YOUR INDEPENDENT HPE SOFTWARE COMMUNITY

Hear How SAP Ensures their Applications are Secure Against Data Breaches using HPE Fortify

April 6, 2016

Brought to you by

Hewlett Packard
Enterprise

vivit
YOUR INDEPENDENT HPE SOFTWARE COMMUNITY

Hosted By:

Robert Linton
V.P. Application Lifecycle Management
CorTechs Inc.
TQA SIG Leader

Today's Speaker

Barbara Kohde
Presales Lead APJ
Quality Assurance and Security Solutions
SAP Australia Pty Ltd

Housekeeping

- This “LIVE” session is being recorded

  Recordings are available to all Vivit members

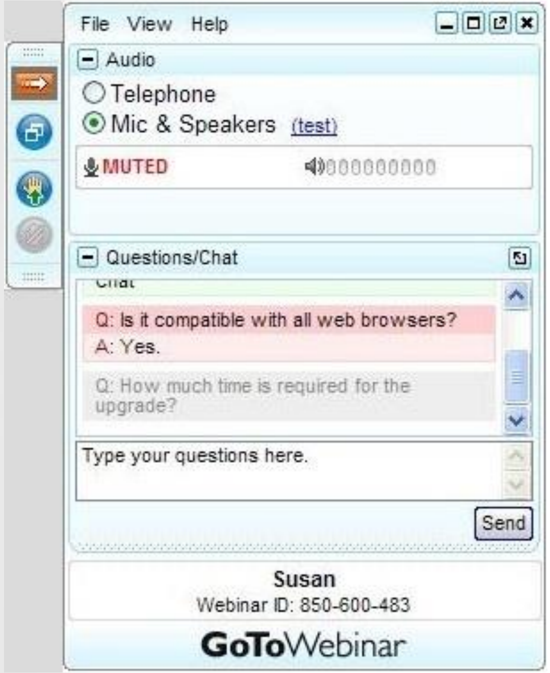
- <u>Session Q&A:</u>

  Please type questions in the Questions Pane

Webinar Control Panel

Toggle View Window between
Full screen/window mode.

Questions



○ Telephone
● Mic & Speakers (test)

MUTED

Q: Is it compatible with all web browsers?
A: Yes.

Q: How much time is required for the upgrade?

Type your questions here.

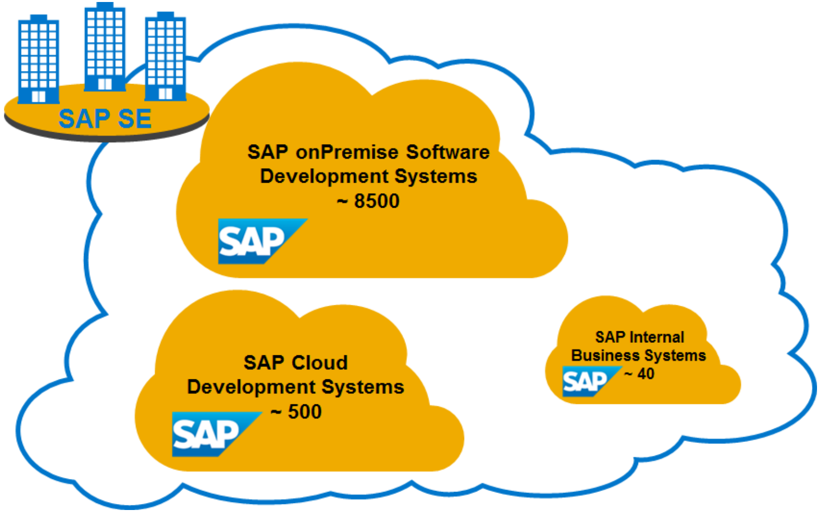
Susan
Webinar ID: 850-600-483

SAP Application Security

Barbara Kohde
Presales Lead APJ
SAP Quality Assurance Solutions

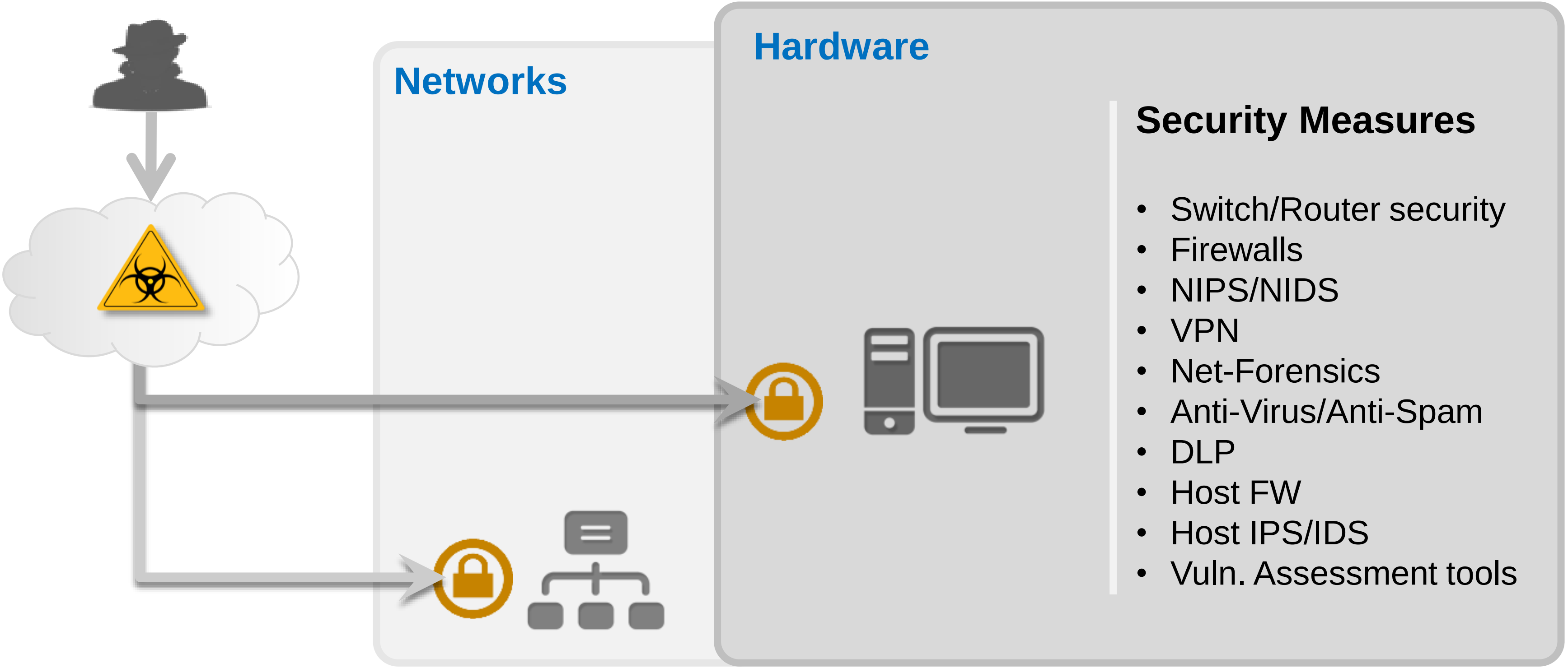
SAP – Fast Facts

- Leading global Enterprise Software company
- Headquarter in Walldorf, Germany
- 77,000 Employees in 130+ countries
- 100+ Innovation and development centers
- 300,000 customers in 190 countries
- 87% Of Forbes Global 2000 are SAP customers
- 85 mil. Subscribers in our cloud user base
- €20.8b Annual revenue (IFRS) in FY2015


SAP SE
SAP onPremise Software Development Systems ~ 8500
SAP
SAP Cloud Development Systems ~ 500
SAP
SAP Internal Business Systems ~ 40
SAP

