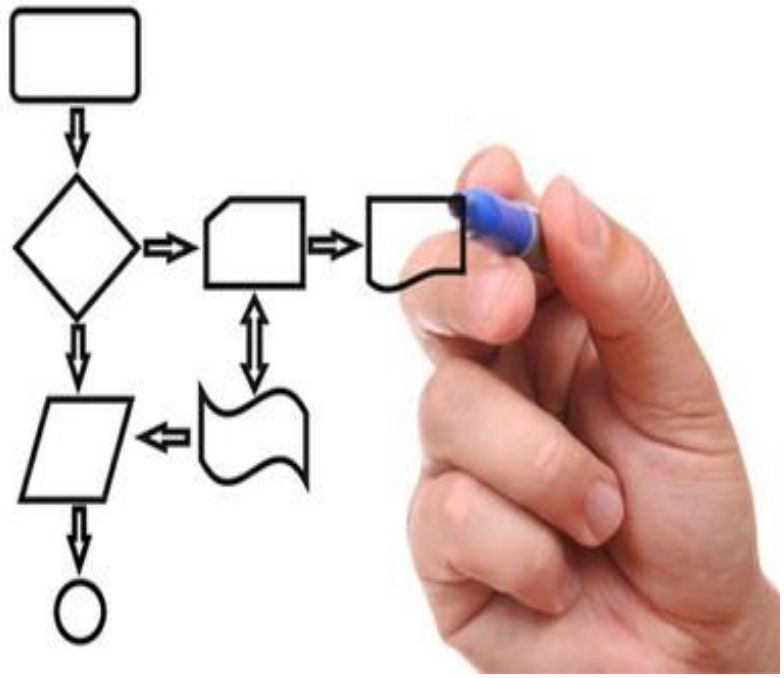
SECURE SOFTWARE ENGINEERING INITIATIVES

LISTING SSE INITIATIVES ACROSS EUROPE AND ABROAD



enisa
European Network and Information Security Agency

About ENISA

ENISA is an agency of the EU, established to contribute to a high level of network and information security within the EU. More information about ENISA and a digital copy of this report can be found on ENISA's website.

Project manager

Vangelis Stavropoulos (ENISA, Secure applications and services group)

Contractor

The list of initiatives has been compiled by Isdefe (Madrid, Spain)

External reviewer

This report has been reviewed by Yaroslav Usenko (KPMG CT, Amstelveen)

Contact details

For enquiries about this report, please email:

Vangelis Stavropoulos (vangelis.stavropoulos (at) enisa.europa.eu) or
Ulf Bergstrom, ENISA spokesperson (press (at) enisa.europa.eu)

enisa
European Network and Information Security Agency

INDEX

EXECUTIVE SUMMARY

Most high-profile cyberattacks are enabled by flaws in computer systems' software, so-called *software vulnerabilities* in the application layer. These software vulnerabilities can have major consequences:

- In March 2011 the LizaMoon attack attackers exploited so-called SQL injection vulnerabilities in an estimated 500.000 websites. The end-goal is to install a rogue antivirus product on the machines of website visitors.

- The Zeus and the Nimkey viruses exploit software vulnerabilities in desktop software. Zeus has been around since 2007 and is used by a variety of criminals to attack government and banking systems, while Nimkey was used this year to steal millions of euros from the EU carbon emissions market.

- Botnets also exploit software vulnerabilities in the application layer[1].

Many cyberattacks use a combination of social engineering and a software vulnerability exploit. The ILOVEYOU virus and the Anna Kournikova virus are well-known examples from ten years ago. A recent example is the attack on RSA in March 2011, which was a spear-phishing attack combined with a zero-day exploit of a software vulnerability.

The abundance of software vulnerabilities in the application layer of computer systems is a growing problem in cyber security and it is not immediately clear how to address it. There is a plethora of guidelines and tools for software developers to help them avoid certain software vulnerabilities, yet the same vulnerabilities continue to emerge and have an impact. As an example to illustrate this,Two well-known awareness documents, the OWASP Top ten and the CWE/SANS Top 25, list injection flaws as the top risks of recent years. There are also many guidelines and tools for website developers that help prevent injection flaws. Yet injection flaws in websites are still very common.

As a preliminary step towards addressing the problem of software vulnerabilities, we have compiled a list of existing initiatives focused on finding and preventing software vulnerabilities. In the remainder of this document we will refer to these initiatives as Secure Software Engineering (SSE) initiatives. This document provides a comprehensive list of different SSE initiatives, with a focus on the EU,but also including some major US and global SSE initiatives. The list covers 80 initiatives in the areas of:

- Requirements engineering
- Procurement criteria for secure software
- Risk-based development
- Security in agile methods
- Policy frameworks for web access control.
- Security testing methodologies and code reviewing, and
- Patch and update management

[1] ENISA recently finished and published a study on the working and impact of botnets.

We appreciate and invite suggestions (from readers) for adding missing initiatives to this list, as we may update this document in the future (see contact information).

Of particular note is that we found no government-driven SEE initiatives in the EU: In the EU, international industry-led initiatives make up the majority, while in the US there are some major government-driven SSE initiatives. For example, Build Security In is an example of such an initiative, run by the Department of Homeland Security (DHS) under the 'software assurance program'.

Our next step is to look beyond the initiatives and understand what their impact might be, or has been, in terms of addressing software vulnerabilities at the application layer, and how we can improve secure software engineering in practice. To discuss these and other ideas, we intend to organise a meeting with representatives of these initiatives in a suitable and convenient forum.

1. INTERNATIONAL SSE INITIATIVES

In this section we provide an overview of international SSE initiatives.

1.1. OPEN WEB APPLICATION SECURITY PROJECT (OWASP)

OWASP is an independent not-for-profit organisation that deals mainly with web applications security. This initiative is organised as a collaborative community, divided into local chapters in order to foster and support collaboration among its members. Almost every EU Member State (MS) has a National OWASP chapter (see section 2.2).

OWASP is not affiliated to any specific technology company. Membership categories can be individual, educational, end-user organisation, consulting organisation or vendor; each one with a different annual membership fee.

OWASP deals with application security as a people, process and technology problem because, according to OWASP, the most effective approaches to application security include improvements in all these areas.

URL	http://www.owasp.org
Contact Method	http://www.owasp.org/index.php/About_OWASP Email, mailing list, web form, phone and address
Country of HQ location **Geographic Scope**	US International
Type	Various Industry (not for profit)

The community works to produce tools and documents in three main areas:

- Protection: aimed at guarding against security-related design and implementation flaws.
- Detection: aimed at finding security-related design and implementation flaws.
- Life-cycle security: aimed at adding security-related activities into the Software Development Life Cycle (SDLC).

Its main outputs are good practice guides on the above-mentioned areas, such as the OWASP Testing Guide, OWASP Code Review or Software Assurance Maturity Model.

It also publishes Top 10 reports on risks to web applications. Another main activity area of OWASP is annual conferences in Asia, Europe and North and South America, and global AppSec summit local chapter meetings and conferences.

OWASP is trying to incorporate an activity area for training and software security knowledge, but this is currently in development. It is expected that efforts from the OWASP Exams, OWASP Academy Portal and OWASP Education projects wil be integrated in a coherent manner.

RELEVANT RESULTS

Communication Media

The OWASP AppSec conference series
Dedicated to bringing together industry, government, and security researchers and practitioners to discuss the state of the art in application security.

OWASP local chapters
Exist in most European countries (see 4,2) and are the focal point for involvement, apart from projects.

OWASP podcast
Publishes in-depth interviews with OWASP volunteers, industry experts and leaders within the field of web application security.

Video collection
Makes available video training and presentations on application security.

Good Practice

OWASP Secure Coding Practices Quick Reference Guide v2.0
[Protection Area] A technology-agnostic set of general software security coding practices, in a comprehensive checklist format, that can be integrated into the development life cycle.

OWASP Developers Guide v2.0 (2005)
[Protection Area] An extensive document covering all aspects of web application and web service security.

OWASP Code Review Guide v1.1
[Detection Area] IA guide that captures best practice for reviewing code.

OWASP Testing Guide v3.0
[Detection Area] A guide on application security testing procedures and checklists.

Standards

Application Security Verification Standard (ASVS)
[Detection Area] The ASVS defines an international standard for conducting application security assessments. It covers both automated and manual approaches for assessing (verifying) applications, using both security testing and code review techniques.

Tools

AntiSamy
[Protection Area] Java and .NET APIs validating rich HTML/CSS input from users to prevent cross-site scripting and phishing attacks.

Enterprise Security API (ESAPI) Project
A collection of free and open source security libraries that can be used by developers to build secure web applications.

JBroFuzz Project
[Detection Area] A web application fuzzer for performing testing over HTTP and/or HTTPS. Its purpose is to provide a single, portable application that offers stable web protocol fuzzing capabilities.

Live CD Project
[Detection Area] This CD collects some open source security projects in a single environment. Web developers, testers and security professionals can boot from this Live CD and have access to a full security testing suite.

WebScarab Project
[Detection Area] An intercepting proxy tool for performing all types of security testing on web applications and web services.

Zed Attack Proxy Project
[Detection Area] Another intercepting proxy tool for finding vulnerabilities in web applications. It is designed to be used by people with a wide range of security experience and, as such, is ideal for developers and functional testers who are new to penetration testing.

DirBuster Project
[Detection Area] An application designed to apply brute force to directories and file names on web/application servers.

SWF Intruder Project
[Detection Area] Is a tool for analysing and testing security of flash applications at run time.

WebGoat Project
[Life cycle security Area] An insecure web application for teaching web application security through interactive practical lessons. In the future, it is expected to become a security benchmarking platform.

OWASP Back-End Security Project
[Protection Area] This project aims to create a new guide that could allow developers, administrators and testers to understand any part of the security process of back-end components that communicate directly with web applications, databases, ldaps, payment gateways and much more. The project comprises three sections: security development, security hardening and security testing. Currently the document is not in guide format.

OWASP Backend Security OWASP O2 Platform
A highly-customisable platform for linking, managing and consuming IT security data, tools and applications.

Documentation Resources

OWASP .NET Project

[Protect Area] The purpose is to provide a central repository of information and tools for software professionals who use the Microsoft .NET Framework for web applications and services.

OWASP Top Ten Project
[Detect Area] An awareness document that describes the top ten web application security risks, and is referenced by the PCI DSS.

OWASP AppSec FAQ Project
[Life-cycle Security Area] A FAQ covering many application security topics.

OWASP Legal Project
[Life-cycle Security area] A project focused on providing contract language for acquiring secure software.

OWASP Application Security Desk Reference (ASDR)
This project is helpful as basic reference material when performing such activities as threat modelling, security architecture review, security testing, code review, and metrics. Any application security risk has a threat agent (attacker) who is using an attack to target a vulnerability (typically a missing or broken control). If successful, this attack will have both a technical and business impact.

OWASP Comprehensive, Lightweight Application Security Process (CLASP)
[Life-cycle Security Area] Provides a structured and organised approach for moving security concerns into the early stages of the SDLC, whenever possible. See section 0.

Software Assurance Maturity Model (SAMM)
An open framework for helping organisations formulate and implement a strategy for software security tailored to the specific risks facing the organisation. See section 0.

AppSensor Project
[Protect Area] Defines a conceptual framework, methodology, pilot implementations and example code that offer prescriptive guidance on how to implement attack-aware intrusion detection and automated real-time response in an existing application.

OWASP Ruby on Rails Security Guide
[Protection Area] Provides a coding and configuration guide addressing the vulnerabilities and their associated countermeasures. Ruby on Rails is an open source web application framework based on the Ruby programming language.

1.2. COMMON CRITERIA (CC)

The Common Criteria for Information Technology Security Evaluation, Common Criteria (CC), is a framework in which computer system users can specify their security, functional and assurance requirements – including software. Vendors can then implement and/or make claims about the security attributes of their products – including software – while testing laboratories can evaluate the products – including software – to determine if they actually justify the claims made. In other words, CC provides assurance that the process of specification, implementation and evaluation of a computer security product, including

software, has been conducted in a rigorous and standard manner.
http://www.owasp.org/index.php/OWASP_AppSensor_Project

The certification is issued by a national public authority, Certification Body (CB). CC evaluation of a product (hardware and software) involves several phases of the SDLC:

- Requirements: Protection Profile (PP) document identifies the security requirements of a product type independently of the implementation
- Implementation: Security Target (ST) document identifies the security properties of a specific product that implements the security requirements
- Testing: The product is evaluated against the ST.

There are seven evaluation levels, Evaluation Assurance Level (EAL), corresponding to different packages of assurance requirements:

- EAL1. Functionally tested
- EAL2. Structurally tested
- EAL3. Methodically tested and checked
- EAL4. Methodically designed, tested and reviewed
- EAL5. Semi-formally designed and tested
- EAL6. Semi-formally verified design and tested
- EAL7. Formally verified design and tested.

Countries sign the Common Criteria Recognition Agreement (CCRA) in order to rely on the CC certificates issued by any CB. This agreement applies from EAL1 to EAL4.

The following countries have signed the CCRA:

- **EU/EFTA Countries**: Austria, Czech Republic, Denmark, Finland, France, Germany, Greece, Hungary, Italy, Netherlands, Norway, Spain, Sweden and United Kingdom

- **Non-EU/EFTA Countries**: Australia, Canada, India, Israel, Japan, Republic of Korea, Malaysia, New Zealand, Pakistan, Singapore, Turkey and United States.

"European Mutual Recognition Agreement of IT Security Evaluation Certificates" or the SOGIS-agreement is an agreement between some European nations with membership of the EU or EFTA concerning mutual recognition of evaluation certificates according to the CC standards for all evaluation levels (EAL1 - EAL7).

URL	http://www.commoncriteriaportal.org/
Contact Method	http://www.commoncriteriaportal.org/contact/ Email
Geographic Scope	International
Type	Government

	Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI)
URL	http://www.ssi.gouv.fr

Contact Method	http://www.commoncriteriaportal.org/ccra/members/ Email, phone and address
Country	France

	Bundesamt für Sicherheit in der Informationstechnik
URL	http://www.bsi.bund.de
Contact Method	http://www.commoncriteriaportal.org/ccra/members/ Email, phone and address
Country	Germany

	Netherlands National Communications Security Agency (NLNCSA)
URL	
Contact Method	http://www.commoncriteriaportal.org/ccra/members/ Email, phone and address
Address	Netherlands

	The Norwegian National Security Authority operates the Norwegian Certification Authority for IT Security (SERTIT)
URL	http://www.sertit.no
Contact Method	http://www.commoncriteriaportal.org/ccra/members/ Email, phone and address
Country	Norway

	Organismo de Certificación de la Seguridad de las Tecnologías de la Información
URL	http://www.oc.ccn.cni.es
Contact Method	http://www.commoncriteriaportal.org/ccra/members/ Email and address
Country	Spain

	Swedish Certification Body for IT-Security (CSEC)
URL	www.csec.se
Contact Method	http://www.commoncriteriaportal.org/ccra/members/ Phone and address
Country	Sweden

	The Communications-Electronics Security Group (CESG) and the Department of Trade and Industry (DTI)
URL	http://www.cesg.gov.uk
Contact Method	http://www.commoncriteriaportal.org/ccra/members/ Email, phone and address
Country	United Kingdom

For instance, EU Directive 1999/93/CE for electronic signatures and EU Decision C(2003) 2439 specify requirements for secure signature-creation devices (SSCD) by CWA 14169 (PP issued by the European Committee for Standardisation – CEN). The Spanish eID card (DNIe) has been evaluated by the Spanish CB (see DNIe v1.13) according to the appropriate PP and required EAL.

Usually, new versions of products need to be certified again, so it is quite difficult for an open source community, without industry or funded support, to evaluate their products according to CC.

The Common Criteria for Information Technology Security Evaluation and the Common Methodology for Information Technology Security Evaluation had been published as ISO standards.

RELEVANT RESULTS

Standards

Common Methodology for Information Technology Security Evaluation and Common Criteria for Information Technology Security Evaluation
These form the technical basis for an international agreement (the CCRA). Version 2.3 has also been published as ISO/IEC 15408:2005 and ISO/IEC 18045:2005.

Future Related Standard

JTC 1/SC 27
ISO/IEC NP 20004
Information technology, Security techniques, Secure software development and evaluation under ISO/IEC 15408 and ISO/IEC 18405. See section **Error! Reference source not found.**.

1.3. IEEE COMPUTER SOCIETY (CS)

IEEE CS is the IEEE chapter related to IT. This initiative is a not-for-profit membership organisation and its main purposes are scientific, literary, and educational in character. The main projects of IEEE CS are aimed at publishing standards on IT technology. SSE can be found at the IEEE Computer Society's Technical Committee on Security and Privacy.

URL	http://www.computer.org
Contact Method	http://www.computer.org/portal/web/guest/contact Email, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Academic (not for profit)

The main outputs of this initiative are books, conferences, conference publications, magazines, online courses, software development certifications, standards and technical journals.

Relevant Results

Communication Media

IEEE Transactions on Software Engineering
An archival Journal, published bimonthly, interested in well-defined theoretical results and empirical studies that have a potential impact on the construction, analysis or management of software.

IEEE Software Magazine
This bimonthly magazine focuses on software development areas: requirements, design, tools, quality, open source issues and terminology. Its mission is to build a community of leading software practitioners. From time to time, there are articles about security on software development.

IEEE Security & Privacy Magazine
This bimonthly magazine covers diverse aspects of the security and dependability of computer-based systems, including legal, ethical and privacy issues. Members of the Build Security In initiative (see section 3.2) often write articles in this magazine about software security best practice. Articles usually address different fields of software security (mainly weakness and models).

International Symposium on Engineering Secure Software and Systems
See section 0.

