Network of Excellence

Deliverable D1.4

NESSoS Joint Virtual Research Lab
<table>
<thead>
<tr>
<th><strong>Project Number</strong></th>
<th>256980</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Project Title</strong></td>
<td>NESSoS</td>
</tr>
<tr>
<td><strong>Deliverable Type</strong></td>
<td>Report</td>
</tr>
</tbody>
</table>

| **Deliverable Number** | D1.4 |
| **Title of Deliverable** | NESSoS Joint Virtual Research Lab |
| **Nature of Deliverable** | R |
| **Dissemination Level** | Public |
| **Internal Version Number** | 1.2 |
| **Contractual Delivery Date** | September 30th, 2013 |
| **Actual Delivery Date** | October 31st, 2013 |
| **Contributing WPs** | WP 1 |
| **Editor(s)** | Cesare Bartolini (CNR)  
Antonia Bertolino (CNR) |
| **Author(s)** | Cesare Bartolini (CNR)  
Fabio Martinelli (CNR)  
Paolo Mori (CNR)  
Marinella Petrocchi (CNR)  
Manuel Clavel (IMDEA)  
Marianne Busch (LMU)  
Nora Koch (LMU)  
Maria Carmen Fernandez Gago (UMA)  
Gerardo Fernandez Navarrete (UMA) |
| **Reviewer(s)** | Marina Egea (ATOS)  
Carmen Fernandez Gago (SIEMENS)  
Marinella Petrocchi (CNR) |
Abstract

The Joint Virtual Research Lab (JVRL) is the technical infrastructure under development for the NES-SoS project. It features a collection of tools and services, provided by the partners and based on the partners' needs, which aim at being combined into a single, global architecture. The aim of the JVRL is to provide a set of facilities to streamline the cooperation between the partners, interaction among softwares, communication, and all sorts of information exchange that may be required by the project. This document describes the tools and services composing the JVRL, the current status of their integration, and what is needed in the short- and long-term future to achieve these goals. The document does not describe in detail each component of the JVRL (some, such as the mobility portal or the Common Body of Knowledge, have specific Work Packages dedicated to them), but only those not covered by other deliverables; for the others, the focus is on their role within the JVRL and how it relates to other components. A description of the plans for the life of the JVRL beyond the project completes the deliverable.

Keyword List

Technological infrastructure, tools, services, integration.
## Document History

<table>
<thead>
<tr>
<th>Version</th>
<th>Type of Change</th>
<th>Author(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>Document structure created.</td>
<td>Cesare Bartolini (CNR)</td>
</tr>
<tr>
<td>0.2</td>
<td>Introduction and section on web portal.</td>
<td>Cesare Bartolini (CNR)</td>
</tr>
<tr>
<td>0.3</td>
<td>Section on VEC.</td>
<td>Gerardo Fernandez Navarrete, Maria Carmen Fernandez Gago (UMA) Cesare Bartolini (CNR)</td>
</tr>
<tr>
<td>0.4</td>
<td>Sections on SVN and LDAP and concluding remarks.</td>
<td>Marianne Busch, Nora Koch (LMU) Manuel Clavel (IMDEA) Fabio Martinelli, Paolo Mori (CNR) Marinella Petrocchi (CNR) Cesare Bartolini (CNR)</td>
</tr>
<tr>
<td>0.5</td>
<td>Section about SDE.</td>
<td></td>
</tr>
<tr>
<td>0.6</td>
<td>Section on mobility portal.</td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>Section on CEE.</td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>Section on mailing lists.</td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>Section on NESSoS VMI for CEE.</td>
<td>Fabio Martinelli, Paolo Mori, Marco Di Stefano (CNR) Cesare Bartolini (CNR) Cesare Bartolini (CNR) Cesare Bartolini (CNR)</td>
</tr>
<tr>
<td>1.0</td>
<td>Added an executive summary.</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>Compliance with reviewers’ comments.</td>
<td></td>
</tr>
<tr>
<td>1.2</td>
<td>Final corrections.</td>
<td></td>
</tr>
</tbody>
</table>