Application Security Challenges

Cyber attacks in the past

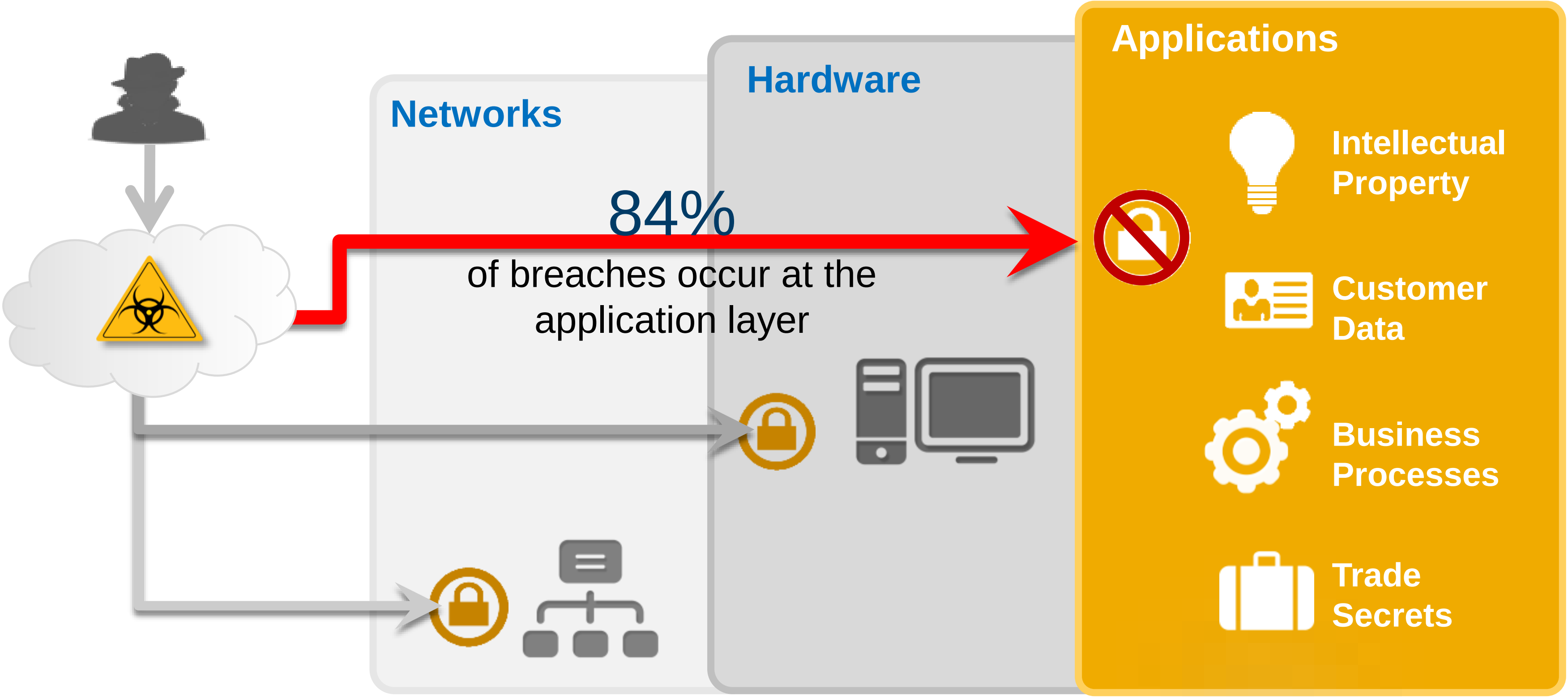


Security Measures

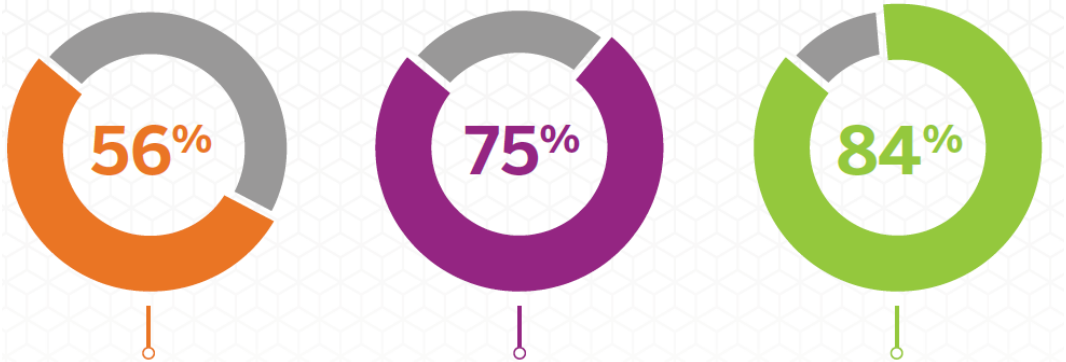
- Switch/Router security
- Firewalls
- NIPS/NIDS
- VPN
- Net-Forensics
- Anti-Virus/Anti-Spam
- DLP
- Host FW
- Host IPS/IDS
- Vuln. Assessment tools

Application Security Challenges

Cyber attackers are now targeting applications



Application Security is the Frontier Now and Future



Security weaknesses reveal information about application or users.

Of mobile applications fail basic security tests*

of breaches occur at the application layer**

68% increase in mobile application vulnerability disclosures

Developers/QA are focused on functionality & performance

Security professionals are overwhelmed by applications

Polling Question 1:

Is application security an important topic in your organization?

1) Yes

2) No

3) Don’t know

What causes software security problems?

All security vulnerabilities in software are the result of security bugs, or defects, within the software.

```
struct group_info init_groups = { .usage = ATOMIC_INIT(2) };
struct group_info *groups_alloc(int gidsetsize){
    struct group_info *group_info;
    int nblocks;
    int i;

    nblocks = (gidsetsize + NGROUPS_PER_BLOCK - 1) / NGROUPS_PER_BLOCK;
    /* Make sure we always allocate at least one indirect block pointer */
    nblocks = nblocks ? : 1;
    group_info = kmalloc(sizeof(*group_info)
```

In most cases, these defects are created by two primary causes:

- non-conformance, or a failure to satisfy requirements.
- an error or omission in the software requirements.

Source: Wikipedia

But my SAP applications are all in-house…

Only my internal users access my systems so I am secure….

Almost all business applications have **web or mobile access** now

Losses through **internal fraud** constituted **7% of yearly revenue average**

(Source: ERPScan Survey 2011)

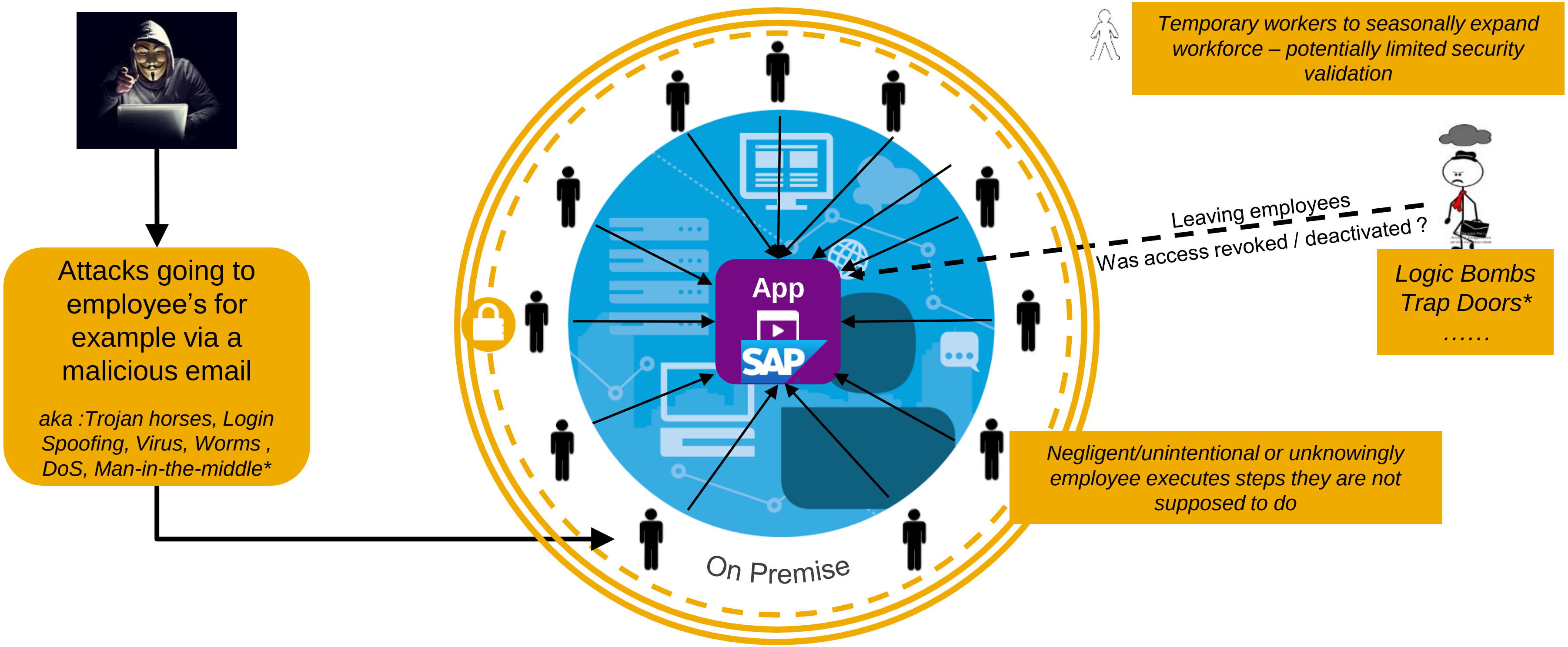
We have implemented Access Control mechanisms so my users can only access what they are supposed to…

Even with access control mechanisms in place the custom application code itself might still be vulnerable

Security considerations for Internal only applications

Examples of attacks for internal only applications

Your Digital Enterprise



Attacks going to employee's for example via a malicious email

*aka :Trojan horses, Login Spoofing, Virus, Worms , DoS, Man-in-the-middle**

Temporary workers to seasonally expand workforce – potentially limited security validation

Leaving employees

Was access revoked / deactivated ?