Good Practice

Guide to the Software Engineering Body of Knowledge (SWEBOK)
The SWEBOK Version 3, alpha version, will include Security as one of the proposed Supplemental Knowledge Areas.

Standards

Software & Systems Engineering Standards Committee (S2ESC)
Formal Liaisons with ISO/IEC JTC1/SC7. See section 1.4.

1.4. INTERNATIONAL ORGANISATION FOR STANDARDISATION (ISO)

This non-governmental organisation is the one of the world's international reference standardisation bodies. It is organised mainly according to science and engineering areas, funding technical committees that deal with the standardisation activities for each technological area. National Standardisation Bodies are members of the organisation. Industry and individual experts usually take part in ISO as members of the technical committees, and they propose standards which must be approved by the ISO members.

There is one main technical committee working on IT related standards: JTC 1. Taking into account SSE, there are 3 sub-committees dealing with topics in this area:

- JTC 1/SC 7: Software and systems engineering,
- JTC 1/SC 22: Programming languages, their environments and system software interfaces, and
- JTC 1/SC 27: IT Security techniques.

ISO's main outputs are published standards that are publicly available for purchase. Related to SSE are 5 published technical reports and standards – ISO/IEC TR 15026-1:2010, ISO/IEC TR 24731-1:2007, ISO/IEC TR 24772:2010, ISO/IEC 15408 and ISO/IEC 18405 – and 2 different ongoing projects related to SSE.

URL	http://www.iso.org
Geographic Scope	International
Type	Network of national standards institutes

	JTC 1/SC 7 – Software and systems engineering
Contact Method	JTC 1/SC 7 – Secretariat Email, phone and address
Country of Secretariat	Canada

	JTC 1/SC 22 – Programming languages, their environments and system software interfaces
Contact Method	JTC 1/SC 22 – Secretariat Email, phone and address
Country of Secretariat	USA

	JTC 1/SC 27 – IT Security techniques
Contact Method	JTC 1/SC 27 – Secretariat Email, phone and address
Country of Secretariat	Germany

RELEVANT RESULTS

Published Technical Reports

JTC 1/SC 7
ISO/IEC TR 15026-1:2010 Systems and software engineering - Systems and software assurance -- Part 1: Concepts and vocabulary.

This ISO document states:"*Within software and systems assurance and closely related fields, many specialties and subspecialties share concepts but have differing vocabularies and perspectives. This part of ISO/IEC 15026 provides a unifying set of underlying concepts and an unambiguous use of terminology across these various fields*".

JTC 1/SC 22
ISO/IEC TR 24731-1:2007 Information technology - Programming languages, their environments and system software interfaces - Extensions to the C library - Part 1: Bounds-checking interfaces.

Specifies a series of extensions of the programming language C, specified by International Standard ISO/IEC 9899:1999. These extensions can be useful in the mitigation of security vulnerabilities in programs, and consist of a new predefined macro, and new functions, macros, and types declared or defined in existing standard headers.

ISO/IEC TR 24772:2010 Information technology - Programming languages - Guidance on avoiding vulnerabilities in programming languages through language selection and use.

Specifies software programming language vulnerabilities to be avoided in the development of systems where assured behaviour is required for security, safety, mission critical and business critical software. In general, this guidance is applicable to the software developed, reviewed, or maintained for any application. Vulnerabilities are described in a generic manner that is applicable to a broad range of programming languages.

It is intended to provide guidance spanning multiple programming languages, so that application developers will be better able to avoid the programming constructs that lead to vulnerabilities in software written in their chosen language and their attendant consequences. This guidance can also be used by developers to produce or select source code evaluation tools that can discover and eliminate some constructs that could lead to vulnerabilities in their software or to select a programming language that avoids anticipated problems.

Projects under Development

JTC 1/SC 7
ISO/IEC FCD 15026-2 - Systems and software engineering - Systems and software assurance -- Part 2: Assurance case.

Specifies minimum requirements for the structure and contents of an assurance case to improve the consistency and comparability of assurance cases and to facilitate stakeholder communications, engineering decisions, and other uses of assurance cases.

According to this ISO document "*An assurance case includes a top-level claim for a property of a system or product (or set of claims), systematic argumentation regarding this claim, and the evidence and explicit assumptions that underly this argumentation. Arguing through multiple levels of subordinate claims, this structured argumentation connects the top-level claim to the evidence and assumptions*".

ISO/IEC CD 15026-3 Systems and software engineering -- Systems and software assurance -- Part 3: Integrity levels. Relates integrity levels to the assurance case and includes related requirements for their use with and without an assurance case.

According to this ISO document "*A software integrity level denotes a range of values of a software property necessary to maintain system risks within tolerable limits*".

JTC 1/SC 27
ISO/IEC NP 20004
Information technology - Security techniques - Secure software development and evaluation under ISO/IEC 15408 and ISO/IEC 18405**.**

Looks into a different and urgent problem associated with practical use of the Common Criteria, namely the relationship between development and evaluation processes dealing with the analysis of potential attacks. It is related to CAPEC initiative (see section 3.5).

1.5. INTERNATIONAL SOCIETY OF AUTOMATION (ISA)

The ISA is a global, non-profit organisation that develops standards for industry, certifies industry professionals, provides education and training, publishes books and technical articles, and hosts conferences and exhibitions for automation professionals.

URL	http://www.isa.org
Contact Method	General contact Email, phone and address Standards contact Email and phone (about standards)
Country of HQ location	US
Geographic Scope	International
Type	Industry (not for profit)

ISA99 standard "Manufacturing and Control Systems Security" has some parts related to SSE. Currently, only parts "*99.01.01 - Terminology, Concepts, and Models*", "*99.02.01 - Establishing an Industrial Automation and Control Systems Security Program*" and "*99.03.01 - Security technologies for Industrial Automation and Control Systems*" are published. ISA and the International Electrotechnical Commission (IEC) negotiated the adoption of ISA 99 standards as IEC 62443 standards as well. ISA members pay a regular fee (annual or biannual), according to their type of membership, to obtain ISA benefits such as access to technical information and professional development resources.

RELEVANT RESULTS

Proposed Standards

ISA TR99.02.03 Patch Management in the IACS Environment
This technical report addresses the topic of patch management in an Industrial Automation and Control Systems (IACS) environment for asset owner and vendor communities. It is aimed at providing guidance in patch-testing and patch-management according to an acceptable level of risk.

ISA 99.03.04 Product Development Requirements
This standard will address the security requirements for product development.

Draft Standards

ISA 99.03.03 System Security Requirements and Security Assurance Levels
This standard defines security requirements that are grouped into seven categories:
1) Access control, 2) Use control, 3) Data integrity, 4) Data confidentiality, 5) Restrict data flows, 6) Timely response to an event and 7) Network resource availability. Each category includes a mapping of security requirements to security assurance levels.

1.6. SOFTWARE ASSURANCE FORUM FOR EXCELLENCE IN CODE (SAFECODE)

SAFECode is a privately-held initiative created by software developers and vendors. This initiative claims to increase the trust in information and communications technology products they sell through the introduction of software assurance methods in their products.

By identifying and promoting best practice in SSE, this initiative claims that the software industry could deliver more secure and reliable software, hardware and services. As its main outputs, this initiative delivers documents where they accumulate best practices, taking into account the software development life cycle.

URL	http://www.safecode.org
Contact Method	http://www.safecode.org/contact.php Email, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Industry (not for profit)

SAFECode states that its future direction is to:

1. Identify and share proven vendor software assurance practices
2. Promote broader adoption of such practices into the cyber ecosystem, and
3. Work with governments and critical infrastructure providers to leverage vendors' practices in order to manage enterprise risks.

RELEVANT RESULTS

Training

Security Engineering Training
A framework for corporate training programs on the principles of secure software development.

Good Practice

Software Integrity Controls
An assurance-based approach to minimizing risks in the software supply chain. Based on the practices of SAFECode members, the report provides software integrity controls for software sourcing, software development, software testing, software delivery and software resilience.

The Software Supply Chain Integrity Framework
This defines risks and responsibilities for making software secure in the global supply chain. Based on the experience of SAFECode members, it describes the software supply chain (staircase model of software suppliers) and the principles for designing software integrity controls.

Fundamental Practices for Secure Software Development
Based on the practices of SAFECode members, this outlines a set of practices for secure software development that can be applied in the different phases of the software development life cycle.

Software Assurance: An Overview of Current Industry Best Practices
This outlines the development methods and integrity controls used by SAFECode members to improve software assurance and security in the delivery.

1.7. SANS SOFTWARE SECURITY INSTITUTE (SSI)

SANS SSI provides training, certification and a library of research and community initiatives to help developers, architects, programmers and application security managers protect their software/web applications.

This initiative gathers and provides up-to-date technical information, as a free resource, on the most recent attack vectors and application security vulnerabilities, including an updated blog, weekly newsletters, webcasts, articles and documents on software security.

URL	http://www.sans-ssi.org
Contact Method	http://www.sans-ssi.org/contact.php Email and address
Country of HQ location	US
Geographic Scope	International
Type	Academic

SSI is a cooperative research and education organisation, funded by certification, students, or companies. The SSI programmes listed below are designed to teach and enable the implementation of secure coding and software development life cycle practices:

- Training for web application security and hacking, secure coding, software security testing, code review and PCI compliance
- Language-specific, secure coding training for Java/JEE, .NET, C, C#, PHP and others
- Programmer/Developer Certification (GIAC Secure Software Programmer Certification)
- Free research and news resources that are up to date with the most recent attack vectors and application vulnerabilities

SANS also publishes yearly reports on the Top 25 most dangerous programming errors (see e.g. http://www.sans.org/top25-software-errors/)

RELEVANT RESULTS

Communication Media

SANS SSI Newsletters
SANS Application Security Street Fighter Blog
Application Security Webcasts

Training

Free Application Security Mini Courses
Free online application security courses of 20-30 minutes

Security Programming Videos
The videos with "introducing" in the title are courses, the rest are mini-lessons.

Application Security Brochure
Brochure on SANS Application Security Training.

GIAC Secure Software Programmer (GSSP) Certification
See section 0.

Resources

Application Security Resources
Application security whitepapers and application security webcasts.

Security Laboratory
The "Security Laboratory" is an informal set of articles and whitepapers about security, IT and the computer security industry.

Internet Storm Center (ISC)
The ISC provides a free analysis and warning service to Internet users and organisations. Volunteers donate their time to analyse defects and anomalies, and post a daily diary of their analysis and thoughts on the Storm Center website.

Application Security Procurement Language
This is a draft software contract for buyers of custom software. Its objective is to make code developers responsible for checking the code and fixing security flaws before delivery of the software.

Top 25 Software Errors
These are listed in three categories:

- Insecure Interaction Between Components
- Risky Resource Management
- Porous Defences.

Each error includes:

- The ranking of each Top 25 entry
- Links to full Common Weakness Enumeration (CWE, see section 3.4) entry data
- Data fields for weakness prevalence and consequences
- Remediation cost
- Ease of detection
- Code examples
- Detection Methods
- Attack frequency and attacker awareness
- Related CWE entries and related patterns of attack for this weakness.

It also includes fairly extensive prevention and remediation steps that developers can take to mitigate or eliminate the weakness.

1.8. WEB APPLICATION SECURITY CONSORTIUM (WASC)

WASC produces open-source best practice for web applications. WASC states its mission as "*to develop, adopt, and advocate standards for web application security*".

URL	http://www.webappsec.org/
Contact Method	contact@webappsec.org Email
Country of HQ location	US
Geographic Scope	International
Type	Industry (non-profit)

RELEVANT RESULTS

Communication Media

Web Security Articles
The Web Security Mailing List

Resources

Web Application Security Scanner Evaluation Criteria
A set of criteria for evaluating web application security.

The Web Hacking Incidents Database
Database of web applications and related security incidents.

The Script Mapping Project
List of ways of executing script within a web page without using <script> tags.

Distributed Open Proxy Honeypots
Analysis of HTTP traffic through specially configured open proxies to categorise the requests into threat classifications.

Web Security Glossary
Index of terms and terminology relating to web applications security.

Web Security Threat Classification
An attempt to develop and promote industry-standard terminology for describing threats to the security of a website.

Web Application Firewall Evaluation Criteria
Development of detailed criteria for evaluating a web application firewall (WAF).

Web Application Security Statistics
Collection of application vulnerability statistics for identifying and mapping application security issues on enterprise websites.

1.9. INSTITUTE FOR SOFTWARE QUALITY (IFSQ)

The Institute for Software Quality, headquartered in the Netherlands, is a group of professionals involved in the development and deployment of computer software. IfSQ pursues a common goal: to raise the standard of software (and software development) around the world by promoting Code Inspection as a prerequisite to Software Testing in the production and delivery cycle.

URL	http://ifsq.nl/
Contact Method	http://ifsq.nl/contact.html
Country of HQ location	The Netherlands

Geographic Scope	International
Type	Industry (non-profit)

IfSQ analysed, quantified and augmented existing research on software quality, and distilled this into a collection of Defect Indicators: strong indications that code will be prone to error, hard to debug or costly to maintain. They have collated these indicators into a coordinated set of three standards, which are published on its website, in booklet form and in the form of courses and workshops. Most of the evaluation criteria, especially those like “major string”, “parameter not checked” and “unexpected state not trapped”, are relevant to improvements in software security.

RELEVANT RESULTS

Resources

Software Quality Standards
Levels 1, 2 and 3 are available.

1.10. MOBILE DEVICE-ORIENTED

This section groups together the mobile industry initiatives on developing secure applications for each operating system, including an initiative aimed at standardising software application development and making it independent of the operating system.

Mobile Operating Systems

The following subsections outline the principal mobile OS in the current market and related initiatives on secure development.

SYMBIAN - NOKIA

Application development can be done with standard C++.

A signed application is required from developers to perform certainn restricted functions in the device.

The following resource is available for Symbian secure development:

Symbian OS Platform Security
This is a book published by Wiley about the security architecture of Symbian OS v9. Its security architecture is relevant to developers who use Symbian OS in the creation of devices or add-on applications. It contains the following specific chapters: “*How to Write Secure Applications*”, “*How to Write Secure Servers*” and “*How to Write Secure Plug-ins*”.
The following resource was available for Symbian secure development:

Apps: Fundamentals of Symbian C++/Platform Security
An article explaining the fundamentals of programming applications in C++ for Symbian, taking into account the architectural restrictions on services and data access. 'Platform security' is the

collective name given to a group of technologies whose primary function is to control application access to data and system services.

ANDROID - GOOGLE

This Is a free, open-source OS based on the Linux kernel.

Application development is done with JAVA, using the Android Software Development Kit (SDK).

Available literature is focused mainly on the Android Security model.

Security and Permissions
Taken from an Android developers' website, this developers' guide explains the security architecture, application signatures and permissions management.

Android Security FAQ
Also taken from an Android developers' website, this is a FAQ about Android security.

Developing Secure Mobile Applications for Android
This guide goes through the Android security model, including many of the key security mechanisms and how they can be used safely.

Understanding Android's Security Framework
This is a tutorial from the Systems and Internet Infrastructure Security (SIIS) Laboratory in the Department of Computer Science and Engineering at Penn State University.

RIM - BLACKBERRY

The BlackBerry API is available for developers, but applications need to be digitally signed to perform some functions. The application development is done with Java.

Security for BlackBerry solutions is highlighted. A Security Development Guide is available from BlackBerry's website:

Security - Development Guide - BlackBerry Java SDK
This document explains the cryptographic API, how to protect the application data (content protection), the control of permitted APIs for developers, and code signing.

IOS - APPLE

An SDK (software development kit) allows third-party developers to make mobile iOS applications. But an application can only be loaded on to the devices through AppStore (property of Apple). Application development is done with Objective-C, as is usual in Mac's OS.

A Secure Coding Guide for developers is available from Apple's website:

Secure Coding Guide

This document discusses several common sources of vulnerability in programs and gives advice on how to avoid them, with a special emphasis on programs that run on the Mac OS X, Mac OS X Server and iOS operating systems.

WINDOWS PHONE 7 - MICROSOFT

This Is a proprietary OS developed by Microsoft and a successor to Windows Mobile.

The main tools used for development are Visual Studio 2010 and Expression Blend, which Microsoft offer as free downloads.

Microsoft's Developer websites include some resources for security and Windows Phone:

Security for Windows Phone
This comes from the Microsoft Developer Network (MSDN), where some Microsoft Security Development Life Cycle tools for security purposes can be found (see section 0).

Windows Phone 7 in 7: Security and Windows Phone 7
This is a Microsoft training video for solution providers.

WHOLESALE APPLICATIONS COMMUNITY (WAC)

The main goal of WAC is to develop a platform to help mobile applications developers follow the existing mobile applications standards. This is based on web technologies that enable the execution of applications on multiple mobile device platforms (a "write once, deploy everywhere" philosophy). WAC will apply standards to its platform, provided mainly by W3C, to establish "best of breed" converged solution/device APIs. These will be standardised.

WAC will allow operators to distribute applications through their respective application shop fronts and charge users via their existing phone bill. Developers will set the application price and receive a revenue share of the transaction.

URL	http://www.wacapps.net
Contact Method	http://www.wacapps.net/web/portal/contact-us Web form and address
Country of HQ location	UK
Geographic Scope	International
Type	Industry (not for profit)

BONDI, JIL and GSMA OneAPI have started to remove or reduce the fragmentation of Internet -based features on mobile devices. All of these initiatives are linked to WAC:

- WAC plans to use both the JIL and BONDI requirements, evolving these into a common specification. The long-term goal is to work collaboratively with the W3C towards a common standard based on a converged solution.

- WAC will also ensure that developers can always gain access to any network and back-end enablers that are exposed by the Operators. The GSMA One API activity will play a key role in this provision.