## Document Review

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Reviewer</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013, Oct 15(^{th})</td>
<td>1.0</td>
<td>Carmen Fernandez Gago (UMA)</td>
<td>Many style fixes, some corrections on the content. Style and contents.</td>
</tr>
<tr>
<td>2013, Oct 28(^{th})</td>
<td>1.0</td>
<td>Marinella Petrocchi (CNR)</td>
<td>(For the editor: please, list a resume of the most relevant reviewers’ comments)</td>
</tr>
</tbody>
</table>
# Table of Contents

**LIST OF FIGURES** ............................................................................................................. 6  
**LIST OF TABLES** ............................................................................................................... 7  
1 **EXECUTIVE SUMMARY** ............................................................................................... 9  
2 **INTRODUCTION** ........................................................................................................... 10  
3 **GENERAL OVERVIEW** ............................................................................................... 11  
4 **WEB PORTAL** ............................................................................................................... 14  
   4.1 New sections ............................................................................................................. 15  
   4.2 Roadmap survey ...................................................................................................... 15  
   4.3 Queries .................................................................................................................... 15  
5 **COMMON BODY OF KNOWLEDGE** ............................................................................ 17  
   5.1 Technical changes .................................................................................................... 17  
   5.2 Changes to contents ............................................................................................... 18  
6 **MOBILITY PORTAL** .................................................................................................... 19  
7 **VIRTUAL EDUCATION CENTRE** ........................................................................... 21  
   7.1 Mahara ................................................................................................................... 21  
8 **SDLC TOOLS WORKBENCH** .................................................................................... 23  
   8.1 SDE – Service Development Environment .......................................................... 23  
   8.2 Integrated Security-related Tools ........................................................................... 24  
   8.3 Connections between CBK and SDE ..................................................................... 25  
9 **NESSoS CLOUD EXECUTION ENVIRONMENT (CEE)** ........................................... 27  
   9.1 Introduction to Cloud ............................................................................................... 27  
   9.2 OpenNebula .......................................................................................................... 27  
   9.3 NESSoS CEE ......................................................................................................... 28  
   9.4 NESSoS Virtual Machine Image (VMI) ................................................................. 29  
10 **MAILING lists** ............................................................................................................ 31  
11 **SUBVERSION REPOSITORY** ................................................................................... 32  
12 **LDAP DIRECTORY** .................................................................................................. 33  
13 **INTEGRATION** .......................................................................................................... 35  
   13.1 Integration status .................................................................................................. 35  
   13.2 Usage analytics ................................................................................................... 36  
   13.3 Future integration ............................................................................................... 36
List of Figures

Figure 3.1: Schematic view of the JVRL ................................................................. 13
Figure 4.1: The main page of the NESSoS web portal ........................................ 14
Figure 5.1: An overview of knowledge objects in the CBK ................................. 17
Figure 5.2: Development of knowledge object numbers over time ....................... 18
Figure 7.1: Mahara e-portfolio ............................................................ 21
Figure 7.2: Exporting a Forum Discussion ....................................................... 22
Figure 7.3: Forum Discussion in Mahara ......................................................... 22
Figure 8.1: SDE look and feel ................................................................. 23
Figure 8.2: Interplay of CBK and SDE ......................................................... 25
Figure 9.1: The NESSoS customized interface for accessing OpenNebula services ........................................................................... 29
Figure 9.2: The NESSoS Virtual Machine Image available on NESSoS CEE ........ 30
Figure 12.1: The NESSoS LDAP tree ............................................................ 34
List of Tables

Table 3.1: JVRL components overview. ............................................................... 12

Table 6.1: Mobility Portal: General statistics....................................................... 19

Table 6.2: Mobility Portal: Unique visitors in each month.................................... 20

Table 10.1: NESSoS Mailing Lists ................................................................. 31

Table 11.1: SVN commits by partner............................................................... 32

Table 12.1: Status of LDAP integration.......................................................... 33

Table 13.1: JVRL components by partner....................................................... 35
List of acronyms