Logic Bombs Trap Doors ……*

Negligent/unintentional or unknowingly employee executes steps they are not supposed to do

Are those users secure ? Are those applications secure ? Is the data secure ?

And let’s not forget…

All enterprise organizations run major software operations.

Business Processes nowadays span across all these applications which can expose also in-house implementations to security breaches.

ERP

inventory management

wikis

PoS

supply chain

mobile apps

HR

billing

website

payments

order entry

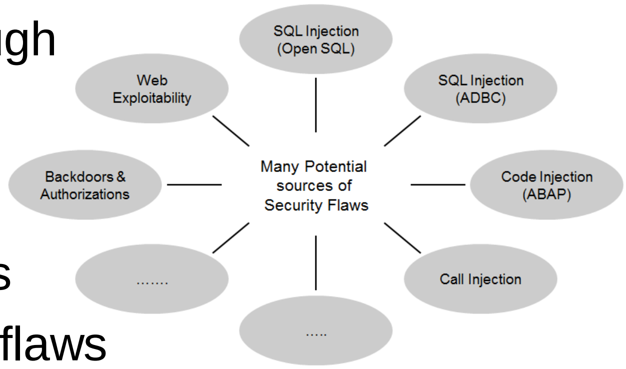
Embedded software

CRM

The Challenge of Security

In order to secure an application, **all** of its components, functions, connected applications and the related threats must be understood

In order to break an application, only **one flaw** in any of its components/functions or the infrastructure may be enough



The problem:

- Each new technology brings with it new vulnerabilities
- Hackers are increasingly aware of typical application flaws
- Firewalls, intrusion detection systems, signatures and encryption alone cannot make an application secure

Polling Question 2:

Who in your organization is responsible for application security?

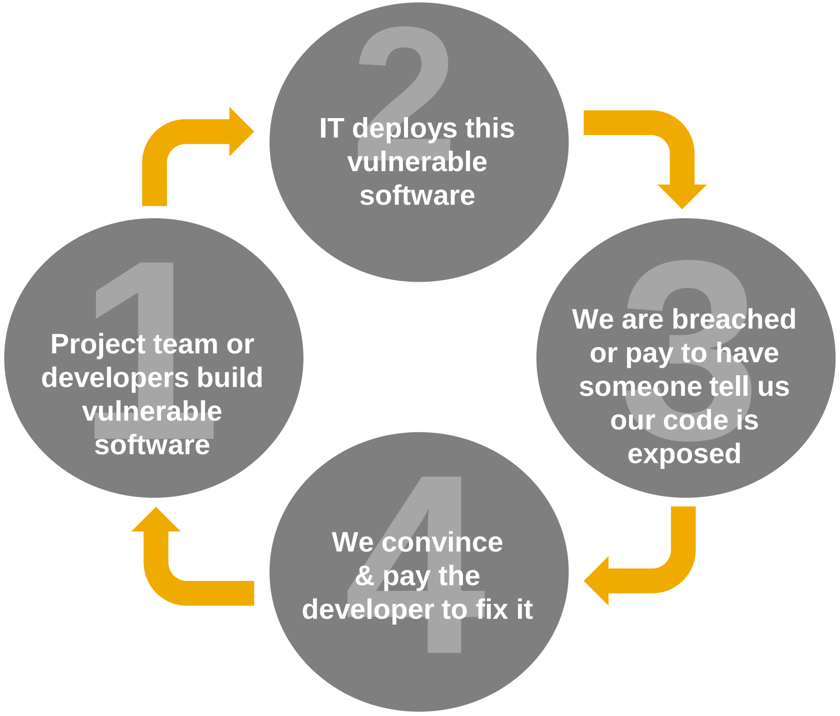
1) **Development**

2) **Security Team**

3) **CISO (Chief Information Security Officer) or Head of Security**

4) **Don’t know**

Typical approach to Application Security in majority of large enterprise organizations is…REACTIVE and COSTLY



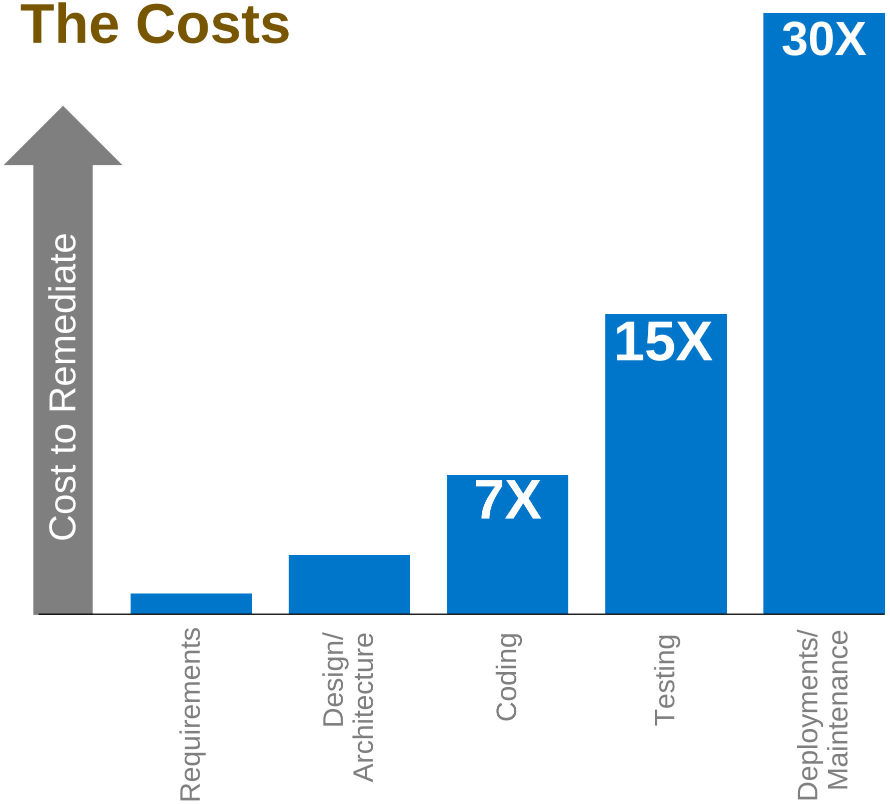
1 Project team or developers build vulnerable software