Mobile operators, device manufacturers, internet companies or interested parties can take part in the WAC at various membership levels. WAC has published its initial SDK to developers:

- *WAC 1.0 specification*, which includes Widget Security
- *WAC 2.0 specification*, which includes both Widget Security and Privacy, has been published as "Proposed Release Version".

Relevant Results

Communication Media

WAC analyst call
These were events held in 2010, providing information about WAC's progress and activities.

WAC Developer event
"WAC Technology" and "WAC delivery, devices and developers" are included in this programme.

Expected Good Practice

WAC will develop good practices based on BONDI, JIL and GSMA OneAPI projects. It is expected that security will be taken into account.

Expected Standards

The WAC platform will be based on standards for establishingh a set of APIs. These will be standardised. It is expected that security will be taken into account.

Specifications

Widget Security (WAC 1.0)
The WAC Security mechanism uses code signing and digital signature verification to prevent unauthorised access to handset APIs and support user protection provided by operators.
A widget using the security mechanism must operate in one of two security domains:

- Identification: A recognised source has signed the widget
- Operator: A mobile operator has signed the widget

Widget Security and Privacy (WAC 2.0)
This is a chapter of the WAC Core Specification which defines the security and privacy requirements, mainly based on BONDI, for the WAC-compliant, web-runtime-environment processing of widgets.

1.11. LIFE CYCLE AND MATURITY MODELS

This section groups the initiatives of both the software development life cycle (SDLC) and maturity models that support secure software development.

OWASP COMPREHENSIVE, LIGHTWEIGHT APPLICATION SECURITY PROCESS (CLASP)

CLASP, OWASP activity (see section 1.1) of the Life cycle security area, provides a structured approach to integrating security process elements into the early stages of the SDLC.

URL	www.owasp.org/index.php/Category:OWASP_CLASP_Project

The CLASP process is presented through five high-level perspectives called CLASP Views. These views are broken down into activities that contain process components. The CLASP views, their description and relationships are:

- Concepts View
  The interactions between CLASP process components are explained.
- Role-Based View
  The roles required by security projects are explained.
  Also applies to activities and vulnerability views.
- Activity-Assessment View
  A CLASP activities assessment is performed according to the vulnerability view.
- Activity-Implementation View
  CLASP activities, as selected in the activity-assessment view, are performed.
- Vulnerability View
  Vulnerabilities and countermeasures are identified in order to feed the activities views.

The aims of this project are to make these materials widely available, as well as to provide a forum for the community through which they can contribute material back to CLASP for everyone's benefit. CLASP version 1.2 is the latest release.

RELEVANT RESULTS

Security Process

CLASP version 1.2

MICROSOFT SECURITY DEVELOPMENT LIFECYCLE (SDL)

Microsoft SDL is a security assurance process focused on software development. It is a collection of mandatory security activities, grouped by the phases of the traditional SDLC. The Microsoft SDL process is shown in the following figure, taken from Microsoft's own website.

Figure 1: Microsoft SDL

Microsoft SDL is a methodology developed and implemented by Microsoft for projects in which security is considered a basic element of the software development life cycle. This methodology can be applied, not only to Microsoft environment-based software, but also to other environments.

Microsoft publishes this methodology by means of different guides, which cover different areas, depending on the types of application or the way software is implemented. The methodology is evolving continuously. Microsoft has recently released version 5.

URL	www.microsoft.com/security/sdl
Contact Method	support.microsoft.com/contactus/?ws=mscom#tab0 Email, chat, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Industry (Microsoft)

Combining a holistic and practical approach, the SDL introduces security and privacy throughout all phases of the development process. Its goal is to protect end-users.

Microsoft's SDL Pro network provides a means of disseminating the methodology to the software engineering community and also enables them to cooperate. The network consists of consultants, training companies and tools providers, who specialise in application security and have substantial experience of, and expertise in, the SDL methodology and technologies. They also offer their services to other companies to help implement the methodology in their processes.

RELEVANT RESULTS

Guidance

Microsoft SDL Process Guidance version 5.0

This guidance illustrates the way Microsoft applies the SDL to its products and technologies. It includes security and privacy requirements and recommendations for secure software development. It addresses SDL guidance for Waterfall and Spiral development, Agile development, web applications and Line of Business applications. IT policy makers and software development organisations can leverage this content to enhance and inform their own software security and privacy assurance programs.

Microsoft SDL for Agile Development
This documentation is not an exhaustive reference for the SDL process as practised at Microsoft, but is for illustrative purposes only.

Microsoft SDL for Line-of-Business Applications
This documentation is not an exhaustive reference on the SDL process as practised at Microsoft, but is for illustrative purposes only.

The Security Development Lifecycle
This is a book that provides guidance through each stage of the SDL, from education and design to testing and post-release. The authors are security experts from the Microsoft Security Engineering Team.

Simplified Implementation of the Microsoft SDL
This document illustrates the core concepts of the Microsoft SDL and discusses the individual security activities that need to be performed in order to claim compliance with the SDL process, including: roles and responsibilities, mandatory security activities, optional security activities and the application security verification process.

SDL Quick Security Reference (QSR)
With the SDL QSR, the SDL team introduces a series of basic guidance papers designed to address common vulnerabilities from the perspective of multiple business roles – business decision-maker, architect, developer and tester/QA.

Securing Applications
This documentation is aimed at developers of .NET Framework for writing security code. It includes: Key Security Concepts, Code Access Security, Role-Based Security, Cryptographic Services, Security Policy Management, Security Policy Best Practice, Secure Coding Guidelines and Security Tools.

Tools and Templates

Microsoft SDL Threat Modelling Tool
Threat modelling allows software architects to identify and mitigate potential security issues early, when they are relatively easy and cost-effective to resolve. It is a free tool requiring Visio 7. The tool is focused on design analysis techniques.

Microsoft SDL Process Template
A downloadable template that automatically incorporates the policy, process and tools associated with the SDL into Visual Studio's software development environment.

MSF-Agile+SDL Process Template

A downloadable template that automatically incorporates the policy, process and tools associated with the SDL for Agile development guidance, into the Microsoft Solutions Framework for Agile software development (MSF-Agile) and the Visual Studio environment.

The Microsoft SDL Tools
A map of the available free tools and templates for each SDL stage.

SOFTWARE ASSURANCE MATURITY MODEL (SAMM)

SAMM is an open framework for helping organisations formulate and implement a strategy for software security that is suited to the specific risks of a particular organisation. The OpenSAMM project, an activity of OWASP (see section 1.1), maintains and updates the SAMM documentation.

URL	www.owasp.org/index.php/Category:Software_Assurance_Maturity_Model www.opensamm.org

From the SAMM project website: "*the resources provided by SAMM will aid in:*

- *Evaluating an organisation's existing software security practices*
- *Building a balanced software security assurance program in well-defined iterations*
- *Demonstrating concrete improvements to a security assurance program*
- *Defining and measuring security-related activities throughout an organisation*"

As an open project, SAMM content is freely available for use.

The model is based on 4 software development business functions and 12 security practices (see figure below taken from the SAMM project website).

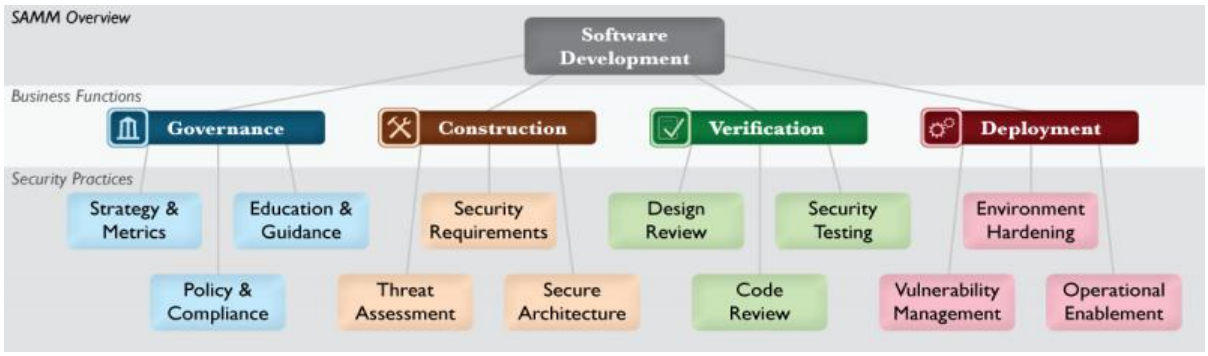


Figure 2: SAMM structure

For each security practice, three Maturity Levels are defined in terms of specific activities and metrics that an organisation could adopt in order to reduce security risks and increase software assurance.

RELEVANT RESULTS

The model is available in XML and has been translated into other languages:

http://www.opensamm.org/download/

This page also lists supporting tools.

Maturity Model

SAMM version 1.0

THE SYSTEMS SECURITY ENGINEERING CAPABILITY MATURITY MODEL (SEE-CMM)

The SSE-CMM is a process reference model. It is focused on the requirements for implementing security in a system. It promotes the integration of security engineering disciplines (e.g. systems, software and hardware).

The International Systems Security Engineering Association (ISSEA), a non-profit international membership-based organisation, is in charge of maintaining the model and its associated materials. ISSEA sponsored the SSE-CMM as an international standard: ISO/IEC 21827.

URL	http://www.sse-cmm.org
Contact Method	http://www.sse-cmm.org/contact/contact.asp Email, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Industry (not for profit)

This model has eleven security process areas where each area includes a set of base practices. These areas focus on controls, threats and the discovery and elimination of vulnerabilities:

- Administer Security Controls
- Assess Impact
- Assess Security Risk
- Assess Threat
- Assess Vulnerability
- Build Assurance Argument
- Coordinate Security
- Monitor Security Posture
- Provide Security Input
- Specify Security Needs
- Verify and Validate Security

Security best practice could be applied in software engineering.

RELEVANT RESULTS

Maturity Model

Model Description

Standard

ISO/IEC 21827

BUILDING SECURITY IN MATURITY MODEL (BSIMM)

The BSIMM is not a complete "how to" guide for software security, but a collection of ideas and activities that are in use today within software development firms.

The BSIMM was created through a process of understanding and analysing real-world data from the software security experiences of nine firms, which it then validated and adjusted with data from twenty-one additional firms. The BSIMM pulls together the experiences of thirty software development firms – most of them established in the US – who have implemented software security initiatives.

URL	http://bsimm.com/
Contact Method	http://bsimm.com/contact/ Email, mailing list and web form.
Country of HQ location	US
Geographic Scope	International (mainly the US)
Type	Industry

BSIMM has developed the Software Security Framework (SSF). SSF provides a common vocabulary for describing the most important elements of a software security framework within a firm.

Domains and practices common to most software security experiences were identified. The BSIMM describes 109 activities that any organisation can put into practice. The activities are described in terms of the SSF, which identifies twelve practices grouped into 4 domains, 3 practices by domain, as shown in the figure below, taken from the BSIMM2 document. For each practice and maturity level there is an association "one activity - one objective".

The domains are:

1. ***Governance***
   Practices that help organise, manage, and measure a software security framework. Staff development is also a central governance practice.

2. ***Intelligence***
   Collections of corporate knowledge used in carrying out software security activities throughout an organisation. Collections include both proactive security guidance and organisational threat modelling.

3. ***SSDL Touchpoints***
   Practices associated with the analysis and assurance of particular software developments, artefacts and processes. All software security methodologies include these practices.

4. ***Deployment***
   Practices that interface with traditional network security and software maintenance. Software configuration, maintenance, and other environment issues have a direct impact on software security.

The Software Security Framework (SSF)			
Governance	**Intelligence**	**SSDL Touchpoints**	**Deployment**
Strategy and Metrics	Attack Models	Architecture Analysis	Penetration Testing
Compliance and Policy	Security Features and Design	Code Review	Software Environment
Training	Standards and Requirements	Security Testing	Configuration Management and Vulnerability Management

Figure 3: BSIMM SSF

The maturity model is presented as a series of activities associated with practices. Goals for each level of practice are identified. Goals can be further split into objectives for the practice/level and are associated with activities. As an example, the following figure, taken from the BSIMM2 document, shows the maturity model for the Training practice of the Governance domain.

GOVERNANCE: TRAINING		
Objective	**Activity**	**Level**
promote culture of security throughout the organization	provide awareness training	1
ensure new hires enhance culture	include security resources in onboarding	
act as informal resource to leverage teachable moments	establish SSG office hours	
create social network tied into dev	identify satellite during training	
build capabilities beyond awareness	offer role-specific advanced curriculum (tools, technology stacks, bug parade)	2
see yourself in the problem	create/use material specific to company history	
reduce impact on training targets and delivery staff	offer on-demand individual training	
educate/strengthen social network	hold satellite training/events	
align security culture with career path	reward progression through curriculum (certification or HR)	3
spread security culture to providers	provide training for vendors or outsource workers	
market security culture as differentiator	host external software security events	
keep staff up-to-date and address turnover	require annual refresher	

Figure 4: Training practice BSIMM

In November 2009, the authors of BSIMM wrote an article in InformIT about BSIMM Europe. The differences between Europe and the US were explored because 9 firms of the BSIMM study are based in Europe. The findings of the article were as follows:

- "*The Europeans tend to carry out fewer assurance activities (for example, reviewing source code to look for bugs) and, instead, focus more energy getting a handle on the problem and meeting compliance criteria through penetration testing*".
- *"European software security initiatives put more emphasis on process than their US counterparts…. this process thinking is at least partially driven by regulatory needs". The article authors "…found plenty of emphasis on privacy in Europe".*
- *"In the case of the European firms, it may have been easier to expand existing frameworks (e.g.,* BS7799, ITIL*) to include software security governance activities, so that happened first".*
- "*…an over-focus on process may cause some of the technical activities to take a back-seat role". For example, there is less emphasis on Code Review and Security Testing in Europe".*
- *"Training is also an important practice area with less emphasis in Europe".*
- *"There are fifteen BSIMM activities that were not observed in BSIMM Europe at all"*
- *The Attack Models practice. The article's authors "…believe this reflects a general cultural reluctance in Europe to share information about attacks (that is, to restrict distribution of attack knowledge to a limited set of people on a need-to-know basis)".*
- *The Security Testing practice. The article's authors relate the above practice to this one "…where the notion of sharing information about security tests with testers cuts against the cultural grain. By contrast, the US market has embraced the*

attackers' perspective, which has come to play a critical role in US assurance and analysis regimes".

RELEVANT RESULTS

Maturity Model

BSIMM2 Describes the maturity model.

MOTOROLA SECURITY SOFTWARE DEVELOPMENT MODEL (MSSDM)

According to Motorola, in 2007 the more popular security-related models did not focus on the software life cycle and how to make it more secure. The emphasis was more on identifying and removing security vulnerabilities in the product, the environment, controls, platform and support structure around the application or product. Motorola felt there was a need for a security model focused on SDLC.

URL	http://www.motorola.com
Contact Method	General contact Email, phone and address
Country of HQ location	US
Geographic Scope	International (Motorola)
Type	Industry (Motorola)

Within Motorola Software, it was important to define a model that was closely aligned with the Capability Maturity Model Integration (CMMI) model – aimed at improving the processes of an organisation – in order to build on the success of the adoption of these models by software centres all over the world. The CMMI model was used as a reference model, the basis on which this security model was developed. The five additional process areas identified for inclusion in the security model were:

1. Secure Development Processes
2. Secure Management Processes
3. Organisation Security Focus
4. Discovery of Security Vulnerabilities and Risks
5. Corrective Security Actions.

From Motorola's point of view, it was clear that secure development practices had to be part of normal SDLC processes. Motorola attempted to incorporate these practices into an existing industry-standard, process-improvement framework like CMMI. Once these security practices are included in the CMMI, Motorola withdraws the MSSDM.

RELEVANT RESULTS

Potential Standard Contribution

- CMMI: An industry-driven initiative (MSSDM) that wants to improve existing standards (CMMI).

1.12. EVENTS AND PERIODICALS

This section groups together specialised events in secure software engineering as well as specialised periodicals.

INTERNATIONAL SYMPOSIUM ON ENGINEERING SECURE SOFTWARE AND SYSTEMS (ESSOS)

This international symposium is coordinated and organised by The DistriNET research group Katholieke Universiteit Leuven, Belgium. ESSoS'11, the third event, takes place on February 9-10, 2011, in Madrid, Spain. Its aim is to bring together researchers and practitioners so that they can advance state-of-the-art secure software engineering and good practice.

In addition to academic papers, the symposium encourages the submission of high-quality, informative papers documenting the successes and failures in security software engineering and the lessons learned.

All lectures are published by Springer in 'Lecture Notes in Computer Science'.

URL	http://distrinet.cs.kuleuven.be/events/essos2011
Contact Method	ESSoS'11 Email
Country of HQ location	Belgium
Geographic Scope	International
Type	Academic (annual symposium)

The 2011 symposium is sponsored by the Association of Computer Machinery (ACM), Special Interest Group on Software Engineering (SIGSOFT), Special Interest Group on Security, Audit and Control (SIGSAC) and IEEE Computer Society.

The list of suggested topics is:

- Scalable techniques for threat modelling and analysis of vulnerabilities
- Specification and management of security requirements and policies
- Security architecture and design for software and systems
- Model checking for security
- Specification for security artefacts
- Verification techniques for security properties
- Systematic support for security best practice
- Security testing
- Security assurance cases
- Programming paradigms, models and DLSs for security

- Program rewriting techniques
- Processes for the development of secure software and systems
- Security-oriented software reconfiguration and evolution
- Security measurement
- Automated development
- Trade-off between security and other non-functional requirements
- Support for assurance, certification and accreditation

RELEVANT RESULTS

Communication Media

The proceedings of the symposium are published by Springer-Verlag in the Lecture Notes in the Computer Science Series (e.g. ESSOS'10).

