan Facts
• Over 2000 developers are using SAST tools
• Over 500 MLOC scanned

Statistics Jan 2012

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

Security Scan Tools used at SAP

Security Requirements
• 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
Continuous Improvement

- 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

Input to Improve Code Scans

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

Lessons Learned

- 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

Agenda

- Why is SAP using Static Code Analysis?
- Secure Development Lifecycle at SAP
- Static Code Analysis at SAP
- Challenges and Outlook
Assume the following index.html:

```html
<TITLE>Welcome!</TITLE>

Hi

...@

</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

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

Dynamically creating script tags

```html
<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)

Combining the complexity of two worlds

```javascript
var entry=JSON.parse(data);
query = "insert into "FOO(.NAME)";
var conn = $.db.getConnection();
conn.execute(query);
```
Challenges: Current Trends

• 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

Conclusion

You cannot pay people well enough, to do proper code audits.
I tried.
(Yaron Minsky, Jane Street Capital)

Thank you!