2 IT deploys this vulnerable software

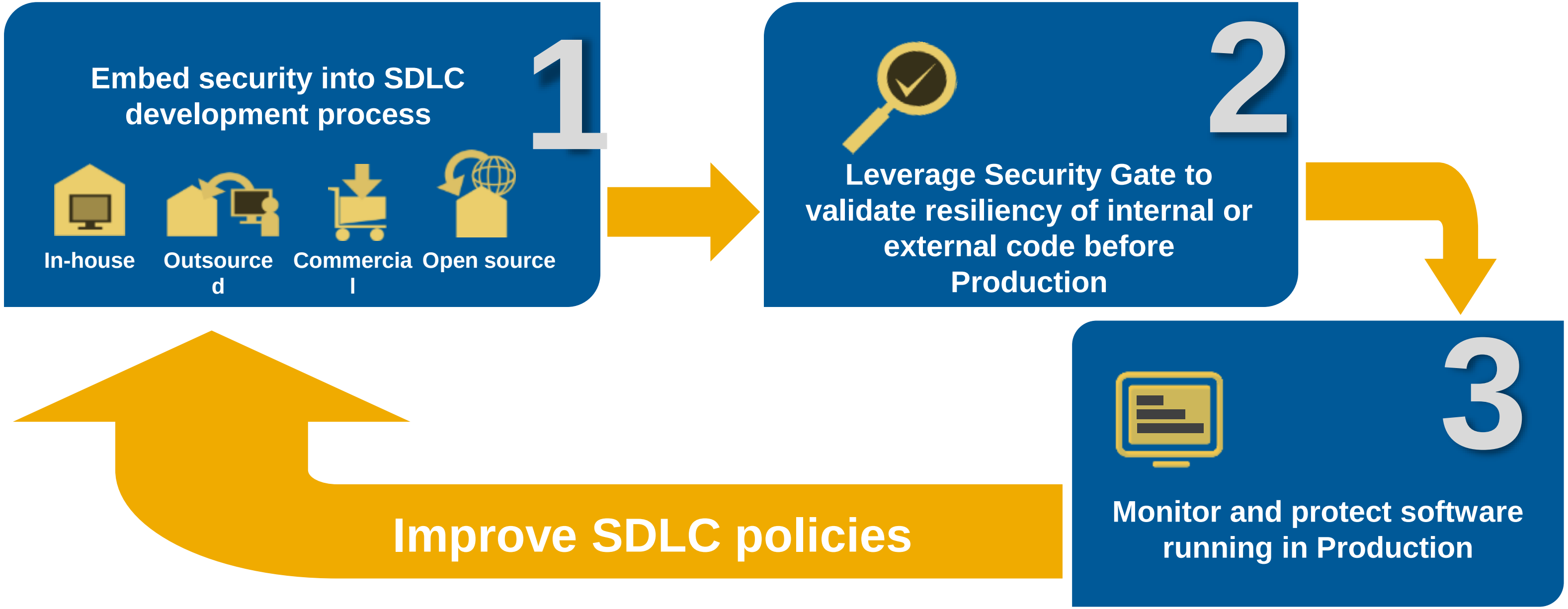
3 We are breached or pay to have someone tell us our code is exposed

4 We convince & pay the developer to fix it

The Costs



The right approach > systematic, proactive



1

Embed security into SDLC development process

In-house

Outsourced

Commercial

Open source

2

Leverage Security Gate to validate resiliency of internal or external code before Production

3

Monitor and protect software running in Production

Improve SDLC policies

This is application security

SAP’s recommended solution…. Software Security Assurance

Software Security Assurance: Process

Software Security Assurance:

- is a process that helps design and implement software that protects the data and resources contained in and controlled by that software.

- is the process of ensuring that software is designed to operate at a level of security that is consistent with the potential harm that could result from the loss, inaccuracy, alteration, unavailability, or misuse of the data and resources that it uses, controls, and protects.

Software Security Assurance: People involved

Code review

	The Task of Creating Secure Code								
	Program Manager (PgM)	Project Manager (PM)	Team Lead	(TM)	(PO)	n & Support	curity Team	anager	Architect
Scan Code									
Triage Code									
Assign Defects									
Fix Code									
Submit Exceptions									

R: Responsible
A=Accountable
C= Consulted
I=Informed

Project Owner	Approve exceptions to the security requirements for the application
Project Manager	Aid the Development Lead in the exception application process
Information Security Team	Establish the security testing process. Assist with code scanning results triage while providing consultation in fixing security defects and submitting exceptions.
Test Manager	Informed as to the identification and remediation of security issues in the application, informed on the scanning process, and consulted on issues submitted for exception
Development Team	Responsible for scanning the application on a regular basis, assigning the defects to individual developers, and submitting exceptions for issues not to be fixed. Also supports the audit and code fixing processes.
Developer	Help set up the automated code scanning, provide contextual input on issues identified during scanning, and fix defects identified in the code

Example Only

Software Security Assurance: Solutions

1

Application Assessment

Assess

Find security vulnerabilities in any type of software

SAP, Mobile, Web, Infrastructure

2

Software Security Assurance (SSA)

In-house Outsourced Commercial Open source

Assure

Fix security flaws in source code before it ships

Secure SDLC

3

Application Protection

Protect

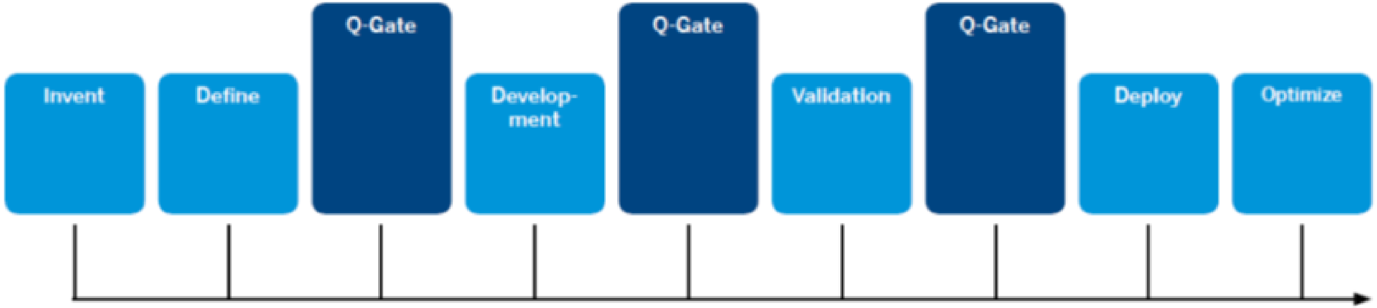
Fortify applications against attack in production

Logging, Threat Protection

How SAP addresses Application Security – The Process

SAP runs Security Tests on all SAP Applications and the standard code as delivered by SAP using the SAP development framework - a rigorous development process that embeds security throughout the product innovation lifecycle .
The framework consists of a core set of rules, product and process standards, and support for best practices that cover the entire software lifecycle – invent, define, develop, validate, deploy, and optimize.

At each step in the lifecycle, SAP software security is checked at quality gates (Q-gates). Q-gates are mandatory milestones, to determine if the software can move to the next lifecycle phase.



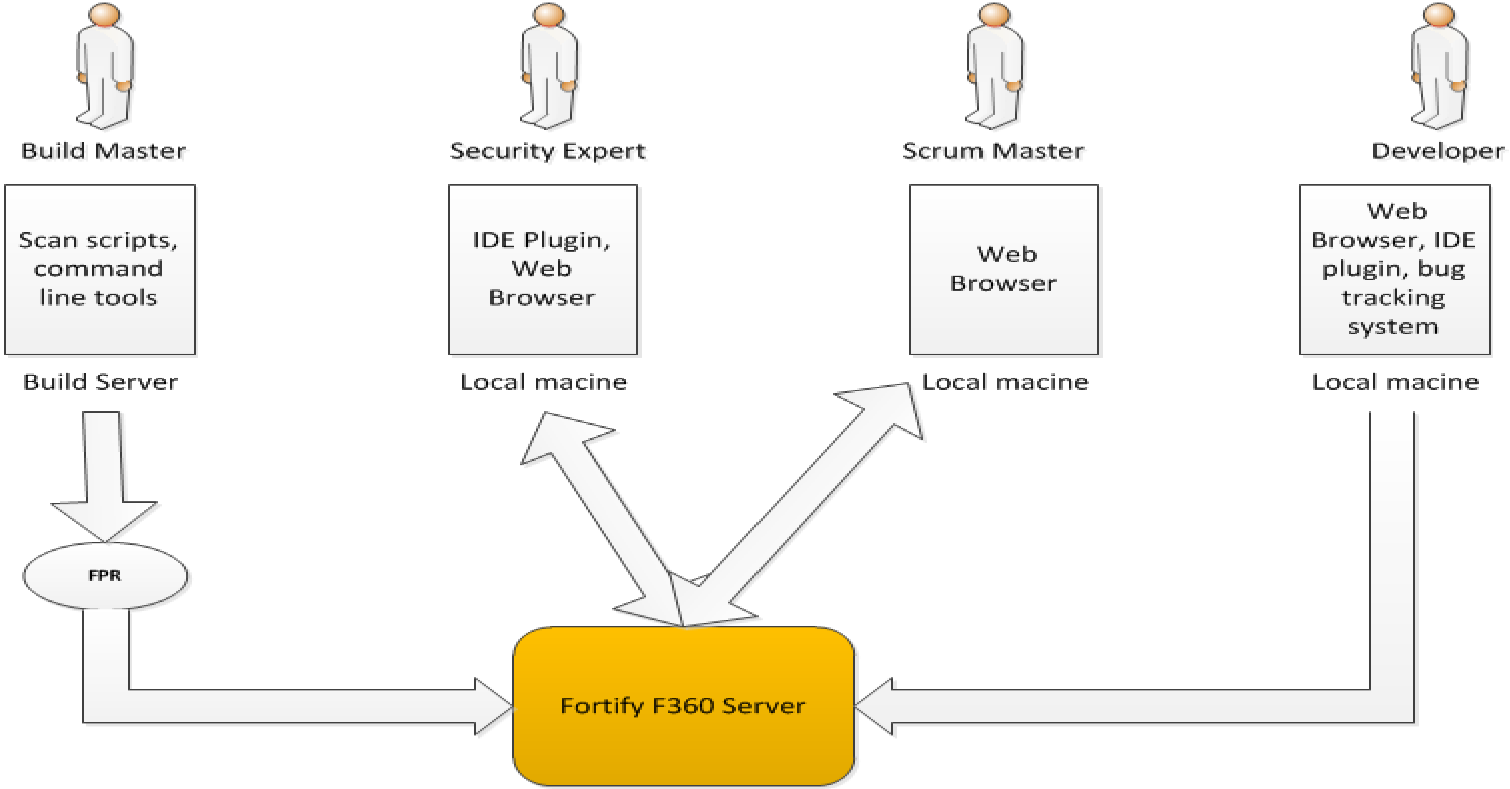
How SAP addresses Application Security – The Process