**BSCW** Basic Support for Cooperative Work  
**CBK** Common Body of Knowledge  
**CEE** Cloud Execution Environment  
**DoW** Description of Work  
**HTTP** Hypertext Transfer Protocol  
**IDE** Integrated Development Environment  
**JVRL** Joint Virtual Research Lab  
**KO** Knowledge Object  
**LDAP** Lightweight Directory Access Protocol  
**LMS** Learning Management System  
**NESSoS** Network of Excellence on Engineering Secure Future Internet Software Services and Systems  
**NoE** Network of Excellence  
**OSGi** Open Service Gateway initiative  
**CLASP** Comprehensive, Lightweight Application Security Process  
**PLE** Personal Learning Environment  
**SCO** Shareable Content Object  
**SCORM** Shareable Content Object Reference Model  
**SDE** Service Development Environment  
**SDL** Security Development Lifecycle  
**SDLC** Software Development Life Cycle  
**SSO** Single Sign-On  
**SMW** Semantic MediaWiki  
**SVN** Subversion  
**UI** User Interface  
**VEC** Virtual Education Centre  
**VM** Virtual Machine  
**WP** Work Package
1 Executive summary

This deliverable highlights the progress of the NESSoS JVRL, the core of WP1, during the third year of the project. Most of the effort on WP1 was focused on the fine-tuning of existing components, better integration, and those improvements which were required according to the project’s and partners’ needs.

The main improvements to the JVRL highlighted in this deliverable can be summed up as follows:

- **the NESSoS web portal** has been used to support the survey to establish the objectives of the research roadmap. Some queries have been added to the web portal for dissemination purposes (e.g., for reports on the NESSoS publications), as well as the new components and deliverables that have been connected to or uploaded onto the web portal;

- **the Common Body of Knowledge** has been enhanced with LDAP capabilities, and thus it is now possible for NESSoS users to log into the CBK using the shared NESSoS credentials. Also, the support for creating CBK-only users (visitors) has been established. The CBK is now integrated in the web usage statistics of the NESSoS project. The content of the knowledge base has been extended with additional knowledge objects, including many contributions on the partners’ part;

- **the mobility portal** has not changed substantially, although it has been used for the purposes of visits related to the research topics of the project. This deliverable highlights the usage statistics of the mobility portal, while the results expressed in the mobility portal in terms of mobility actions in the NESSoS project are deferred to the deliverables related to the mobility WP (WP3);

- **the Virtual Education Centre** also has not been structurally changed during the last year. Although content has been added to the system, the structure remains as during the previous year. This deliverable only highlights the essential interactions between Moodle and Mahara, two of the technologies underlying the Virtual Education Centre;

- **the SDLC tools workbench** has been thoroughly extended, and thanks to the contributions of many partners it now sports a large number of integrated tools;

- **the Cloud Execution Environment** has been finalized, integrated with the NESSoS LDAP authentication system, and also enhanced by defining a virtual machine preset with a complete instance of the SDLC tools workbench. Users can now log into the Cloud Execution Environment and, in addition to creating their own virtual machines, can also retrieve an instance of the predefined NESSoS virtual machine, which enables them to run all the tools included therein;

- **the mailing lists** were not changed at all. This deliverable only contains a brief report on usage statistics of the mailing lists;

- **the Subversion repository** has been used a lot throughout the project. Recently, it has been connected to the LDAP directory. This deliverable also contains a per-partner usage report of the SVN repository;

- **the LDAP directory** is now, under the hood, the core of the integration of the NESSoS JVRL facilities. In a completely transparent way, accesses to (almost) all the components of the JVRL is handled by LDAP. In the deliverable, the status of the existing connections to the NESSoS LDAP directory is described.

Finally, a brief description of the integration between the JVRL components in a composite view of the JVRL is given.
2 Introduction

The purpose of the NoE NESSoS is to establish connections between researchers in the field of secure service engineering. Partners of the NoE cooperate in their research, through scientific publications and information exchange.

The JVRL offers the technical infrastructure for the NoE’s members. Made up of an ample collection of tools and softwares, the JVRL aims at being the shared technical platform underlying all of NESSoS’s collaborative work. The JVRL addresses several needs of the cooperative research in secure software engineering, from tools for the organisation of meetings and workshops to publication-sharing areas, from cooperative development to e-learning facilities, from means of communication to development environments.

The JVRL is therefore a collection of many tools for different purposes, provided and developed by the partners, always keeping in mind their own needs and those of other members of the consortium of the NoE. But these tools are not supposed to live in separate universes, with each tool limited to achieving its individual purpose. The JVRL aims at being more than just a collection of tools. NESSoS is about cooperation and integration, and these must be achieved not only in the research, but also in the technical infrastructure. The JVRL is supposed to be an integrated set of tools, where the various components interact among themselves, and the content or data provided within one of them is readily (and possibly automatically) available elsewhere.