Recent Past Editions

ESSOS'09
The 18 accepted papers of ESSOS'09 were divided into five categories:

- Policy verification and enforcement
- Attack analysis and prevention
- Secure system development
- Refinement and transformation
- Testing and faults.

The papers had additional tags: 'industrial' or 'short' paper.

ESSOS'09 also included two keynotes (each keynote from one of the listed category) and two tutorials (about using a specific methodology for engineering philosophy and risk analysis).

ESSOS'10
The 18 accepted papers of ESSOS'10 were divided into three categories:

- Policy verification and enforcement
- Attack analysis and prevention
- Secure system and software development.

The reduction in the number of categories appears to be a simplification of the paper categorisation (there were papers about tests or reusability of models) in order to state the purpose of the paper. The Papers had no additional tags.

ESSOS'10 also included two parallel Workshops on:

- Security Predictions (transform climate prediction models into security models)
- Policies for the Future Internet (mainly privacy policy and trust management in SOA)

INTERNATIONAL WORKSHOP SOFTWARE ENGINEERING FOR SECURE SYSTEMS (SESS)

This international symposium is coordinated and organised by the "*Dipartimento di Informatica e Comunicazione*" of the Università di Milano, Italy. SESS'11 will take place at Waikiki, Honolulu, Hawaii, in May 2011 (at the time of writing the exact dates are yet to be decided). The 2011 symposium, run in association, with the 33rd International Conference on Software Engineering (ICSE 2011), will be the seventh such annual workshop.

It aims to provide a venue where software engineers and security researchers can exchange ideas and techniques.

Usually, approximately 50% of the conference attendees are from Europe. Accepted papers are included in ICSE proceedings. The 2011 symposium is sponsored by ICSE.

URL	http://homes.dico.unimi.it/~monga/sess11.html
Contact Method	http://homes.dico.unimi.it/~monga/sess11.html Email
Country of HQ location	Italy
Geographic Scope	International
Type	Academic (yearly workshop)

The list of areas of interest includes:

- Security requirements management
- Architecture and design of trustworthy systems
- Architecture and design of protection systems
- Separation of the security concern in complex systems
- Model-driven security
- Secure programming
- Reliability of black box components
- Security testing
- Static analysis for security
- Reliability verification and clearance
- Defining and supporting the process of building secure software
- Deployment of secure applications
- Monitoring and maintenance of the security solution
- Security usability
- Modelling and integration of dependability requirements with security constraints
- Secure software/process certification and accreditation in socio-technical environment

RELEVANT RESULTS

Communication Media

Accepted papers are included in ICSE proceedings. These are available from the ACM Digital Library and the IEEE Digital Library.

Recent Editions

SESS 2009
Was titled "*A secure software engineering*". The SESS organising committee accepted ten papers, divided into two main sections: Full papers and Position papers. As the organising committee's classification is generic, we propose a classification by topic:

Policy verification and enforcement
Its main objective is to model or specify requirements to SSE (1 full paper and 1 position paper).

Secure system and software development
Its main objective is to assess the security of the whole system (2 position papers and 3 full papers).

Attack analysis and prevention
Its main objective is to describe attack patterns or countermeasures against an attack of some kind (2 position papers and 1 full paper all of them related to XSS).

SESS 2010
Was titled "*New horizons for secure systems*". The SESS organising committee accepted 10 papers, divided into 2 main sections: Full papers and Position papers. As the organising committee's classification is generic we propose a classification by topic:

Policy verification and enforcement
Its main objective is to model or specify requirements to SSE (2 full papers and 2 position papers).

Secure system and software development
Its main objective is to assess the security of a whole system (1 position paper and 2 full papers).

Attack analysis and prevention
Its main objective is to describe attack patterns or countermeasures against an attack of some kind (2 position papers and 1 full paper).

INTERNATIONAL JOURNAL OF SECURE SOFTWARE ENGINEERING (IJSSE)

The IJSSE is an academic international journal – an official publication of the Information Resources Management Association – whose chief editor is from Qatar University (Qatar). Its Editorial Review Board is composed mainly of experts from various international universities. ENISA is a member of the Review Board (Associate Editors).

The objective is to publish original research on the security concerns that arise during the software development phase. IJSSE includes all aspects of software security in software systems' development, deployment and management processes.

The IJSSE's mission is to provide a forum where software engineers and security experts can exchange innovative ideas about security-aware software systems and address security concerns in software development practices.

Some sample articles are publicly available.

URL	http://www.igi-global.com/Bookstore/TitleDetails.aspx?TitleId=1159
Contact Method	http://www.igi-global.com/contact.aspx Email, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Academic (scholarly journal)

IJSSE is published quarterly in both printed and online editions, Its main funding comes from both individual and institutional subscriptions.

Topics discussed in this journal include (but are not limited to) the following:

- Aspect-oriented software development for secure software
- Built-in security
- Reliable systems
- Experience related to secure software systems
- Global security systems
- Maintenance and evolution of security properties
- Metrics and measurement of security properties
- Process for building secure software
- Programming security
- Relationships between security and other quality concerns
- Secure deployment of software applications
- Security artefacts, evolution, and documentation
- Security assurances, standards, and policies
- Security audit and control
- Security composition in component- and service-based software
- Security in software architecture and design
- Security literacy and education
- Security patterns
- Security requirements engineering
- Security testing and validation
- Static and dynamic analysis of security

RELEVANT RESULTS

Communication Media

Free Sample Copy
Selection of online articles.

Content of previous issues
Paper Abstracts from previous issues.

1.13. CERTIFICATION

International certification related to SSE was split into the following subsections. The types of certification selected were determined by desktop research, taking account of our own experience and knowledge.

GIAC SECURE SOFTWARE PROGRAMMER (GSSP) CERTIFICATION

The GSSP Certification Exam was a joint development, involving the SANS Institute, CERT/CC, several US government agencies and leading companies in the US, Japan, India and Germany. SANS (see section 1.7) is the certifier.

URL	http://www.sans-ssi.org/certification/

This certification focuses on the real issues behind the most common vulnerabilities and security issues in applications. The exams are technical and language-specific (Java or C#) and many of the questions use real code examples. The exams help organisations meet four objectives, which are to:

- Identify shortfalls in the security knowledge of in-house programmers and help the individuals close the gap.
- Ensure that outsourced programmers have adequate secure-coding skills.
- Appoint new employees who will not need remedial training in secure programming.
- Ensure that each major development project has at least one person with advanced secure programming skills.

After acquiring this certification, programmers will be aware of the common security flaws found in specific programming environments (JAVA or .NET), and will know how to avoid those problems that are due principally to application vulnerabilities.

The GSSP certification will be valid for four years.

INTERNATIONAL COUNCIL OF E-COMMERCE CONSULTANTS (EC-COUNCIL) CERTIFICATIONS

The EC-Council is a member-based organisation which certifies individuals in various e-business and information security skills.

URL	http://www.eccouncil.org
Contact Method	http://www.eccouncil.org/contact_us.aspx

	Email, web form, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Industry

The different types of certification offered by the EC-Council in SSE-related areas are described in the following sections.

CERTIFIED ETHICAL HACKER (CEH)

The CEH brochure states that "*The CEH Program certifies individuals in the specific network security discipline of Ethical Hacking from a vendor-neutral perspective*" and "*A CEH is a skilled professional who understands and knows how to look for the weaknesses and vulnerabilities in target systems and uses the same knowledge and tools as a malicious hacker*".

CEH has 26 modules, of which the following are related to SSE:

- Module 17: Web Application Vulnerabilities
- Module 19: SQL Injection
- Module 24: Buffer Overflows
- Module 26: Penetration Testing Methodologies

CERTIFIED SECURITY ANALYST (ECSA)

The ECSA brochure states that this certification complements the CEH certification (see above) "*by exploring the analytical phase of ethical hacking*" and "*ECSA takes it a step further by exploring how to analyse the outcome from these tools and technologies. Through ground-breaking network penetration testing methods and techniques, ECSA class helps students perform the … assessments required to … identify and mitigate risks to the security of the infrastructure*".

The objective of ECSA is to "*add value to … security professionals by helping them analyse the outcomes of their tests*".

ECSA has 47 modules, of which the following are related to SSE:

- Module 10: Advanced Exploits and Tools
- Module 11: Penetration Testing Methodologies
- Module 27: Stolen Laptop, PDAs and Cellphones Penetration Testing
- Module 28: Application Penetration Testing
- Module 40: Security Patches Penetration Testing
- Module 41: Data Leakage Penetration Testing
- Module 42: Penetration Testing Deliverables and Conclusion
- Module 43: Penetration Testing Report and Documentation Writing
- Module 44: Penetration Testing Report Analysis

- Module 45: Post-Testing Actions

CERTIFIED SECURE PROGRAMMER (ECSP)

The ECSP brochure states that "*The ECSP lays the basic foundation required by all application developers and development organisations to produce applications with greater stability and posing lesser security risks to the consumer…. The ECSP certification is intended for programmers who are responsible for designing and building secure Windows/Web-based applications with .NET/Java Framework. It is designed for developers who have C#, C++, Java, PHP, ASP, .NET and SQL development skills*".

ECSP has 33 modules, of which the following are related to SSE:

- Module 01: Introduction to Secure Coding
- Module 02: Designing Secure Architecture
- Module 03: Cryptography
- Module 04: Buffer Overflows
- Module 05: Secure C and C++ Programming
- Module 06: Secure Java and JSP Programming
- Module 07: Secure Java Script and VBScript Programming
- Module 08: Secure Microsoft.NET Programming
- Module 09: Secure PHP Programming
- Module 10: Securing Applications from Bots
- Module 11: Secure SQL Server Programming
- Module 12: SQL Rootkits
- Module 13: Secure Application Testing
- Module 14: VMware Remote Recording and Debugging
- Module 15: Writing Secure Documentation and Error Messages
- Module 16: Secure ASP Programming
- Module 17: Secure PERL Programming
- Module 18: Secure XML, Web Services and AJAX Programming
- Module 19: Secure RPC, ActiveX and DCOM Programming
- Module 20: Secure Linux Programming
- Module 21: Secure Linux Kernel Programming
- Module 22: Secure Xcode Programming
- Module 23: Secure Oracle PL/SQL Programming
- Module 24: Secure Network Programming
- Module 25: Windows Socket Programming
- Module 26: Writing Shellcodes
- Module 27: Writing Exploits
- Module 28: Programming Port Scanners and Hacking Tools
- Module 29: Secure Mobile Phone and PDA Programming
- Module 30: Secure Game Designing
- Module 31: Securing E-Commerce Applications
- Module 32: Software Activation, Piracy Blocking and Automatic Updates
- Module 33: PCI Compliance and Secure Programming

MICROSOFT CERTIFIED SYSTEMS ENGINEER (MCSE) SECURITY ON WINDOWS SERVER 2003

The MCSE certification covers skills in designing, implementing, and administering infrastructures for business solutions based on Windows Server 2003 and Microsoft Windows 2000 Server. Microsoft issues the certification. Implementation responsibilities include installing, configuring, and troubleshooting network systems.

MCSE specialisations provide more focused programmes than the MCSE certification. MCSE Security on Windows Server 2003 is the specialisation that focuses on security.

URL	http://www.microsoft.com/learning/en/us/certification/mcse.aspx
Contact Method	http://support.microsoft.com/contactus/?ws=learning#tab0 Email, chat, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Industry (Microsoft)

To qualify for the MCSE Security on Windows Server 2003 certification, eight exams – in any order – must be passed:

The following four exams on networking systems:

- Managing and Maintaining a Windows Server 2003 Environment.
- Implementing, Managing, and Maintaining a Windows Server 2003 Network Infrastructure.
- Planning and Maintaining a Windows Server 2003 Network Infrastructure.
- Planning, Implementing, and Maintaining a Windows Server 2003 Active Directory Infrastructure.

One exam on client operating systems, selected from the following:

- TS: Configuring Windows Vista Client.
- Installing, Configuring, and Administering Windows XP Professional.

One exam on design:

- Designing Security for a Windows Server 2003 Network.

Two exams on security specialisation, selected from the following:

- Implementing and Administering Security in a Windows Server 2003 Network.
- Implementing Microsoft Internet Security and Acceleration (ISA) Server 2004.
- TS: Microsoft Internet Security and Acceleration (ISA) Server 2006, Configuring.
- Third-party certifications, that could be:
    - CompTIA Security+

- Systems Security Certified Practitioner (SSCP) or Certified Information
- Systems Security Professional (CISSP) from (ISC)²
- Certified Information Security Auditor (CISA) or Certified Information Security Manager (CISM) from ISACA.

Many exams in this certification track have been withdrawn. If a required exam was passed before it was withdrawn, it can be used towards certification. The certification will not expire.

CERTIFIED SOFTWARE SECURITY LIFECYCLE PROFESSIONAL (CSSLP) AND CERTIFIED INFORMATION SYSTEMS SECURITY PROFESSIONAL (CISSP)

The CSSLP is intended to validate secure software development knowledge and best practice. The CSSLP is code-language neutral and applicable to anyone involved in the SDLC.

The certification is issued by the International Information Systems Security Certification Consortium, (ISC)², a global not-for-profit organisation specialising in educating and certifying information security professionals. It provides vendor-neutral education products.

URL	https://www.isc2.org/csslp/default.aspx
Contact Method	CSSLP Contact Web form CISSP Contact Web form General Contact Web form, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Industry (not for profit)

According to (ISC)², the CSSLP is designed to:

- Establish best practice in order to limit the proliferation of security vulnerabilities that result from insufficient development processes
- Attest to the certified professional's ability to mitigate the security concerns and risks that surround application development throughout the SDLC, from the original specification and design to implementation, maintenance and disposal

- The following domains make up the CSSLP Common Body of Knowledge (CBK) which focuses on the need for security to be built into the SDLC:

- Secure Software Concepts: security implications in software development.
- Secure Software Requirements: capturing security requirements in the requirements gathering phase

- Secure Software Design: translating security requirements into application design elements
- Secure Software Implementation/Coding: unit testing for security functionality and resilience against attack, and developing secure code and exploit mitigation
- Secure Software Testing: integrated quality assurance testing for security functionality and resilience against attack
- Software Acceptance: security implications in the software acceptance phase
- Software Deployment, Operations, Maintenance and Disposal: security issues around steady-state operations and software management.

The CSSLP qualification is valid for three years, after which it must be renewed. It can be renewed if the exam is re-taken or, as is more common, by acquiring and reporting continuing professional education (CPE) credits. CSSLPs are required to earn a minimum of 15 CPEs (of the 90 CPE certification cycle total requirements) and pay the annual maintenance fee for the three-year certification cycle.

The CISSP, another certification programme from (ISC)² with similar rules, is intended for professionals who develop policies and procedures in information security.

Certified Information Security Auditor (CISA) and Certified Information Security Manager (CISM)

The CISA and CISM from ISACA are certifications intended for IT auditors and managers to validate their knowledge in areas ranging from IT governance to protection of information assets and the development process. IT security forms a large part of these certifications, but not much emphasis is placed on Secure Software Engineering.

URL	https://www.isaca.org/
Contact Method	General Contact Web form, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Industry (not for profit)

According to ISACA, the CISA is designed to cover the following areas:

- The Process of Auditing Information Systems
- IT Governance and Management
- Information Systems Acquisition, Development and Implementation
- Information Systems Operations, Maintenance and Support.

- Protection of Information Assets and CISM is designed to cover the following areas:

- Information Security Governance
- Information Risk Management
- Information Security Program Development
- Information Security Program Management
- Incident Management and Response

The CISA and CISM certificates have to be maintained by the reporting of continuing professional education (CPE) credits. CISAs and CISMs are required to earn a minimum of 20 CPEs (of the total 120 CPE certification, 3-year cycle requirements) and pay the annual maintenance fee for the three-year certification cycle.

1.14. TRAINING COURSES

This section provides samples of international secure programming courses. The following list is not exhaustive but is intended to illustrate some of the courses available.

SECURE CODING IN C AND C++

The course is based on Addison-Wesley's material: "Secure Coding in C and C++" and "The CERT C Secure Coding Standard". This SEI (see section 3.1) training course may be offered at non-US locations.

URL	http://www.sei.cmu.edu/training/p63.cfm
Contact Method	http://www.sei.cmu.edu/training/p63.cfm Email and phone
Country of HQ location	US
Geographic Scope	International
Type	Academic (SEI)

This course provides a detailed explanation of common programming errors in C and C++ and describes how these errors can lead to code that is vulnerable to exploitation. The course concentrates on security issues intrinsic to the C and C++ programming languages and associated libraries.

Participants will acquire a working knowledge of common programming errors that lead to software vulnerabilities, how these errors can be exploited, and effective mitigation strategies for preventing the introduction of such errors. In particular, participants will learn how to:

- Improve the overall security of any C or C++ application
- Thwart buffer overflows and stack-smashing attacks that exploit insecure string manipulation logic
- Avoid vulnerabilities and security flaws resulting from the incorrect use of dynamic memory management functions
- Eliminate integer-related problems: integer overflows, sign errors, and truncation errors
- Correctly use formatted output functions without introducing format-string vulnerabilities
- Avoid I/O vulnerabilities, including race conditions

Moreover, this course encourages programmers to adopt best security practice and develop a security mindset that can help protect software.

FOUNDSTONE (MCAFEE) COURSES

Foundstone provides a network security training curriculum for developing skilled security professionals.

URL	http://www.foundstone.com
Contact Method	http://www.mcafee.com/us/about/contact-us.aspx Email, web form, phone and address
Country of HQ location	US
Geographic Scope	International
Type	Industry (McAfee)

The following subsections detail the Foundstone's computer-based training courses related to SSE.

1.14.1.1. THREAT MODELLING

This computer-based training course explains the processes and concepts of building secure software by defining a security framework and then by identifying threats and countermeasures. Students will understand how to use threat modelling to improve the SDLC.