All products developed in Fortify supported language, must be scanned for violation of security requirements

- All findings should be audited and exploitables fixed before delivery to customers
- Starting development from **the Date**
- Have a release to customer date five months later after **the Date**

How SAP addresses Application Security – The People

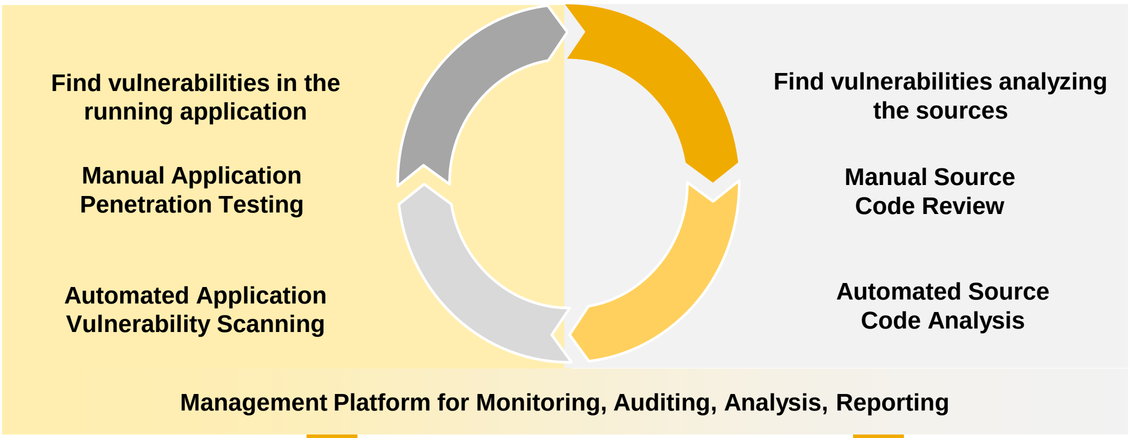
How SAP addresses Application Security – The People



How SAP addresses Application Security – The Solutions

DAST
Dynamic Application Security Testing

SAST
Static Application Security Testing



Find vulnerabilities in the running application

Manual Application Penetration Testing

Automated Application Vulnerability Scanning

Find vulnerabilities analyzing the sources

Manual Source Code Review

Automated Source Code Analysis

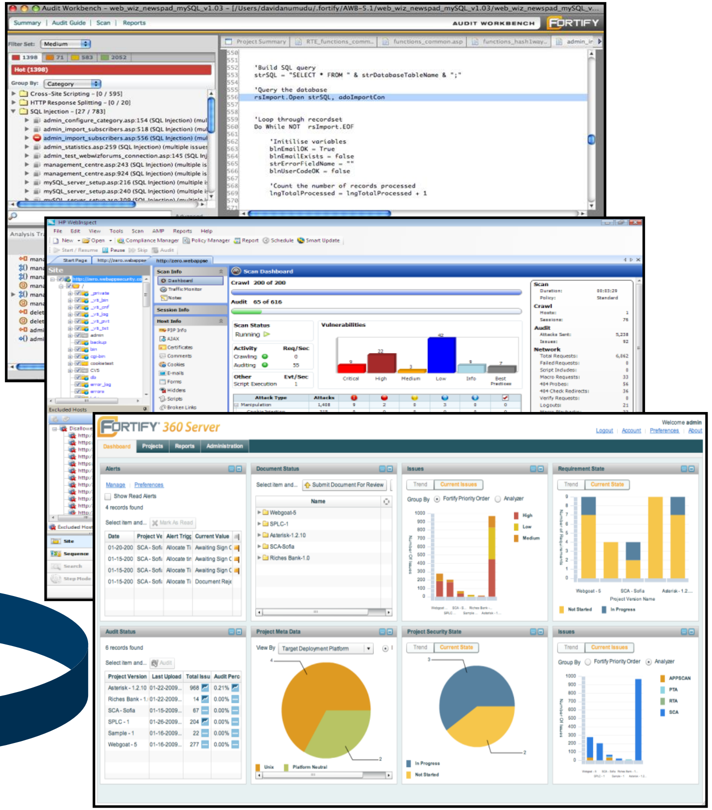
Management Platform for Monitoring, Auditing, Analysis, Reporting

non-ABAP non-SAP

SAP Fortify by HP &

ABAP

SAP NetWeaver Application Server, add-on for code vulnerability analysis (CVA)



Audit Workbench – web_wiz_newspad_mySQL_v1.03 – [/Users/davidanumudu/.fortify/AWB-5.1/web_wiz_newspad_mySQL_v1.03/web_wiz_newspad_mySQL_v...
Summary | Audit Guide | Scan | Reports
AUDIT WORKBENCH
FORTIFY
Filter Set: Medium
1398
71
583
2052
Hot (1398)
Group By: Category
Project Summary
RTE_functions_comm...
functions_common.asp
functions_hash1way...
admin_ir
HP WebInspect
File Edit View Tools Scan AMP Reports Help
New Open Compliance Manager Policy Manager Report Schedule Smart Update
Start / Resume Pause Skip Audit
Start Page
Scan Dashboard
Vulnerabilities
FORTIFY 360 Server
Dashboard Projects Reports Administration
Alerts
Document Status
Issues
Requirement State
Audit Status
Project Meta Data
Project Security State
Issues

Cross-Site Scripting – [0 / 595]
HTTP Response Splitting – [0 / 20]
SQL Injection – [27 / 783]
admin_configure_category.asp:154 (SQL Injection) (mul
admin_import_subscribers.asp:518 (SQL Injection) (mul
admin_import_subscribers.asp:556 (SQL Injection) (mul
admin_statistics.asp:259 (SQL Injection) (multiple issues
admin_test_webwizforums_connection.asp:145 (SQL Inj
management_centre.asp:243 (SQL Injection) (multiple is
management_centre.asp:924 (SQL Injection) (multiple is
mySQL_server_setup.asp:216 (SQL Injection) (multiple i
mySQL_server_setup.asp:240 (SQL Injection) (multiple i

```
'Build SQL query
strSQL = "SELECT * FROM " & strDatabaseTableName & ";"

'Query the database
rsImport.Open strSQL, adoImportCon


'Loop through recordset
Do While NOT  rsImport.EOF

    'Initilise variables
    blnEmailOK = True
    blnEmailExists = false
    strErrorFieldName = ""
    blnUserCodeOK = false

    'Count the number of records processed
    lngTotalProcessed = lngTotalProcessed + 1
```

Finding security issues at design time instead of in production is easier and less expensive!

SAP CVA – Review custom ABAP code & fix vulnerabilities

Scan efficiently

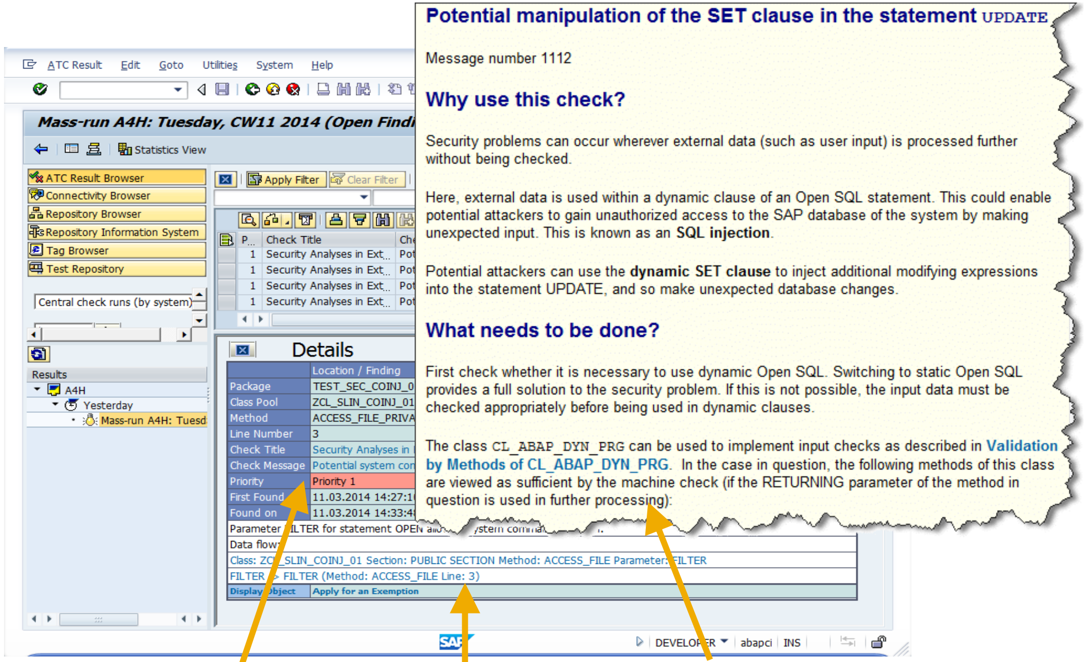
- Scanning directly from within the ABAP development environment
- Reduced false-positive rate by dataflow analysis

Developer guidance

- Detailed help and explanations to all errors
- Assistance to find the right location for the fix
- Approval workflows for false positives included

Integration

- Integrated into standard ABAP check frameworks, SAP transport system and ABAP Test Cockpit (ATC)



P...	Check Title	Che
1	Security Analyses in Ext...	Pot
1	Security Analyses in Ext...	Pot
1	Security Analyses in Ext...	Pot
1	Security Analyses in Ext...	Pot

	Location / Finding
Package	TEST_SEC_COINJ_0
Class Pool	ZCL_SLIN_COINJ_01
Method	ACCESS_FILE_PRIVA
Line Number	3
Check Title	Security Analyses in
Check Message	Potential system con
Priority	Priority 1
First Found	11.03.2014 14:27:1
Found on	11.03.2014 14:33:4

Parameter FILTER for statement OPEN allo... system comma...

Data flow:

Class: ZCL_SLIN_COINJ_01 Section: PUBLIC SECTION Method: ACCESS_FILE Parameter: FILTER

FILTER -> FILTER (Method: ACCESS_FILE Line: 3)

Potential manipulation of the SET clause in the statement `UPDATE`

Message number 1112

Why use this check?

Security problems can occur wherever external data (such as user input) is processed further without being checked.

Here, external data is used within a dynamic clause of an Open SQL statement. This could enable potential attackers to gain unauthorized access to the SAP database of the system by making unexpected input. This is known as an **SQL injection**.

Potential attackers can use the **dynamic SET clause** to inject additional modifying expressions into the statement UPDATE, and so make unexpected database changes.

What needs to be done?

First check whether it is necessary to use dynamic Open SQL. Switching to static Open SQL provides a full solution to the security problem. If this is not possible, the input data must be checked appropriately before being used in dynamic clauses.

The class `CL_ABAP_DYN_PRG` can be used to implement input checks as described in **Validation by Methods of CL_ABAP_DYN_PRG**. In the case in question, the following methods of this class are viewed as sufficient by the machine check (if the RETURNING parameter of the method in question is used in further processing):

Security Vulnerability

Line of Code

Recommended Fix

Static Security Scanning – non-ABAP and non-SAP applications
SAP Fortify by HPE - Static Code Analyzer

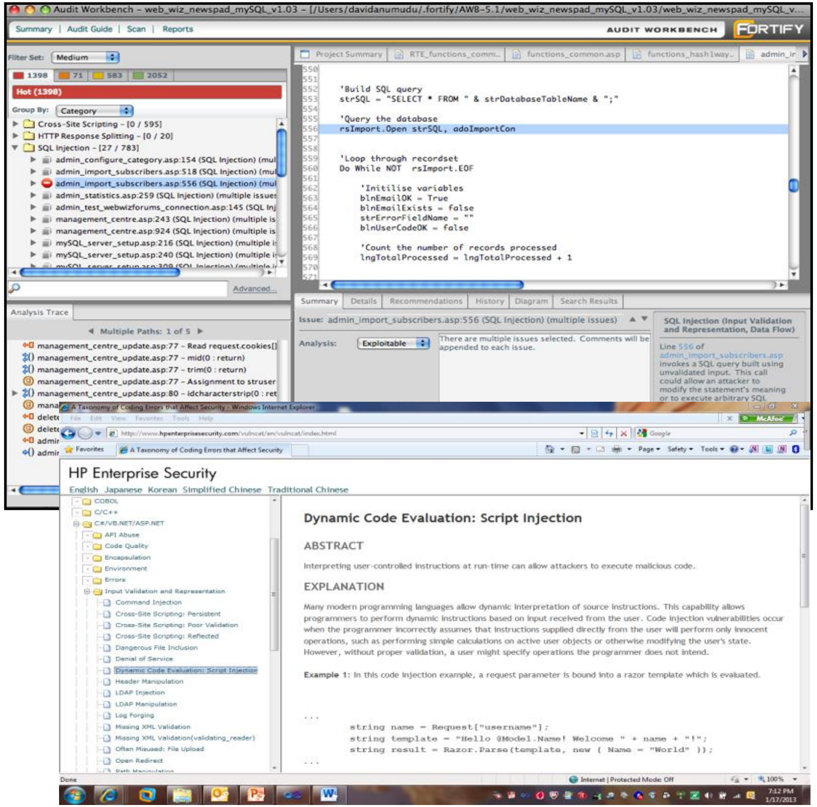
Automated static source code analysis – find and fix security issues in your code during development

Features:

- **Pinpoint the root cause of vulnerabilities with line of code** details and remediation guidance during development

- **Prioritize all application vulnerabilities** by severity and importance

- **Supports 24+ languages, 600+ vulnerability categories** including: ASP .NET, C/C++, COBOL, Flex, Java, JSP, PHP, Python, VB.NET, VBScript, C#, Classic ASP, Cold Fusion, HTML, JavaScript/AJAX, Objective C, PL/SQL, T-SQL, VB6, XML Core Ruby, Django 1.7, Jave Bytecode


Audit Workbench – web_wiz_newspad_mySQL_v1.03 – [/Users/davidanumudu/.fortify/AWB-5.1/web_wiz_newspad_mySQL_v1.03/web_wiz_newspad_mySQL_v...
Summary | Audit Guide | Scan | Reports
AUDIT WORKBENCH
FORTIFY
Filter Set: Medium
1398
71
583
2052
Hot (1398)
Group By: Category
Advanced...
Project Summary
RTE_functions_comm...
functions_common.asp
functions_hash1way...
admin_ir
Summary
Details
Recommendations
History
Diagram
Search Results
Analysis Trace
Multiple Paths: 1 of 5
A Taxonomy of Coding Errors that Affect Security - Windows Internet Explorer
File Edit View Favorites Tools Help
McAfee
http://www.hpenterprisesecurity.com/vulncat/en/vulncat/index.html
Google
Favorites
A Taxonomy of Coding Errors that Affect Security
Page Safety Tools
HP Enterprise Security
English Japanese Korean Simplified Chinese Traditional Chinese
COBOL
C/C++
C#/VB.NET/ASP.NET
API Abuse
Code Quality
Encapsulation
Environment
Errors
Input Validation and Representation
Command Injection
Cross-Site Scripting: Persistent
Cross-Site Scripting: Poor Validation
Cross-Site Scripting: Reflected
Dangerous File Inclusion
Denial of Service
Dynamic Code Evaluation: Script Injection
Header Manipulation
LDAP Injection
LDAP Manipulation
Log Forging
Missing XML Validation
Missing XML Validation(validating_reader)
Often Misused: File Upload
Open Redirect
Path Manipulation
Done
Internet | Protected Mode: Off
100%
7:12 PM
1/17/2013