The course has the following modules:

- Introduction to Threat Modelling and Hacme Books
- Identify Security Requirements
- Understand the System and the Application
- Identify Threats and Countermeasures
- Post-Threat Modelling Activities.

1.14.1.2. WRITING SECURE CODE - ASP.NET (C#)

This computer-based training course explains the key security features of the .NET platform, the common web security traps developers fall into and how to build secure and reliable web applications using ASP.NET. Students are led through hands-on code examples that highlight issues and prescribe solutions.

The course has the following modules:

- Introduction to Software Security
- Data Protection – in Storage and in Transit
- Authentication
- Authorisation

- Data Validation
- Process Handling
- Error Handling and Exception Management
- Logging and Auditing
- Memory Management

1.14.1.3. WRITING SECURE CODE - C++

This computer-based training course explains the key security features of the C++ language, the common security traps developers fall into and how to build secure and reliable enterprise applications using C++. Students are led through hands-on code examples that highlight issues and prescribe solutions.

The course has the following modules:

- Introduction to Software Security
- Data Protection – in Storage and in Transit
- Authentication
- Authorisation
- Data Validation
- Process Handling
- Error Handling and Exception Management
- Logging and Auditing
- Memory Management

1.14.1.4. WRITING SECURE CODE - JAVA (J2EE)

This computer-based training course explains the key security features of the J2EE platform, the common web security traps developers fall into and how to build secure and reliable web applications using Java. Students are led through hands-on code examples that highlight issues and prescribe solutions.

The course has the following modules:

- Introduction to Software Security
- Data Protection – in Storage and in Transit
- Authentication
- Authorisation
- Data Validation
- Process Handling
- Error Handling and Exception Management
- Logging and Auditing
- Memory Management

ORACLE COURSES

Oracle University is the premier provider for training for Oracle technologies and products. It offers class-based, on-site, virtual and CD-ROM courses, many of which focus on programming Java or Oracle products.

URL	http://education.oracle.com
Contact Method	Education Contact Email and phone
Country of HQ location	US
Geographic Scope	International
Type	Industry (Oracle)

The following subsections describe examples of the courses offered by Oracle worldwide. They focus on Oracle technologies and in Java programming language; some courses are given in classes, others online.

1.14.1.5. DEVELOPING SECURE JAVA WEB SERVICES, JAVA EE 6

The Developing Secure Java Web Services course provides business component and client developers with the information they need to design, implement, deploy, and maintain secure web services and web service clients, using Java technology components and the Java Platform, Enterprise Edition 6 (Java EE 6 platform).

Students learn about the need to secure web services and the challenges associated with web services security. Students also learn about prominent industry standards and initiatives developed to provide comprehensive security solutions for web services and how to apply them to secure web services. In particular, students learn how to secure web services by using application-layer security, transport-layer security and message-layer security technologies, such as those specified by the WS-* security extensions.

This course also introduces identity management concepts, drivers behind identity management solutions and Sun Java System Access Manager functions.

The objectives of the course are as follows:

- Identify the need to secure web services
- List and explain the primary elements and concepts of application security
- Outline the factors that must be considered when designing a web service security solution
- Describe the issues and concerns related to securing web service interactions
- Analyse the security requirements of web services
- Identify the security challenges and threats in a web service application
- Evaluate the tools and technologies available for securing a Java web service
- Secure web services by using application-layer security, transport-layer security and message-layer security
- Describe the concept of identity and the drivers behind identity management solutions
- Explain the role of Sun Java System Access Manager in securing web services
- Secure web services by using UserName token profile
- Secure web services by relying on Sun Java System Access Manager

The course covers the following topics:

- Encapsulating the Basics of Security
- Examining Web Services Security Threats and Countermeasures
- Securing Java Web Services Using JavaEE
- Introduction to Web Services Security
- Web Services Security with JAX-WS and Project Metro
- Authentication in JAX-WS
- Identity Management and OpenSSO

1.14.1.6. MySQL and PHP - Developing Dynamic Web Applications

The MySQL and PHP - Developing Dynamic Web Applications course explains how to develop applications in PHP and how to use MySQL efficiently for those applications. With a hands-on approach, this instructor-led course will improve the PHP skills and combine them with proven database management techniques to create best-of-breed web applications that are efficient, solid and secure.

The objectives of the course are to:

- Design web-based applications
- Design schemas based on MySQL
- Use 'include files' to make code easier to maintain
- Use PHP 5 and take advantage of its advanced features
- Build applications, following a precise flow
- Authenticate users in a secure way against a database
- Handle errors in your PHP applications efficiently and elegantly
- Write composite queries using JOINs and subqueries
- Use indexing in order to manipulate large amounts of data efficiently
- Use JOINS to extract data from multiple tables
- Use GROUP BY clauses and aggregate functions
- Write applications whose components can be scaled to meet increased demand
- Build a complete application that includes authentication and session management
- Understand how PHP, MySQL and the Apache web server work together to deliver dynamic web content

The course covers the following topics:

- PHP Foundations
- MySQL Foundations
- Manage Databases
- Manage Tables
- SQL SELECT Commands
- SQL Expressions
- SQL DML Commands
- SQL JOINS
- MySQL Database-Driven Web-Based Forms
- Session Handling

- Object-Oriented Programming
- Authentication
- Securing PHP and MySQL

GOOGLE GRUYERE

Google Code University provides a free lab environment called Gruyere, where students can try to hack web applications. Students have the opportunity to do some real penetration testing, exploiting real examples with increasing complexity. Specifically, students can learn:

- How an application can be attacked using common web security vulnerabilities, like cross-site scripting vulnerabilities (XSS) and cross-site request forgery (XSRF)

- How to find, fix, and avoid these common vulnerabilities, and other bugs that have a security impact, such as denial-of-service, information disclosure or remote code execution.

OTHER TRAINING COURSES

OWASP (see 3.1) provides free training materials, videos and presentations, and provides training opportunities at its application security conferences. It also engages with third-party education providers to develop undergraduate skills.

2. EUROPEAN SSE INITIATIVES

In this section we list European SSE initiatives. These initiatives have been categorised according to their geographic scope and type (academic, government or industry), but the structure of this section is:

Ungrouped Initiatives
Each of them constitutes an isolated category according to its objectives and results. NESSI, OWASP Local Chapters, MISRA and Serenity Forum are 'ungrouped initiatives'.

Grouped Initiatives
Each of them is grouped in the following subsections according to the objectives, results and structure under section 1:

- Events and Periodicals
- Certifications
- Academic Education.

These initiatives could be classified with multiple tags according to their relevant or expected results in SSE: standardisation, industry platform, vulnerability detection, vulnerability protection, information sharing, specialised workshop, certification and training.

2.1. NETWORKED EUROPEAN SOFTWARE AND SERVICES INITIATIVE (NESSI)

NESSI is the European Technology Platform dedicated to Software and Services. The main focus of NESSI is on strengthening Internet services through activities in research, standards and policies, and building contributions via an industry/academia community.

NESSI participants are divided into three groups:

- NESSI partners: mainly industrial but some academic profiles – who coordinate the platform and provide the financial support for NESSI's daily operations
- NESSI members: industry, academia and users – who represent major stakeholders from the ICT services provider domain. A fee is not required
- NESSI subscribers: who use different information channels to keep up to date with NESSI activities.

URL	http://www.nessi-europe.com
Contact Method	NESSI Contacts Email
Country of HQ location	Belgium
Geographic Scope	Europe
Type	Industry

Until 2010, NESSI participation was managed through NESSI Working Groups (NWGs):

- Trust, Security and Dependability NWG has hitherto dealt with Security in SOA projects, which could be related to SSE. Now, task forces will replace the NWGs and probably the task forces in the security area.
- NESSI Strategic Research Agenda (NESSI SRA) is going through a continuous update process in line with the requirements of the FP7 Work Programme.
- NESSI SRA Volume 3 - *"NESSI Roadmap"* plans the short-, mid- and long-term phases in the execution of NESSI.

National and regional technology platforms are part of the NESSI network. They handle NESSI objectives from a local point of view (see 0).

The NESSI focus from 2010 to 2015 could have some links with SSE in the following:

- Identifying future research directions in services
- Making formal contributions to key areas
- Building on the NESSI network to enhance the coordination between European, national and regional research programmes.

Relevant Results

Communication Media

Newsletters

Research Agenda

NESSI strategic research agenda (Lastest version)
One of the Research Priorities for 2009-2010 (Volume 3.2 - Revision 2 - May 2009) is "End-to-end Trust, Security, Privacy and Resilience".

Working Group related to SSE

Trust, Security and Dependability NWG
This NWG reports on the state of play regarding web services trust, security and dependability (reliability), as well as giving recommendations on future priorities, producing guidelines and identifying best practice. Task forces in security areas are expected to replace this NWG.

NESSI National Platforms

The overall aim of the NESSI Platforms is to promote the development and application of service and ICT technologies to address future challenges within European industry and government.

An open collaboration space for NESSI Platforms contributions is provided by the following website:

URL	http://nessiplatforms.ning.com

The following subsections detail the activity of each national platform that handles NESSI objectives from a local point of view and publishes its national SRA.

NESSI-NORWAY

NESSI Norway is the Norwegian branch of the NESSI. Its main objective is to establish a Norwegian arena for stakeholders in industry, research/academia and the public sector and to influence the Norwegian Government's ICT research strategy.

URL	http://www.nessi-norway.no
Contact method	From NESSI Norway Web form From NESSI Platforms Email

Norwegian SRA status is mature, but is revised each year.

The basis of this activity is that NESSI will take responsibility for the content and implementation of the EU 7th Framework Programme for R&D. They invite anyone involved in R&D activity to participate in this work.

NESSI-SLOVENIA

NESSI-Slovenia was founded in 2006. It is a forum for exchanging knowledge, developing strategies and searching for new potential in, and the faster development of, the internationally competitive IT and service industry. The central vision of the NESSI-Slovenia platform is to enable new service-oriented business models.

URL	http://www.nessi-slovenia.com
Contact method	From NESSI Slovenia Email, phone and address

The most recent version of the Slovene SRA is dated 2006.

INES-SPAIN

INES - Spanish Software and Services Platform, NESSI's Spanish Platform, was founded in 2005. It is a network of scientific and technological cooperation comprising relevant technical agents in different areas (businesses, universities, technology centres, etc.). The ultimate aim of INES is to improve the competitiveness of the Spanish ICT industry.

URL	http://www.ines.org.es
Contact method	From INES Email

INES participation is managed through several Working Groups in different sectors and technological areas. INES issues an “aligned” or “strategic” stamp of approval to support its members’ R&D projects.

The most recent version of the Spanish SRA is dated June 2010.

INES is open to any Spanish legal entity with an interest and expertise in software technologies and services.

EETP-TURKEY

EETP - Turkish National Technology Platform for Electrics and Electronics, NESSI’s Turkish Platform, was founded in 2008. EETP aims to improve Turkey’s international competitiveness, to identify priority areas for technology development and innovation and create a Strategic Action Plan and Roadmap.

URL	http://www.eetp-tr.org
Contact method	From EETP Email From NESSI Platforms Email

This platform has the following Working Groups:

- IT
- Security
- Telecommunications
- Industrial Electronics
- Consumer Electronics

IIP SAAS-NETHERLANDS

IIP SaaS is the Dutch platform for Software as a Service (SaaS), NESSI’s Dutch Platform, which brings Dutch industry together to deal and work with it. IIP SaaS works closely with the research programme Jacquard.

URL	http://www.iipsaas.nl
Contact method	From IIP-SaaS Web form and Phone From NESSI Platforms Email

The most recent version of the Dutch SRA is dated 2009.

NESSI-BULGARIA

NESSI-Bulgaria was founded in 2005. It is a forum for exchanging knowledge, developing strategies and searching for new potential in, and the faster development of, the internationally

competitive IT and service industry. The central vision of the platform is to enable new service-oriented business models.

URL	http://www-it.fmi.uni-sofia.bg/nessibg
Contact method	From NESSI Network Email

Their aims are to:

- Define a Bulgarian roadmap and SRA for the future evolution of the Bulgarian R&D and innovation programme.
- Support R&D activities in the area of software and services.
- Provide education: new courses, MSc programmes, PhD programmes and training.

NESSI-HUNGARY

NESSI-Hungary was founded in 2007 with the purpose of evolving the direction of strategic research and development in the field of software and services, based on a unified approach.

URL	http://www.nessi-hungary.com http://www.nessi.hu/
Contact method	From NESSI Hungary Email, phone and address From NESSI Platforms Email

This platform's Working Groups are split into two sub-groups: domain-oriented and technological-oriented.

The most recent version of the Hungarian SRA is dated 2009. The Platform is open to any other Hungarian organisation.

GERMANY BICC-NET

BICC-NET, NESSI's German Platform, is Germany's Bavarian ICT cluster. Founded in 2007, it wishes to selectively stimulate innovation.

BICC-NET comprises the following:

- Software development and distribution
- Hardware development and distribution
- Telecommunications
- Embedded software and hardware systems in products
- Software-based processes in development, production, services and public administration
- Services in the above-mentioned areas

URL	http://www.bicc-net.de
Contact method	From BICC-NET Web form From NESSI Platforms Email

BICC-NET is used to ensure the growth of ICT in Bavaria. It is driven by the official BICC 'cluster' office, which has been directly commissioned by the Bavarian State Ministry for Economic Affairs, Infrastructure, Transport and Technology.

BICC-NET will support Bavarian ICT companies' innovation profiles and ongoing developments.

NESSI-SWEDEN

NESSI Sweden was founded in 2010. The overall objective of NESSI Sweden is to promote the development and application of service and ICT technologies in order to address future challenges within Swedish industry and government.

URL	http://nessisweden.ning.com
Contact method	From NESSI Platforms Email

A message on NESSI Sweden website's says: "Network not available".

NESSI-ROMANIA

NESSI Romania was founded in 2010.

URL	http://sprers.eu/tech-platforms/ro-nessi
Contact method	From NESSI Romania Email, phone and address From NESSI Platforms Email

The short-term aims of NESSI-Romania are to:

- Establish national working groups on different topics defined in NESSI SRA
- Define a national SRA for the future evolution of the national R&D and innovation programme related to software and services
- Disseminate the achievements of NESSI strategic and compliant projects.

2.2. OWASP LOCAL CHAPTERS

The following subsections explain the initiatives of OWASP local Chapters in Europe. Their activities are developed from a local perspective and publicised in the corresponding entry of the OWASP Local Chapter directory.

OWASP BELGIUM LOCAL CHAPTER

URL	http://www.owasp.org/index.php/Belgium
Contact Method	Contact Leader's name and email.

The main activities carried out by this Chapter involve organising meetings, 4 during 2010, and on how to defend web applications from attacks.

OWASP DENMARK LOCAL CHAPTER

URL	http://www.owasp.org/index.php/Denmark
Contact Method	Contact Leader's name and email.

The main activities carried out by this chapter involve organising meetings, 3 during 2010, on different information security topics related to web applications. Presentations are available on its web page.

OWASP FRANCE LOCAL CHAPTER

URL	http://www.owasp.org/index.php/France
Contact Method	Contact Leader's name and email.

The main activities carried out by this Chapter involve organising meetings and translating OWASP documentation into French. This Chapter also provides training on OWASP projects and resources through the programme *"OWASP projects and resources you can use today",* which is intended to promote OWASP projects by providing a selection of mature and enterprise-ready projects, together with practical examples of how to use them.

OWASP GERMANY LOCAL CHAPTER

URL	http://www.owasp.org/index.php/Germany
Contact Method	Contact Board members and their emails.

The main activities carried out by this Chapter involve organising meetings, namely the AppSec Germany Conference, which takes place annually.

OWASP Geneva Local Chapter

URL	http://www.owasp.org/index.php/Geneva
Contact Method	Contact Leader's name and email.

The main activities carried out by this Chapter involve organising meetings related to digital identities and authentication in web applications.

OWASP Greece Local Chapter

URL	http://www.owasp.org/index.php/Greece
Contact Method	Contact Leader's name and email.

The Greek OWASP Working Group was established in 2005 with the aim of informing the Greek community about, and alerting them to, the security risks in web applications. The main reason for its creation was the ever-increasing number of security incidents on the Internet, such as phishing incidents in Greek banks.

Today, the Greek team operates OWASP projects with Free/Open software and Greek translations of OWASP, so as to promote the idea of OWASP locally. They issue a monthly newsletter, maintain a mailing list for updates and manage online debates on topical security issues.

The Greek community OWASP wants to bring together all those interested in, and concerned about, the security of web applications. At the same time, it welcomes volunteers who are willing to work on projects coordinated by the OWASP, using free/open source software. They invite anyone to share their ideas, thoughts and reflections on the attacks, defence, response methods, tools and best practice in Internet security.

OWASP Ireland Local Chapter

This country has two local Chapters represented in OWASP in separate locations, Dublin and Limerick, but the most active chapter is Dublin.

URL	http://www.owasp.org/index.php/Ireland-Dublin
Contact Method	Contact Board members and their emails.

The activities of Ireland's local Dublin Chapter involve organising events and conferences. It was particularly active in 2010, with more than 10 events during the year.

This chapter also provides training on OWASP projects and resources through the programme *"OWASP projects and resources you can use today".* This aims to promote OWASP projects by providing a selection of mature and enterprise-ready projects together with practical examples of how to use them.

URL	http://www.owasp.org/index.php/Ireland-Limerick
Contact Method	Contact Leader's name and email.

OWASP Italy Local Chapter

URL	http://www.owasp.org/index.php/Italy
Contact Method	Contact Leader's name and email.

This Chapter's activities involve event organising and tools development. The Chapter tries to arrange at least 2 conferences per annum, in the spring and autumn. Recently, they have been working on the development of sqlmap, an automatic SQL injection tool developed in Python.