Cross-Site Scripting - [0 / 595]
HTTP Response Splitting - [0 / 20]
SQL Injection - [27 / 783]
admin_configure_category.asp:154 (SQL Injection) (mul
admin_import_subscribers.asp:518 (SQL Injection) (mul
admin_import_subscribers.asp:556 (SQL Injection) (mul
admin_statistics.asp:259 (SQL Injection) (multiple issues
admin_test_webwizforums_connection.asp:145 (SQL Inj
management_centre.asp:243 (SQL Injection) (multiple is
management_centre.asp:924 (SQL Injection) (multiple is
mySQL_server_setup.asp:216 (SQL Injection) (multiple i
mySQL_server_setup.asp:240 (SQL Injection) (multiple i
mySQL_server_setup.asp:309 (SQL Injection) (multiple i

```
550
551
552      'Build SQL query
553      strSQL = "SELECT * FROM " & strDatabaseTableName & ";"
554
555      'Query the database
556      rsImport.Open strSQL, adoImportCon
557
558
559      'Loop through recordset
560      Do While NOT  rsImport.EOF
561
562          'Initilise variables
563          blnEmailOK = True
564          blnEmailExists = false
565          strErrorFieldName = ""
566          blnUserCodeOK = false
567
568          'Count the number of records processed
569          lngTotalProcessed = lngTotalProcessed + 1
570
571
```

management_centre_update.asp:77 – Read request.cookies[]
management_centre_update.asp:77 – mid(0 : return)
management_centre_update.asp:77 – trim(0 : return)
management_centre_update.asp:77 – Assignment to struser
management_centre_update.asp:80 – idcharacterstrip(0 : ret
mana
delete
delete
admir
admir

Issue: admin_import_subscribers.asp:556 (SQL Injection) (multiple issues)

Analysis: Exploitable

There are multiple issues selected. Comments will be appended to each issue.

SQL Injection (Input Validation and Representation, Data Flow)

Line 556 of admin_import_subscribers.asp invokes a SQL query built using unvalidated input. This call could allow an attacker to modify the statement's meaning or to execute arbitrary SQL

Dynamic Code Evaluation: Script Injection

ABSTRACT

Interpreting user-controlled instructions at run-time can allow attackers to execute malicious code.

EXPLANATION

Many modern programming languages allow dynamic interpretation of source instructions. This capability allows programmers to perform dynamic instructions based on input received from the user. Code injection vulnerabilities occur when the programmer incorrectly assumes that instructions supplied directly from the user will perform only innocent operations, such as performing simple calculations on active user objects or otherwise modifying the user's state. However, without proper validation, a user might specify operations the programmer does not intend.

Example 1: In this code injection example, a request parameter is bound into a razor template which is evaluated.

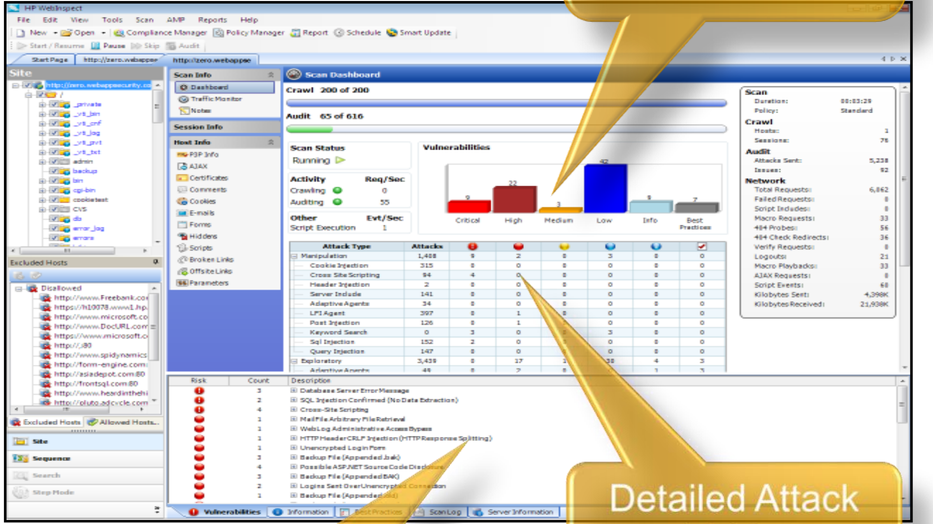
```
...
        string name = Request["username"];
        string template = "Hello @Model.Name! Welcome " + name + "!";
        string result = Razor.Parse(template, new { Name = "World" });
...
```

Dynamic Security Scanning
SAP Fortify by HPE - WebInspect

Dynamics analysis – find critical security issues in running applications

Features:

- Quickly **identify vulnerabilities in running applications**, prioritizing the most critical issues for root-cause analysis
- Automate dynamic application security testing of any technology, from development through production
- Streamline the process of remediating vulnerabilities



Crawl 200 of 200

Audit 65 of 616

Scan Status
Running

Activity	Req/Sec
Crawling ●	0
Auditing ●	55

Other	Evt/Sec
Script Execution	1

Attack Type	Attacks	❗	●	●	●	●	☑
⊟ Manipulation	1,488	9	2	0	3	0	0
Cookie Injection	315	0	0	0	0	0	0
Cross Site Scripting	94	4	0	0	0	0	0
Header Injection	2	0	0	0	0	0	0
Server Include	141	0	0	0	0	0	0
Adaptive Agents	34	0	0	0	0	0	0
LFI Agent	397	0	1	0	0	0	0
Post Injection	126	0	1	[illegible]	0	0	0
Keyword Search	0	3	0	0	3	0	0
Sql Injection	152	2	0	[illegible]	0	0	0
Query Injection	147	0	0	[illegible]	0	0	0
⊟ Exploratory	3,439	0	17	1	38	4	3
Adaptive Agents	49	0	2	0	[illegible]	1	3

Scan	
Duration:	00:03:29
Policy:	Standard
Crawl	
Hosts:	1
Sessions:	76
Audit	
Attacks Sent:	5,238
Issues:	92
Network	
Total Requests:	6,862
Failed Requests:	0
Script Includes:	0
Macro Requests:	33
404 Probes:	56
404 Check Redirects:	36
Verify Requests:	0
Logouts:	21
Macro Playbacks:	33
AJAX Requests:	0
Script Events:	68
Kilobytes Sent:	4,398K
Kilobytes Received:	21,938K

Risk	Count	Description
❗	3	⊞ Database Server Error Message
❗	2	⊞ SQL Injection Confirmed (No Data Extraction)
❗	4	⊞ Cross-Site Scripting
●	1	⊞ MailFile Arbitrary File Retrieval
●	1	⊞ WebLog Administrative Access Bypass
●	1	⊞ HTTP Header CRLF Injection (HTTP Response Splitting)
●	1	⊞ Unencrypted Login Form
●	3	⊞ Backup File (Appended .bak)
●	4	⊞ Possible ASP.NET Source Code Disclosure
●	3	⊞ Backup File (Appended BAK)
●	2	⊞ Logins Sent Over Unencrypted Connection
●	1	⊞ Backup File (Appended .old)