The initiative is supported by partners as IsecLab, ClusIT and ISACA Rome.

OWASP Latvia Local Chapter

URL	http://www.owasp.org/index.php/Latvia
Contact Method	Contact Leader's name and email.

The OWASP Latvian Chapter was only recently created (in October 2007). This chapter's main activity is to organise events. However, despite having organised several conferences during 2008, the Chapter has not shown much strong activity in the last two years.

OWASP Leeds/Northern UK Local Chapter

URL	http://www.owasp.org/index.php/Leeds_UK
Contact Method	Contact Chapter's Leaders and email.

This is a new and very active Chapter. It has held meetings across northern England including in Leeds, Manchester and Newcastle-upon-Tyne.

OWASP London Local Chapter

URL	http://www.owasp.org/index.php/London
Contact Method	Contact Leader's name and email.

OWASP London's activities focus on preparing and organising events, conferences and presentations. The Chapter registered high activity during 2010.

It also provides training on OWASP projects and resources through the programme *"OWASP projects and resources you can use today",* which aims to promote OWASP projects by providing a selection of mature and enterprise-ready projects together with practical examples of how to use them.

OWASP Luxembourg Local Chapter

URL	http://www.owasp.org/index.php/Luxembourg
Contact Method	Contact Leader's name and email.

Luxembourg's activities involve preparing and organising events and conferences such as the Java User Group (YAJUG) or Chaos Computer Club Letzebuerg (C3L). Currently there appears to be little activity in this group.

OWASP Norway Local Chapter

URL	http://www.owasp.org/index.php/Norway
Contact Method	Contact Leader's name and email.

OWASP Norway's activities involve preparing and organising events and conferences. This chapter was highly active during past year, having organised 8 conferences in Norway in the period.

OWASP Poland Local Chapter

URL	http://www.owasp.org/index.php/Poland
Contact Method	Contact Leader's name and email.

The main activity that this chapter is to organise events. This chapter seems to be a very active one, as they were involved in 11 conferences during 2010 and the chapter's activities appear to ber continuing into this year. The initiative is supported by ISSA.

OWASP Portugal Local Chapter

URL	http://www.owasp.org/index.php/Portuguese
Contact Method	Contact Leader's name and email.

This Chapter's activities involve organising conferences and publications. In the past year it has organised one of OWASP's major events: the Ibero-American Web Application Security Conference IBWAS'2010. Refer to the Chapter's web page for presentations, accepted papers and videos of conferences and events.

OWASP Scotland Local Chapter

URL	http://www.owasp.org/index.php/Scotland
Contact Method	Contact Leader's name and email.

The main activities carried out by this Chapter, according to its web page, involve providing responses jointly with other local British chapters to several UK Government offices. This chapter also appears to organise annual meetings.

OWASP Slovakia Local Chapter

URL	http://www.owasp.org/index.php/Slovakia
Contact Method	Contact Leader's name and email.

The main activity of this Chapter is to organise events. The Chapter appears to have increased its activity during the past year, organising two events in the period.

OWASP Slovenia Local Chapter

URL	http://www.owasp.org/index.php/Slovenia
Contact Method	Contact Leader's name and email.

This Chapter's main activity is organising events. According to the information provided on its web page, it usually organises 2 conferences per annum. It also provides slides of the presentations on its web page.

OWASP Spain Local Chapter

URL	http://www.owasp.org/index.php/Spain
Contact Method	Contact Leader's name and email.

This Chapter carries out two main activities. On the one hand it is actively collaborating with OWASP on a project to provide Specifications on Legal Requirements for Web Applications. On the other, like most of the other local chapters in this section, it organises annual events and conferences and also participates in the IBWAS'2010 event in collaboration with the Portuguese Chapter.

OWASP Sweden Local Chapter

URL	http://www.owasp.org/index.php/Sweden
Contact Method	Contact Leader's name and email.

This Chapter focuses on organising meetings and events. In the past year it has organised four conferences, one of them in cooperation with other northern Chapters, such as the Norwegian and Finnish.

OWASP Switzerland Local Chapter

URL	http://www.owasp.org/index.php/Switzerland
Contact Method	Contact Leader's name and email.

This chapter organises meetings on a regular basis, mainly in the German-speaking part of Switzerland. On its web page the Chapter advises French-speaking Swiss to contact OWASP's Geneva Chapter. Their meetings and events are mainly on topics like security testing, secure development, hacking and secure architectures. Visitors to its web page will find slides of events and conferences.

OWASP Ukraine Local Chapter

URL	http://www.owasp.org/index.php/Ukraine
Contact Method	Contact Leader's name and email.

As far as one can tell from its web page, this is a recently-formed Chapter. It is currently enlisting members. No activities or events are, or have been, planned.

2.3. MOTOR INDUSTRY SOFTWARE RELIABILITY ASSOCIATION (MISRA)

MISRA is a Motor Companies Consortium within the UK. Its research, research results and de facto standards and guidelines are aimed mainly at safe, reliable software for embedded systems in the motor industry.

In the early 1990s, the MISRA project was conceived in order to develop guidelines for the creation of embedded software in road vehicles' electronic systems. After official funding ceased, MISRA members decided to continue working together.

MISRA is a collaboration between vehicle manufacturers, component suppliers and engineering consultancies. It seeks to promote best practice in the development of safety-related electronic systems in road vehicles and other embedded systems.

Its documentation is not publicly accessible but can be bought on the consortium's web page.

URL	http://www.misra.org.uk
Contact Method	MISRA Contact Email, phone and address
Country	UK
Geographic Scope	National
Type	Industry

MISRA's current work-in-progress includes:

Model based development and autocode
Encourages good modelling practices and avoids poorly-defined features of the modelling language.

- MISRA C++

Production of a set of guidelines for the use of C++ in critical systems.

- MISRA C3 (3rd review of MISRA C)
- Seeks to promote best practice in developing safety-related electronic systems in road vehicles and other embedded systems (It has been adopted and used across a wide variety of industries and applications including the rail, aerospace, military and medical sectors)

MISRA Safety Analysis
Guidance on requirements for decomposition. It describes how the safety life cycle of automotive systems fits into a vehicle'stotal development life cycle.

RELEVANT RESULTS

Communication Media

The MISRA Bulletin Board

(Forum format, registration required to read the posts)
Exists to serve anyone who is interested in, or is a user of, MISRA publications.

Publications
Can be bought on the consortium's web page.

Good Practice

- Guidelines for the Use of the C Language in Vehicle Based Software, ISBN 978-0-9524156-6-5, April 1998, October 2002.
- Guidelines for the Use of the C Language in Critical Systems, ISBN 0 9524156 2 3 (paperback), ISBN 0 9524156 4 X (PDF), October 2004.
- Guidelines for safety analysis of vehicle based programmable systems, ISBN 978-0-9524156-5-7 (paperback), ISBN 978-0-9524156-7-1 (PDF), November 2007.
- Guidelines for the Use of the C++ Language in Critical Systems, ISBN 978-906400-03-3 (paperback), ISBN 978-906400-04-0 (PDF), June 2008.

Standards

- MISRA AC GMG: Generic modelling design and style guidelines, ISBN 978-906400-06-4 (PDF), May 2009.

2.4. EUROPEAN SPACE AGENCY (ESA)

Since the early 1990s ESA has been busy defining software quality products for its own, internally-developed software, as well as for externally-sourced software components. The PSS family of standards (later replaced by ECSS standards) includes a software engineering standard and a set of guides.

URL	http://www.esa.int
Contact Method	ESA Contact Email, phone and address
Headquaters	Paris
Geographic Scope	Europe
Type	Collaboration of Several European Countries

One of the widely-used software standards in that series, called 'Guide to applying the ESA Software Engineering Standards to small software projects' is available at ftp://ftp.estec.esa.nl/pub/wm/wme/bssc/Bssc962.pdf

This standard defines a number of quality criteria for software requirements and design, which have a direct and indirect influence on software security. For the quality criteria requirements the following are relevant:

- Are the characteristics of users and of typical usage mentioned? (No user categories missing)
- Are all the external interfaces of the software explicitly mentioned? (No interfaces missing)
- Is each requirement prioritised? (Is the meaning of the priority levels clear?)

- Is each requirement verifiable (in a provisional acceptance test)? (Measurable: where possible, quantify; capacity, performance, accuracy)
- Are the requirements consistent? (Non-conflicting)
- Are the requirements sufficiently precise and unambiguous? (Which interfaces are involved, who has the initiative, who supplies what data, no passive voice.)
- Are the requirements complete? Can everything not explicitly constrained indeed be viewed as developer freedom? Is a product that satisfies every requirement indeed acceptable? (No requirements missing)
- Are the requirements understandable to those who will need to work with them later?
- Are the requirements realisable within budget?
- Most of the design quality criteria are relevant to software security.

Relevant Results

Good Practice

- The PSS family of standards for Software Quality.
- A guide to applying ESA Software Engineering Standards to small software projects is available at ftp://ftp.estec.esa.nl/pub/wm/wme/bssc/Bssc962.pdf
- Eindhoven University of Technology provides further simplified requirements and design checklists.

2.5. Serenity Forum

This forum has been created by Serenity project partners (a funded R&D FP6 project until June 2009) to continue the community established during the project. Serenity day event, June 2009, was promoted by this forum and had an agenda featuring lots of of secure software engineering topics. There appears to be little activity arising from that event.

URL	www.serenity-forum.com
Contact Method	Not available
Country of HQ location	
Geographic Scope	European
Type	Academic

SERENITY Forum is charged with providing a radically new approach to Security Engineering through a wide set of security patterns and integration schemes. It is made up of Serenity Project Members and individuals.

Relevant Results

Communication Media

Serenity day event

2.6. EVENTS AND PERIODICALS

One European event and one online magazine were found, but they were grouped in the Events and Periodicals section in order to maintain the established structure in section 1.

INTERNATIONAL WORKSHOP ON SECURE SE (SECSE)

This is an annual workshop organised by SINTEF ICT (the largest independent research organisation in Scandinavia). It focuses on techniques, experiences and lessons learned from secure engineering and reliable software.

Membership of the Programme Committee comprises 3/4 from Europe and 1/4 from North America and Asia. ENISA is on the Programme Committee.

This international workshop takes place in conjunction with the "Availability, Reliability and Security" ARES Conference, funded by ARES sponsors and attendees. The next one, the fifth, will be from 22 to 26 August, 2011 in Vienna, Austria.

Even though it is an international workshop, this initiative is considered to be a European one, because of the majority European representation on the Programme Committee and in the number of accepted papers (90% in 2009 and 70 % in 2010).

All papers accepted will be published as ISBN proceedings by the IEEE Computer Society.

URL	http://www.sintef.org/secse
Contact Method	http://www.sintef.org/secse Email
Country of HQ location	Norway
Geographic Scope	European
Type	Academic (workshop)

This workshop focuses on the techniques, experiences and lessons learned from the engineering of secure, reliable software in order to build better, more robust and more secure systems. Even more importantly, however, these standards need to be achieved for all software systems, not just the ones that need special protection.

The suggested topics for the next edition are:

- Secure architecture and design
- Security in agile software development
- Aspect-oriented software development for secure software
- Security requirements
- Risk management in software projects
- Secure implementation
- Secure deployment

- Testing for security
- Quantitative measurement of security properties
- Static and dynamic analysis for security
- Verification and assurance techniques for security properties
- Lessons learned
- Security and usability
- Teaching secure software development
- Experience reports on the successful introduction to developers of secure software engineering. .

RELEVANT RESULTS

Communication Media

All accepted papers will be published as ISBN proceedings by the IEEE Computer Society and will be available online through IEEE Xplore (e.g. IEEE Availability, Reliability, and Security, 2010. ARES '10 International Conference.)

Recent Editions

SecSE2009
The 10 papers from SecSE2009 that were accepted (an acceptance ratio of 66%), were organised in the following categories:

- Education and other Vulnerabilities – the papers in this section dealt with technical countermeasures related to software development, such as static code analysis. The aim of these papers was to educate software developers to improve security implementation.

Secure Software Development Life Cycles and Re-use – the papers in this section dealt with methodologies and tools to improve security in the software engineering life cycle.

Model-driven Development and Checklists – the papers in this section dealt with models related to general mechanisms for detecting code-based vulnerabilities, the design and implementation of secure applications (based on cryptography), privacy by design, and software inspections based on checklists.

- SecSE2010

The 10 papers from SecSE2010 that were accepted (an acceptance ratio of 56%) were classified in three main categories (different from the ones in the 2009 edition):

Agile development and hot patching – the papers in this section are related to agile development, proposing a methodology for web-app, and presenting the results of a SOA application security assessment. Related to hot patching, one paper presented a framework designed for these purposes.

Testing, monitoring and validation – in this section the selected papers deal with vulnerability detection and monitoring in different platforms. One of the papers is

related to input validation. Another introduces a new technique called configuration fuzzing to detect buffer overflows. A third paper deals with buffer overflow classification.

Security modelling and vulnerabilities – the papers in this section deal with the modelling and checking of the security-related elements throughout the entire software development life cycle.

Security Acts

Security Acts is a periodical magazine aimed at IT professionals who are working, or have a professional interest, in IT security. It is published in Germany by the consultancy Díaz & Hilterscheid Unternehmensberatung GmbH. The magazine usually publishes articles related to SSE, but it has not had a special section related to this topic. Four editions were published from October 2009 to August 2010.

URL	www.securityacts.com
Contact Method	http://www.securityacts.com/contact.html Email, phone and address
Country of HQ location	Germany
Geographic Scope	Europe
Type	Industry (Magazine)

Participation is open to all professionals, from the security manager to the penetration tester and ethical hacker. They are all expected to publish their know-how and daily work experiences in this magazine.

Anyone can subscribe to it free of charge and download an online version. It is funded by advertisements.

Relevant Results

Communication Media

Online Magazines.

2.7. Certifications

Only one European certification initiative was found, but it has been included in a Certifications section to maintain the structure established in section 1.

International Secure Software Engineering Council (ISSECO)

ISSECO promotes training courses on SSE to software engineers so that they can obtain a Certification standard (ISSECO Certified Professional for Secure Software Engineering). The certification is provided by the International Software Quality Institute (iSQI).

According to this initiative, "*ISSECO's focus is on the production of such secure software and its goal is to establish a secure computing environment for all*". It is not focused on specific programming languages.

Membership of ISSECO can only be by individuals, most of them either University Professors or consultants. A condition of membership is that the applicant has no direct business interest.

URL	www.isseco.org
Contact Method	ISSECO Contact Email iSQI Contact Email, phone and address
Country of HQ location	Germany
Geographic Scope	National
Type	Industry (not for profit)

The main topics of the certification include:

- Viewpoints of attackers and customers
- Trust and threat models
- Methodologies
- Requirements engineering with respect to security
- Secure design
- Secure coding
- Security testing
- Secure deployment
- Security response
- Security metrics
- Code and resource protection

The activities of this initiative are supported by different partners:

- Supporters (financial aid)
- Training providers (training material and classes)
- Certifiers (certification and certificate quality)

Discussions are in progress on publishing ISSECO course material under the OWASP label. This might prompt a change in the business case.

2.8. ACADEMIC EDUCATION

Some universities in Europe offer Bachelor of Science (BS/BSc) or Master of Science (MS/MSc) qualifications in Security Computer Engineering or similar, where the security

component of software engineering is highlighted. Some universities offer some courses related to SSE.

The following list is not exhaustive but is intended to illustrate the range of courses available. European universities were chosen following desktop research that looked at:

- Specialised websites of postgraduate courses
- University papers presented at specialised workshops or symposiums

The following subsections detail some qualifications and courses available from European universities, whose descriptions and programmes were taken from the corresponding website.

Birmingham City University - Computer Networks and Security BSc

URL	www.bcu.ac.uk

This BSc teaches the ability to design, implement and evaluate identification systems, data capture systems and communications networks and their associated security protocols within a business environment.

The main objective of the course is for students to acquire knowledge about how to develop a secure network and communication systems to combat fraud and malicious attacks, together with the requirements for handling errors and misfortune.

Dublin City University - M.Sc. in Security and Forensic Computing

URL	http://www.dcu.ie

This MSc addresses issues related to the practical examination of computer crime and the principles underlying its prevention. It adopts a holistic approach to the study of forensic computing and provides an understanding of the legal, technical, information-management and ethical issues that impact the discipline.

Included in the course is a final practical project, which involves developing a secure software system prototype that solves a real-world problem. The projects typically require the preparation of a feasibility study, followed by the creation of a project plan and the development of a software application or piece of theoretical analysis.

University of Oxford - Design for Security (DES)

URL	http://www.softeng.ox.ac.uk

This course teaches the ability to develop systems that fulfil security goals. It shows students how to achieve cost-effective solutions to security needs by working with well established architectural and detailed security principles. The students must always meet requirements with established solutions, striking a balance between security and other system requirements. The main objectives are to understand the strengths and weaknesses of different security design techniques and to be able to specify security solutions to meet specific design requirements.

The course contents include:

Managing Security
Enterprise business strategies; Promoting security; Information security policy;

Security Requirements
Motivation for security requirements; Security requirements artefacts; Specifying security requirements;

Security Design
Process Business continuity; Principles of security design; AEGIS design methodology;

Security Architectures
Security design patterns; Platform and channel security components; Enterprise security architectures;

Designing Access
Control Security and access control; Access control policy; Security policy models;

Designing Secure Systems
Security standards; Security decision-making; Design principles; Architecture principles; Security vs. other architectural goals.

UNIVERSITY OF OXFORD - SECURE AND ROBUST PROGRAMMING (SRO)

URL	http://www.softeng.ox.ac.uk

This course deals with how to handle problems that can cause failures and security vulnerabilities from the perspective of programming. Some of these problems include: inadequate handling of exceptional situations, poor understanding of the details of the programming language in use, incomplete descriptions of the interfaces between components, and insufficient care in the treatment of concurrency and threading issues.

Course contents include:

Motivation
Explores the causes behind some well known software errors and provides examples. Defines terms used throughout the course (e.g. robustness, correctness, defensive programming)

Static Semantics
Introduces types; type checking for core programming languages, modules and objects; ownership issues and generics. Also includes a discussion on units checking. Standard type checkers will be used to demonstrate a wide variety of robustness concerns.

Dynamic Semantics
Introduces the main concepts in modelling run-time behaviour; provides fragments of semantic definitions, including objects. Describes "managed code" and uses relevant software to analyse source programs. Discusses thread safety.

Design by Contract
Introduces notions of program correctness and refinement. Uses JML or equivalent to demonstrate model checking of example programs.

Robustness in Context
Summarises the techniques and tools used during the course. Justifies coding standards and security metrics in terms of what has been learnt.

UNIVERSITY OF GLAMORGAN - MSC COMPUTER SYSTEMS SECURITY

URL	http://courses.glam.ac.uk

This specialist postgraduate computing degree focuses on the technical aspects of computer systems security and systems administration, particularly penetration testing. It provides in-depth knowledge of security issues at a technical and managerial level.

The modules include:

- Project Management and Research Methodology
- Security Management
- Network Security
- Practical Unix Security
- Practical Windows Security
- Independent Study in Computing
- Wireless Security
- Project: the development and evaluation of an application or subject area of your choice relating to Computer Systems Security

UNIVERSITY OF LIVERPOOL- MSC IN COMPUTER SECURITY

URL	http://uol.ohecampus.com

This MSc in Computer Security teaches different disciplines, such as cryptography, forensics, network design and Internet programming, taking account of the legal considerations that influence security policy.

In addition to the mandatory or core modules, students can customise their degree by choosing two optional modules according to their individual requirements. At the end of the course, students complete an original dissertation.

The modules of this MSc include:

- **Core modules**
  - Computer structures
  - Professional issues in computing
  - Computer communications and networks
  - Security engineering
  - Computer forensics
  - Programming the internet

- **Elective modules**
  - Databases
  - Software engineering
  - Management of quality assurance and software testing
  - Object-oriented programming in Java
  - Web XML applications
  - E-commerce

UNIVERSITY OF HERTFORDSHIRE- SECURE COMPUTING SYSTEMS MSC, PGD, PGC

URL	http://www.herts.ac.uk/

This MSc explores computer systems security. Advanced topics studied include cryptography, security protocols and the relative strengths and weaknesses of programming languages and software features. The course teaches awareness of security requirements, services, threats and counter-measures, and develops the skills required to identify and evaluate the methods employed by those wishing to break into insecure systems. Students are taught how to write more secure code and also learn about network systems administration.

It includes core and optional modules. Two or more optional ones are selected by students:

- **Core modules**
  - Distributed Systems Security
  - Network System Administration
  - Secure Computing Systems (MSc Project)
  - Secure Systems Programming

- **Optional modules**
  - Advanced Database
  - Human Computer Interaction: Principles and Practice
  - Measures and Models for Software Engineering
  - Mobile Standards, Interfaces and Applications
  - Multimedia Specification, Design and Production
  - Software Engineering Practice and Experience
  - Web Services

UNIVERSITY OF LONDON - MSC IN INFORMATION SECURITY

URL	http://www.rhul.ac.uk

This MSc in Information Security provides instruction in cryptography, computer security, network security, digital forensics and fraud detection, as well as considering the management of security and the many trade-offs and subjective issues that need to be addressed when implementing information security within an organisation. The course objective is that students understand the current threats to the security of electronic information and the measures available to counteract them.

The MSc can be studied via two distinct pathways:

- **Technical Pathway**
  This Pathway places considerably more emphasis on Computer and Network Security and includes the following set of prescribed (core) modules:
    - Security Management
    - Introduction to Cryptography and Security Mechanisms
    - Network Security
    - Computer Security (Operating Systems)

- **Secure Digital Business Pathway**
  This Pathway focuses on security infrastructures and legal aspects and includes the following set of prescribed (core) modules:
    - Security Management
    - Introduction to Cryptography and Security Mechanisms
    - Security Technologies
    - Legal and Regulatory Aspects of Electronic Commerce

- **Options for both pathways**
  Both pathways of this MSc include optional modules from which two or three are selected by students:
    - Regulatory Aspects of Electronic Commerce (Technical Pathway option only)
    - Application and Business
    - Security Developments
    - Standards and Evaluation Criteria
    - Database Security
    - Computer Crime
    - Smart Cards/Tokens Security & Applications
    - Software Security
    - Digital Forensics
    - Security Testing Theory and Practice

At the end of the course, each student will complete a project.

BRITISH INSTITUTE OF TECHNOLOGY AND E-COMMERCE - MSC SECURITY TECHNOLOGY

URL	http://bite.ac.uk/

This MSc Security Technology focuses on topics such as security systems, security science, cyber security and communication security addressing the problem areas of security.

The 4 modules of this MSc are:

- **Security Systems**

This module covers Fundamentals of Security System, Network Security, Information Security and Network Intrusion.

- **Security Science**
  This module covers Fundamentals of Security Science, Artificial Intelligence, Stream Cipher and Security Intelligence.

- **Cyber Security**
  This module covers Fundamentals of Cyber Security, Computer Forensic, Cyber GIS and Cyber Forensic.

- **Communication Security**
  This module covers Fundamentals of Communication Security, Mobile Communication Security, Mobile Communication Surveillance and Global Positioning System.

TELECOM BRETAGNE - SOFTWARE DEVELOPMENT AND SECURITY MSC

URL	http://www.telecom-bretagne.eu

This MSc studies the concept of the security of information systems as a set of methods, techniques and tools to protect the resources of an information system to ensure service availability, confidentiality and integrity of information systems. The *Sécurité des systèmes d'information*" MSc provides the ability to understand, in all its dimensions, issues related to the security of information systems.

Advanced topics studied include the following roles: security manager of information systems; secure applications designer/evaluator; expert in information systems security.

UNIVERSITÉ DU LUXEMBURG - MASTER IN INFORMATION AND COMPUTER SCIENCES

URL	http://wwwfr.uni.lu

This Master is divided into two stages; the first phase is compulsory and covers the fundamentals of computer science. In the second phase, the student selects courses based on one or more optional profiles. Profiles are similar to specialisations with the added benefit that multiple profiles can be built up. Each profile has a set of required courses and a set of related (optional) courses associated with it. The subjects included in the "information security" profile are listed below.

The available profiles are:

- Adaptive Computing
- Communication Systems
- Information Security:
  - Principles and Applications of Information Security
  - Algorithms for Numbers and Public-Key Cryptography
  - Symmetric-Key Cryptography

- Security Protocols
- Security Modelling
- Management of Information Security
- Cryptography in the Real World
- Open Network Security
- Intelligent Systems
- Network Systems
- Reliable Software Systems

The learning outcomes of the "information security" profile equip students to:

- Understand the principles and techniques of information security
- Derive new applications based on information security
- Understand the most important standards related to the management of information security and apply them in a given context
- Engage in life-long learning about new developments in information security.

ICAL UNIVERSITY OF DENMARK - MSC IN COMPUTER SCIENCE AND ENGINEERING

URL	http://www.dtu.dk

This MSc focuses on the design and use of computing components, software or hardware, to solve technical problems in an efficient and competitive way. The course provides the ability to model, analyse, design, implement and validate complex IT systems based on theoretically and technologically well-founded methods, tools and techniques. The main topics studied are computer science, computer engineering, mathematics, logic, systems engineering and project management.

The programme covers the following subject areas:

- Software Development
- Safe and Secure IT Systems
- Algorithms, Logic and Knowledge-Based Systems
- Embedded Systems

The programme has the following modules:

- Digital Systems
- Efficient and Intelligent Software
- Embedded and Distributed Systems
- Software Engineering
- Reliable Software Systems

IT systems increasingly shape the infrastructure of society. These must be designed from the outset, taking account of performance, safety and security concerns. The development phase requires state-of-the-art methods and techniques for analysing and checking software to ensure the absence of abnormal behaviour. This study module explores methods and techniques for the development of reliable software systems.

AALTO UNIVERSITY SCHOOL OF SCIENCE AND TECHNOLOGY - MASTER'S PROGRAMME IN MOBILE COMPUTING - SERVICES AND SECURITY

URL	http://information.tkk.fi

This Master explores the areas of network applications, services, information security and mobile networking. Special attention is paid to information security, which is a critical issue when developing and deploying services on public networks.

The students acquire knowledge of the fundamental technologies and design principles of communication networks, the Internet and mobile systems, including applications, services and service management, in order to develop new communication network solutions, service platforms and service architectures.

The disciplines of information security include common security mechanisms in software systems and communications networks, together with their design principles and limitations. These are required for threat analysis performance, security requirements specification and the design and implementation of secure information systems.

The programme consists of six study modules:

- Special Module in Computer Science
- Advanced Module in Technical Information Security

  This consists of comprehensive courses in technical security, including the basics of cryptography and cryptosystems, information technology and network security. The module provides both theoretical knowledge and practical skills for understanding and developing secure systems:

  - Information Security Technology
  - Cryptosystems
  - Seminar on Network Security
  - Seminar on Internetworking

- Advanced Module in Network Services and Applications
- Methodological Principles
- Elective Studies
- Master's Thesis

NORDSECMOB - MASTER'S PROGRAMME IN SECURITY AND MOBILE COMPUTING

URL	http://nordsecmob.tkk.fi/index.html

NordSecMob - a Master's Programme in Security and Mobile Computing offered jointly by five universities:

- Aalto University School of Science and Technology (formerly Helsinki University of Technology, TKK)

- Royal Institute of Technology (KTH)
- Norwegian University of Science and Technology (NTNU)
- Technical University of Denmark (DTU)
- University of Tartu (UT)

During the programme, the students study at two of the five Nordic partner universities. Student mobility is implemented by dividing the studies of each student between the 'home' and 'host' university.

The student takes courses focusing on advanced topics in a selected area of specialisation:

- Aalto: Technical Information Security and Network Services
- KTH: Communications Systems Design
- NTNU: Information Security
- DTU: Reliable Software Systems
- UT: Mathematical Foundations of Cryptography

The NordSecMob focuses on fundamental mobile computing technologies. The technical basis of the programme includes current and future Internet and wireless network technologies, mobile devices and general software engineering skills. The subject areas include tools for safe software development and formal and mathematical validation and analysis. Throughout the programme, special attention is paid to information security.

This NordSecMob programme leads to two officially recognised MSc degrees issued by both home and host universities.

- AALTO MSc (Technology), Security and Mobile Computing
- KTH MSc (two years)
- NTNU MSc in Security and Mobile Computing
- DTU MSc in Engineering, Security and Mobile Computing
- UT MSc in Engineering (Computer Science)

EINDHOVEN UNIVERSITY OF TECHNOLOGY (TU/E), THE NETHERLANDS - MASTER'S PROGRAMME IN INFORMATION SECURITY TECHNOLOGY

URL	http://www.tue.nl/en/university/departments/mathematics-and-computer-science/studying/graduate-programs/masters-programs/information-security-technology/

This special Master, within the Master's programme, titled Computer Science and Engineering of TU/e, is offered by the Kerckhoffs Institute, a collaboration on computer security between the computer science departments of three leading Dutch universities: Eindhoven University of Technology, the University of Twente and Radboud University of Nijmegen. The programme consists of:

- Six compulsory courses (36 ECTS), in which students develop basic skills

- Three elective courses (18 ECTS), which can be selected from a list of courses in security offered by the Kerckhoffs Institute
- 36 ECTS for the other courses, which comprise both elective courses and courses that are compulsory at the three specific universities
- Master's thesis project (30 ECTS).

ETHZ ZURICH, SWITZERLAND - INFORMATION SECURITY MASTER'S TRACK

URL	http://www.infsecmaster.ethz.ch/

The Information Security Master's Track is one of the specialisation tracks of the Computer Science Master Programme at ETH Zurich. It lasts for three semesters, i.e. a year of courses and 6 months for the Master thesis. The programme offers 15+ security-related courses covering a number of topics, including cryptography, formal methods, system security, wireless and wired network security, e-privacy, fault tolerance etc. After successful completion of the programme, the student receives a Master of Science ETH in Computer Science with a focus on Information Security.

During the course, a student is expected to collect 90 ETCS credit points (26CP Focus Courses, 20CP Elective Courses, 8CP Multidisciplinary Courses, 4CP Foundations of Computer Science, 2CP GESS Course (2 CP) and the 30CP Master Thesis).

Generally, all students with a Bachelor of Science degree in CS or related fields can apply to this programme.

3. SSE INITIATIVES IN THE US

In this section we provide an overview of SSE initiatives in the US. These have been categorised according to their type (Academic or Government).

Secure Coding in C and C++ activity from the CERT Secure Coding has been included in the international initiatives' Training Courses (see section 0).

3.1. CERT SECURE CODING

The CERT Secure Coding Initiative is a security initiative of the Computer Emergency Response Team (CERT) programme. This programme is part of the Software Engineering Institute (SEI) at Canergie Mellon University (Pennsylvania, US). Some of its programmes are funded by the US Government.

In November 1988, the Defense Advanced Research Projects Agency (DARPA) charged the SEI with setting up a centre to coordinate communication between experts during security emergencies and to help prevent future incidents. As part of this task, CERT developed the Software Assurance Initiative, which includes Secure Coding Standards, Source Code Analysis Lab, Vulnerability analysis and Function extraction for malicious code.

The SEI is a federally-funded research and development centre, conducting software engineering research in acquisition, architecture and product lines, process improvement and performance measurement, security, and system interoperability and dependability.

The SEI works closely with defence and government organisations, mainly the Office of the Secretary of Defense/Acquisition, Technology, and Logistics (OSD/AT&L), industry, and academia, to continually improve software-intensive systems.

URL	http://www.cert.org/secure-coding/
Contact Method	http://www.cert.org/contact_cert/ Email, phone and address
Country	US
Geographic Scope	National
Type	Academic

The working areas of CERT Secure Coding are:

- Secure coding standards
  Proposes standards for enhancing the security of programming languages.
- International Standards Development
  Development of International Standards.
- Source Code Analysis Laboratory (SCALe)
  SCALe allows source code to be assessed against a set of secure coding standards. SCALe issues and certifies conformance testing when the test's findings have been addressed by the developers.

- Development Tools and Libraries
  Tools and libraries for secure software development

- TSP Secure
  Secure Team Software Process methodology.

CERT Secure Coding wishes to influence vendors to improve the basic, as shipped, security within their products. In order to achieve this, CERT Secure Coding works with software developers and software development organisations to reduce vulnerabilities resulting from coding errors (C, C++ or Java programming languages) before they are deployed. Also, CERT analysts evaluate the root causes of vulnerabilities and establish secure coding practices.

CERT collaborates with ISO in the establishment of several standards on secure coding.

RELEVANT RESULTS

Communication Media

Vulnerability Analysis Blog
Blog for security professionals and software developers

Training

Secure Coding in C and C++
Course of secure coding in C and C++ based on Addison-Wesley's material: "Secure Coding in C and C++" and "The CERT C Secure Coding Standard". See section 0.

Standards for Software Developers

CERT C Secure Coding Standard, Version 2.0
CERT C++ Secure Coding Standard
CERT Oracle Secure Coding Standard for Java
Participated in ISO TR 24731-1 (see section **Error! Reference source not found.**)

3.2. BUILD SECURITY IN

Build Security In is a US Government initiative based on an on-line website, where information related to software assurance is gathered. It is a project of the Strategic Initiatives Branch of the National Cyber Security Division (NCSD) of the US DHS. This initiative aims to provide software developers with practical guidance on how to produce secure software.

The US DHS Software Assurance Program seeks to reduce software vulnerabilities, minimise exploitation and address ways of improving the routine development and deployment of trustworthy software products. These activities will enable more secure and reliable software to support mission-critical requirements across enterprises and critical infrastructure.

Build Security In tries to shift the security paradigm from patch management to software assurance. This shift is designed to encourage software developers to raise overall software quality and security from the start, rather than relying on applying patches to systems after vulnerabilities are discovered.

This project intends to be a site where the US Software Engineering community (software developers and software development organisations) can find information and practical guidance on how to produce secure and reliable software.

Information in this site is split into three main areas. Once in a specific area, users can find links to relevant topics and, inside each, related articles, usually with a short abstract.

The three main areas are:

- Best Practice. Current best thinking, available technology, and industry practice
- Knowledge. Factual security-related knowledge that all engineers should be aware of
- Tools. Information about general classes of tools, with references to specific tools.

URL	https://buildsecurityin.us-cert.gov/bsi/home.html
Contact Method	https://buildsecurityin.us-cert.gov/bsi/bsi.html Email Additional Contact Email and phone
Country	US
Geographic Scope	National
Type	Government

The content of Build Security In is based on the principle that software security is fundamentally a software engineering problem and must be addressed in a systematic way throughout the software development life cycle. It contains, and is linked to, a broad range of information from different US sources about best practice, tools, guidelines, rules, principles, and other knowledge to help organisations build secure and reliable software.

Staff at Carnegie Mellon University's SEI (see section 3.1) contribute and review articles and maintain the site. Content has also been contributed by researchers and practitioners from Cigital, Inc. and other organisations (see Contributing Authors).

Members of the software assurance community are invited to submit articles for publication on the Build Security In website or to review submitted articles.

RELEVANT RESULTS

The description of each content area is taken from the Build Security In website.

Articles about Best Practice

- Acquisition
  The objective is to raise provider awareness. The articles describe an acquisition life-cycle framework for security activities, products, and reviews and for selected

acquisition contexts and life-cycle phases. The authors provide additional guidance on methods and resources for identifying and managing security risks.