Live Scan Dashboard

Detailed Attack Table

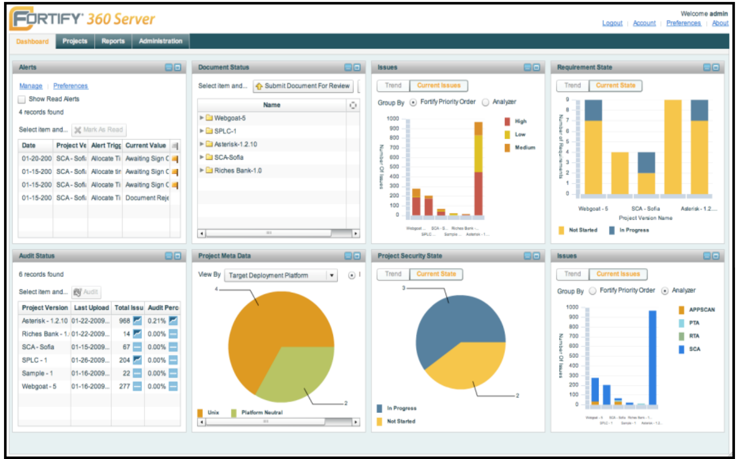
Vulnerabilities found in application

Audit, Management, Reporting Platform
SAP Fortify by HPE - Software Security Center server

Management, tracking and remediation of enterprise software risk

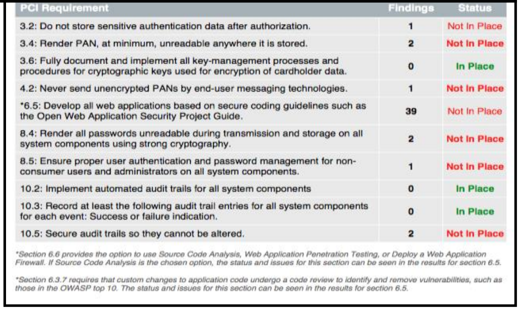
Features:

- **Flexible repository and reporting platform for security status, trending and compliance**
- Specify, communicate and track security activities on all software projects
- **Role-based, process-driven management of software security** program



Date	Project Ve	Alert Trigg	Current Value
01-20-200	SCA - Sofi	Allocate Ti	Awaiting Sign C
01-15-200	SCA - Sofi	Allocate tin	Awaiting Sign C
01-15-200	SCA - Sofi	Allocate Ti	Awaiting Sign C
01-15-200	SCA - Sofi	Allocate Ti	Document Reje

Project Version	Last Upload	Total Issu	Audit Perc
Asterisk - 1.2.10	01-22-2009...	968	0.21%
Riches Bank - 1.	01-22-2009...	14	0.00%
SCA - Sofia	01-15-2009...	67	0.00%
SPLC - 1	01-26-2009...	204	0.00%
Sample - 1	01-16-2009...	22	0.00%
Webgoat - 5	01-16-2009...	277	0.00%



PCI Requirement	Findings	Status
3.2: Do not store sensitive authentication data after authorization.	1	Not In Place
3.4: Render PAN, at minimum, unreadable anywhere it is stored.	2	Not In Place
3.6: Fully document and implement all key-management processes and procedures for cryptographic keys used for encryption of cardholder data.	0	In Place
4.2: Never send unencrypted PANs by end-user messaging technologies.	1	Not In Place
*6.5: Develop all web applications based on secure coding guidelines such as the Open Web Application Security Project Guide.	39	Not In Place
8.4: Render all passwords unreadable during transmission and storage on all system components using strong cryptography.	2	Not In Place
8.5: Ensure proper user authentication and password management for non-consumer users and administrators on all system components.	1	Not In Place
10.2: Implement automated audit trails for all system components	0	In Place
10.3: Record at least the following audit trail entries for all system components for each event: Success or failure indication.	0	In Place
10.5: Secure audit trails so they cannot be altered.	2	Not In Place

**Section 6.6 provides the option to use Source Code Analysis, Web Application Penetration Testing, or Deploy a Web Application Firewall. If Source Code Analysis is the chosen option, the status and issues for this section can be seen in the results for section 6.5.*

**Section 6.3.7 requires that custom changes to application code undergo a code review to identify and remove vulnerabilities, such as those in the OWASP top 10. The status and issues for this section can be seen in the results for section 6.5.*

Polling Question 3:

What are you doing in your organization to improve security on an application level?

1) **Penetration Testing**
2) **Focussed on perimeter defences (firewalls, encryption, virus scans etc.)**
3) **Periodic manual code reviews**
4) **Application security testing program in place and enforced**
5) **Don’t know**

Lessons Learned

- **Developers want to see only bugs, not potential issues**
- **Audit should be done by security experts**
- **Build experts usually does not care about security. Fortify is yet another tool that they must integrate**
- **Do not introduce code scanning in a „brute force“ manner**
- **Do not underestimate :**
  - **Required infrastructure**
  - **Required human resources**

SAP Runs SAP: Securing ABAP® Coding with SAP NetWeaver® AS, Add-On for Code Vulnerability Analysis

Company
SAP AG

Headquarters
Walldorf, Germany

Industry
High Tech

Products and Services
Enterprise application software and services

Employees
66,000

Web Site
www.sap.com

Objectives

- Manage more than 500 million line items of code
- Mitigate security risks for both SAP and its customers
- Strengthen SAP® applications powered by the SAP HANA® platform
- Help developers make the best security decisions while programming

The Resolution

- Step-by-step implementation of the code vulnerability analysis add-on for the SAP NetWeaver® Application Server (NetWeaver AS) component across all SAP Business Suite applications powered by SAP HANA and across all development projects on the SAP NetWeaver technology platform
- Secure programming training provided to more than 5,500 developers using the ABAP® programming language
- Automated tests on all consolidation systems built since 2011
- Availability in all development systems for new application releases

Benefits

- Scalable solution allowing regular, automated checks of the complete ABAP code base
- Efficient help for developers to avoid security coding bugs
- Secure business applications for both SAP and its customers

> "The code vulnerability analysis add-on for SAP NetWeaver AS allows developers to check coding for security bugs during all development phases. The tool is best integrated into the development environment. This enables efficient use and thus contributes significantly to secure products."

Dr. Uwe Sodan; Manager of Static Application Security Testing for Architecture Communication, Education, and Security; SAP AG

Secure

Regular, automated checks of the complete ABAP code base for transparent code security status at any time

Scalable

Implementation across SAP Business Suite powered by SAP HANA and all development projects on SAP NetWeaver, plus availability for new application releases

Streamlined

Tools and training to help developers efficiently and effectively avoid ABAP coding bugs

Enterprise Application Security Program

Program :
“As of 2012, SAP had performed static analysis on approximately **178 million lines of code using HP Fortify software**.”

HP Fortify SSC :
“HP Fortify software is important in realizing our Product Security Strategy, because it helps us **detect vulnerabilities early in the development lifecycle**. This is essential for us, because the earlier we find vulnerabilities, **the more efficiently we can repair them**.”

“The Eclipse plug-in for HP Fortify SCA is especially useful in this environment. This plug-in enables developers to perform instantaneous checks, so they can **improve the code directly in the process**.”

HP Fortify Benefit Conclusion :
“The most expensive fixing is when a bug makes it all the way into production, and a customer or an external security expert reports it back to us, we **count on HP Fortify software to help us meet stringent requirements**, protecting both our customers and our corporate brand. I can definitely say that **HP Fortify software has helped SAP in producing more secure code.**”

Summary: Your Way to Secure your Custom Code

One weakness is enough to put your business at a risk!

- **Regularly check your source code and ensure that code fits to state of the art security programming best practices**
- **Train the developers to ensure they know the common weakness**
- **Don’t expect that security is a once in a lifetime project – security improvements are part of your daily work!**

Polling Question 4:

Do you want to be contacted by an SAP or HPE representative to have a detailed discussion to take this forward?

1) **Yes**
2) **No**

Questions & Answers

Thank you

Barbara Kohde
SAP Quality Assurance Solutions
Email: b.kohde@sap.com

Shlomi Shaki
HPE - Fortify
Email: shlomi.shaki@hpe.com

Hewlett Packard
Enterprise

Discover 2016

Las Vegas June 7–9

Discover 2016 is Hewlett Packard Enterprise's must-attend global customer and partner event. Why attend?

- Explore how Hewlett Packard Enterprise is delivering IT solutions for the New Style of Business to help you go further, faster
- Network with 10,000+ attendees, including C-level executives, IT directors, engineers and HPE experts
- Find content for you, choosing from our broad array of technical and business sessions
- Explore the latest innovations from HPE in the Transformation Zone
- Find thousands of experts on hand to answer your questions and address your challenges
- Exchange ideas, information and best practices with other IT professionals and industry leaders

Register Now and receive your member discount with this

Vivit registration link:
https://www.hpe.com/events/discoverSWVivit

Thank you

- Complete the short survey and opt-in for more information from Hewlett Packard Enterprise.

www.hpe.com

www.vivit-worldwide.org