- Architectural Risk Analysis
  Presents best practice for reviewing, assessing, and validating the specification, architecture, and design of a software system with respect to software security, reliability and performance goals. It includes a discussion on the identification, assessment, prioritisation, mitigation and validation of the risks associated with architectural flaws.

- Code Analysis
  Presents best practice in performing code analysis to uncover errors in, and improve the quality of, source code. Methods include manual code auditing, walkthroughs, static analysis, dynamic analysis, metric analysis, testability analysis, crypto analysis, random number analysis and fault injection.

- Deployment and Operations
  The objective is to describe, and provide pointers to, commonly accepted best practice and processes and the relevant characteristics of organisations that demonstrate competence in sustaining adequate security during deployment and operations.

- Governance and Management
  These articles provide a recommended sequence of steps to take in order to govern and manage enterprise, information, and software security. Detailed "how-to" guidance is not provided. Security at the enterprise and organisational level is addressed.

- Incident Management
  Incident management is defined. Examples of best practice in building an incident management capability are presented. It also takes a look at one particular component of an incident management capability, a computer security incident response team (CSIRT), and discusses its role in the systems development life cycle.

- Legacy Systems
  Describes the kinds of security risks that can be present in legacy systems, both in-house and commercially off-the-shelf, and offers guidance for assessing those risks and making sound decisions about addressing them.

- Measurement
  Best practice is described in relation to measurements for managing the quality of software systems during development. Several proposed measures for characterising specific security-related features are discussed, and the current extent of the practice of software measurement with specific attention to the use of security-related measures is described.

- Penetration Testing
  The concepts and goals of traditional penetration testing are discussed and recommendations are made on how these can be adopted to better suit the needs

of software developers. Additionally, the present state of the available tool base is described.

- Project Management
  Focuses on how security influences project management tasks and suggests refinements to existing practices. For example, project management can affect how well security requirements are satisfied, in terms of how the inputs from the technical, management, and operational communities are coordinated. Planning has to reflect the resources, effort, and risks associated with securing a new technology, such as Web Services. Design and implementation decisions may create new security threats, which should be represented in both project monitoring and planning.

- Requirements Engineering
  Best practice for security requirements engineering is presented, including processes that are specific to eliciting, specifying, analysing and validating security requirements. Specific techniques that are relevant to security requirements, such as the development of misuse/abuse cases, attack trees and specification techniques are also discussed or referenced.

- Risk Management
  A framework for identifying, tracking and managing software risks is provided. Best practices associated with software risk management are presented, together with content that discusses understanding software risks in a business context, identifying business and technical risks, prioritising business and technical risks, and defining risk mitigation strategies.

- Security Testing
  The primary objective is to improve the understanding of some of the processes of security testing, such as test vector generation, test code generation, results analysis and reporting. This will help testers to improve the generation of test vectors and increase their confidence when testing security function behaviours.

- Software Assurance
  A series of documents on software assurance in acquisition and outsourcing, software assurance in development, the software assurance life cycle and software assurance measurement and information needs.

- System Strategies
  System complexity is an aggregate of technology, scale, scope, operational, and organisational issues. Business usage, the technologies applied, and the changing operational environment raise software risks that are typically not addressed in current practice. It discusses the effects of the changing operational environment on the development of secure systems. Vulnerability analysis has typically concentrated on errors in coding or in the interfaces among components; however, system interactions can also be a seed bed for vulnerabilities. One article in this content area includes discussions on the software assurance challenges inherent in networked systems development and proposes a structured approach, using scenarios, to analysing potential system stresses.

- Training and Awareness
  This examines current practice in software security training and awareness offerings across the industry. It also briefly describes and compares the commercial sector's training offerings with current academic curricula in some of the US's top universities.

- White Box Testing
  This presents best practice in performing white box activities for testing code construction. The activities that provide the basis for white box dynamic analysis include: specifying the operational or expected usage or test profile; specifying key interfaces that feed data into the software; and compiling a list (or partial list) of undesirable output events, for which the software's behaviour should be monitored. Also discussed are strategies for examining the internal structure of a program, statement coverage, decision coverage, condition coverage, decision/condition coverage, and multiple-condition coverage.

Articles on Tools

- Black Box Testing
  Information is provided about black box testing tools. This term is used to refer to tools that take a black box view of the system under test; they do not rely on the availability of software source code and, in general, take an outside-in view of the software, which means that they try to explore the software's behaviour from the outside. The focus is on black box testing technologies that are unique to software.

- Modelling Tools
  This provides an introduction to modelling in the context of security analysis and discusses how tools can support security analysis during development. A model is an abstract representation of an object. The decomposition of a system might be grouped into components and their dependencies. A model can demonstrate the consistency of the system specifications or be a predictor of system behaviour. The analysis of system performance in data throughput or computation efficiency, so as to meet critical real-time performance requirements, depends on how that aspect of system behaviour is modelled.

- Penetration Testing Tools
  Information about penetration testing tools is provided.

- Source Code Analysis
  Outlines what automated security analysers can do, provides a business case for their use, and provides some criteria for evaluating individual tools. Code samples are provided for running tools against, in order to verify that the tools are able to detect known problems in the code.

Knowledge Articles

- Assurance Cases
  This introduces the concepts and benefits of creating and maintaining assurance cases for security. A security assurance case uses a structured set of arguments

and a corresponding body of evidence to demonstrate that a system satisfies specific claims with respect to its security properties.

- Attack Patterns
  These articles discuss the concept of attack patterns as a mechanism for capturing and communicating the attacker's perspective. Attack patterns are descriptions of common methods of exploiting software.

- Business Case Models
  This presents a conceptual framework for quantifying the cost and benefits of investments in secure coding techniques. Guidance on implementing the framework will include the variables and data elements to focus on and the means of measuring and quantifying them. With these measurements, one can calculate the economic benefits (cost) of these investments. Details are also provided on current practice and current research on the case for secure coding techniques.

- Coding Practices
  Describes methods, techniques, processes, tools and runtime libraries that can prevent or limit exploits against vulnerabilities. Each document describes the development and technology context in which the coding practice is applied, as well as the risk of not following the practice and the type of attacks that could result.

- Coding Rules
  Coding rules are representations of knowledge gained from real-world experience of potential vulnerabilities that exist in programming languages like C and C++. Creating and using software with a given coding environment enables the discovery of, and learning about, vulnerabilities that exist in this environment, how to recognise whether they crop up in our code and how to fix them. Coding Rules are the codification of this knowledge. They help software developers, whether manually or in conjunction with tools, to discover, explore, remove and eventually prevent security vulnerabilities in their code.

- Guidelines
  This provides information and data for educating software development professionals on the concept, applicability and value of design guidelines. In addition, this section collects, and makes available, a set of Design Guidelines to assist software development professionals with identifying and removing potential vulnerabilities in software systems.

  They are building, as well as developing, more mature and security-knowledge-aware design practices for future software systems.

- Lessons Learned
  This describes the lessons learned as a result of actual project experience. Lessons learned can be both positive and negative, providing both the opportunity to learn about techniques and approaches that can be followed on future projects and about the pitfalls to avoid.

- Principles

This provides information and data for educating software development professionals on the concept, applicability and value of software security principles. It also contains a set of key secure software principles that will help software development professionals analyse and create their software architectures from a security perspective and gain a greater understanding of the key underlying concepts and patterns that, depending on how they are addressed, can make software either more, or less, secure.

- SDLC Process
  This discusses the application of software assurance best practice in the context of various SDLC methodologies.

3.3. SOFTWARE ASSURANCE METRICS AND TOOL EVALUATION (SAMATE)

SAMATE is a US Government software assurance initiative, an inter-agency project between the US DHS and the National Institute of Standards and Technology (NIST). Its objective is to improve software assurance by developing metrics and methods for evaluating software security tools and identifying vulnerabilities related to coding practices and software engineering methods.

SAMATE's reference project develops test cases in order to examine the source code of tools and applications. It detects and reports weaknesses, so as to provide end users and software assurance tool developers with a set of known security flaws by which to evaluate their own tools.

The main output of this initiative is the SAMATE Reference Dataset (SRD), which is an online database fed regularly by SAMATE. This publicly available online database provides test cases to developers and end users with which to carry out security tools assessments.

URL	http://samate.nist.gov
Contact Method	http://samate.nist.gov Email
Country	US
Geographic Scope	National
Type	Government

SAMATE is dedicated to improving software assurance by developing methods for enabling software tool evaluations, measuring the effectiveness of tools and techniques, and identifying gaps in tools and methods. This project supports the US DHS's Software Assurance Tools and R&D Requirements Identification Program (in particular, Part 3, Technology (Tools and Requirements)), that addresses the identification, enhancement and development of software assurance tools.

The scope of the SAMATE project is broad, ranging from operating systems to firewalls, SCADA to web applications, source code security analysers to correct-by-construction methods

The SAMATE project consists of two parts:

- Development of metrics for the effectiveness of software security assessment (SSA) tools.
- Assessment of current SSA methods and tools in order to identify deficiencies which can lead to software product failures and vulnerabilities.

Finally, SAMATE is also developing some specifications aimed at software assurance tool developers, for classifying and evaluating these kinds of tools.

RELEVANT RESULTS

Communication Media

- **Annual workshop SATE 2010**
  Static Analysis Tool Exposition (SATE) has been set up to advance research (based on large test sets) in, and improvement of, static analysis tools that find security-relevant defects in source code. Briefly, participating tool makers run their tools on a set of programs. Researchers, led by NIST, analyse the tool reports. The results and experiences are reported at a workshop. The tool reports and analysis are made publicly available later.

- Publications
  Collection of SAMATE papers, Workshops and presentations.

Specifications

- Source Code Security Analysis
  Specifications and test plans for source code security analyser tools. This type of tool examines source code in order to detect and report weaknesses that can lead to security vulnerabilities.

- Web Application Scanner Specification
  "*Web Application Scanner Functional Specification Version 1.0*". These specifications are brought together in NIST Special Publication 500-269.

Test Cases

- SAMATE reference datasheet
  Provides users, researchers, and software security assurance tool developers with a set of known security flaws. These will allow end users to evaluate tools, and tool developers to test their methods.

- SRD database
  A collection of test cases aimed at detecting code weaknesses.

3.4. COMMON WEAKNESS ENUMERATION (CWE)

CWE is an initiative supported and co-sponsored by the NCSD of the US DHS and the NIST. It is currently maintained and led by MITRE Corporation.

The CWE is a formal list, or taxonomy, which classifies common types of software weakness. The main aims of CWE are to:

- Be the common taxonomy for classifying common software weaknesses related to its:
  - Architecture
  - Design
  - Code
- Serve as a standard classification for software security tools dealing with this type of software weakness.
- Provide a baseline from which to help the SSE community identify, mitigate and prevent this type of software weakness.

URL	http://cwe.mitre.org/ http://nvd.nist.gov/cwe.cfm
Contact Method	http://cwe.mitre.org/ Email
Country	US
Geographic Scope	National
Type	Government

This project uses the results of the SAMATE project to create the CWE list of weaknesses and its associated taxonomy and classification tree (see figure below taken from NIST)

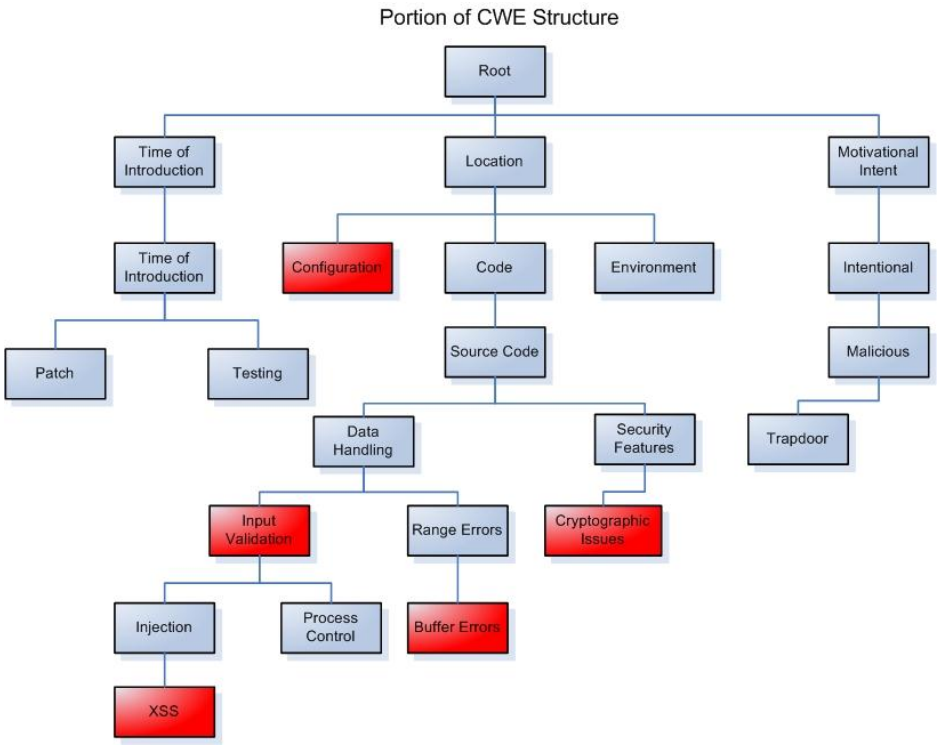


Figure 5: Portion of the CWE classification tree

It must also be pointed out that CWE is a community-developed, formal list of common software weaknesses involving academia, the commercial sector and the US Government.

RELEVANT RESULTS

CWE List
CWE's definitions and descriptions support the discovery of common types of software security flaws in code, prior to fielding. This means that both the users and developers of software assurance tools and services can use CWE as a mechanism for describing software security flaws.

The CWE List is offered in three different formats:

- High level dictionary of the identified software weaknesses
- Classification tree view, which can provide access to weaknesses through classification layering
- Graphical view of the above-mentioned classification tree to better understand weaknesses.

3.5. COMMON ATTACK PATTERN ENUMERATION AND CLASSIFICATION (CAPEC)

CAPEC is an initiative co-sponsored by the NCSD of the US DHS and led by the firm Cigital.

Secure software builders must protect themselves from relevant potential vulnerabilities. To identify and mitigate relevant vulnerabilities in software, the development community needs to understand the attacker's perspective and the approaches used to exploit software.

Attack patterns are descriptions of common methods for exploiting software, providing both the attacker's perspective and guidance on ways of mitigating their effect. They derive from the concept of design patterns applied in a destructive, rather than constructive, context and are generated from an in-depth analysis of specific examples of real-world exploits.

This initiative aims to provide a publicly available catalogue of attack patterns, together with a comprehensive schema and classification taxonomy. The philosophy is to evolve the catalogue with public participation and contributions and so form a standard mechanism for identifying, collecting, refining, and sharing attack patterns among the software community.

URL	http://capec.mitre.org
Contact Method	http://capec.mitre.org Email
Country	US
Geographic Scope	National
Type	Government

According to this initiative, the attack patterns information *"when captured in such a formalised way can bring considerable value for software security considerations through all phases of the SDLC and other security-related activities, including:*

- Requirements gathering
  Identification of relevant security requirements, misuse and abuse cases
- Architecture and design
  Provide context for architectural risk analysis and guidance for security architecture
- Implementation and coding
  Prioritise and guide activities of secure code review
- Software testing and quality assurance
  Provide context for appropriate risk-based and penetration testing
- Systems operation
  Leverage lessons learned from security incidents into preventative guidance
- Policy and standard generation
  Guide the identification of appropriate prescriptive organisational policies and standards"

RELEVANT RESULTS

Communication Media

Newsletter
Allows receipt of information and updates in the mailbox.

Attack Patterns

CAPEC List
Is a list, created by the community, of the latest release of common attack patterns.

Glossary and Abbreviations

API	Application Programming Interface. A particular set of rules and specifications that a software program can follow in order to access and make use of the services and resources provided by another particular software program, serving as an interface between different software programs.
CSS	Cascading Style Sheets
DHS	Department of Homeland Security
EC	European Commission
FP7	Seventh Framework Programme for Research and Technological Development. This is the European Union's main instrument for funding research in Europe and it will run from 2007-2013.
Fuzzer tool	A tool for testing the inputs of a program
HQ	Headquarters
ICT	Information and Communications Technology
IT	Information Technology
Maturity Model	A set of structured levels that describe how well the behaviours, practices and processes of an organisation can reliably and sustainably produce required outcomes.
OWASP	Open Web Application Security Project
SDK	Software Development Kit. A set of development tools that allows the creation of applications for a specific software package, software framework, hardware platform, computer system, video game console, operating system or similar platform.
SDLC	Software Development Life Cycle. A structure imposed on the development of a software product.
SEI	Software Engineering Institute
SOA	Service-Oriented Architecture

Software Assurance	A confidence level, indicating that software is free from vulnerabilities, either those intentionally designed into the software or accidentally inserted at any time during its life cycle and that it functions in the intended manner.
Software Engineering	The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, and the study of these approaches: i.e. the application of engineering to software.
SSE	Secure Software Engineering. SSE is using practices, processes, tools, and techniques that enable security issues to be addressed in every phase of the SDLC. Software that is developed with security in mind is typically more resistant to both intentional attack and unintentional failures. The aim of software security engineering is to build better, defect-free software. Systems that are constructed using more securely developed software are better able to: • Continue operating correctly in the presence of most attacks, by either resisting the exploitation of weaknesses in the software by attackers or tolerating the failures that result from such exploits • Limit the damage resulting from any failures caused by attack-triggered faults that the software was unable to resist or tolerate, and to recover as quickly as possible from those failures
W3C	World Wide Web Consortium
Widget	Standalone Web application that uses browser technologies