To a large extent, this objective has been reached with the JVRL. During the first year of the project, most of the tools were put into place, and the first steps toward integration have been taken; during the second year, additional components where added, and the major integration effort was carried out; during the third year, finalization and refinement have completed the work. Some of the tools are under further development. The objective of making the JVRL a coordinated entity, a “virtual laboratory” where the user, after being granted access (where access is required), can use all the facilities made available to his or her privilege level, has been mostly achieved.

In some cases, there were logical or systematic issues against integration. For example, adding usage statistics to the web sites of some components hosted in Germany was against national regulations and would have required some administrative workaround that was not advisable. The BSCW has never been added to the single sign-on procedure supported by the LDAP directory, mainly for security and liability reasons. Apart from these criticalities, which have individually been analyzed and discussed before deciding against merging them into the overall infrastructure, the JVRL’s degree of integration is on par with the original expectations of the project.

Up to three different access levels can be found in any JVRL component: an external level for public visitors, an internal level for registered members, and an administrative level. The external level is made up primarily of web sites, and provides all public information to outsiders. Information about the partners, the status of the NoE, published papers, tools developed, related events, public training courses and so on are all available on the various web sites composing the NESSoS “outer” layer. Internal visibility grants access to all resources, such as development code, deliverables’ work in progress, publishing facilities, inter-partner communication and the like. The administrative level is accessible only to the developers and maintainers of each individual JVRL component, and provides a superuser access for invasive operations such as changing, fixing, rewriting, moving. Within these macro-levels, of course, each individual component will have its own access policies for different users.

This deliverable is structured in such a way as to describe all the components, their current integration status, and future integration plans. Specifically, after a brief overview of the main building blocks of the JVRL, each chapter shall focus attention on individual components, describing them at a high level (leaving a more detailed description to deliverables for their specific work packages), and how they fit into the global view of the JVRL.
3 General overview

The JVRL is composed of several components. Some are services offering external visibility through a public web site, while others are tools accessible only by NESSoS partners and used only for internal collaboration.

The current JVRL components are the following:

- the NESSoS web portal, located at http://www.nessos-project.eu/, is the main component for external visibility, and the central hub of the JVRL. The web portal contains all the information about the project; some are excerpts from the DoW, such as the project objectives or WP description, but most are updates on the project's advancement. Important news, accepted deliverables, and naturally links to the other JVRL components all have their natural home in the web portal. It is also used to collect and store NESSoS-related publications; it contains a calendar of related events (conferences, meetings, schools, workshops) where every member of the consortium can publish an event; and recently it has been extended with a plugin to create and manage surveys. Details on the advancement of the NESSoS web portal are in Chapter 4;

- the mobility portal, located at http://www.nessos-project.eu/mbp, provides information about NESSoS’s mobility program, for a tighter collaboration among the partners involved. The mobility portal is part of the mobility program, described in detail in the deliverables for WP3. An update on the status of the mobility portal is in Chapter 6;

- the CBK is another web-based service located at http://www.nessos-project.eu/cbk. This is a shared knowledge base for all the tools, technologies and methodologies created under the NESSoS flag. While a detailed description of the CBK is in the WP5 deliverables, its status within the JVRL is given in brief in Chapter 5;

- the Virtual Education Centre http://www.nessos-project.eu/vec is the last and most recent of the currently-existing public web sites for the NESSoS project. It is a front-end for the tool for interactive learning based on the Moodle platform. The Virtual Education Centre is part of WP13, and a detailed report will be published in the deliverables for that WP. Its current status in the JVRL is described in Chapter 7;

- the Software Development Life Cycle (SDLC) tools workbench, also called Service Development Environment (SDE), has a website at http://www.nessos-project.eu/sde, although it is not a web-based service like those mentioned above, rather it is an implementation of the CBK tools. It is a modified version of the Eclipse Integrated Development Environment (IDE) where the tools developed by the partners (and described in the CBK) can be integrated. Towards the end of the project, the degree of integration among the tools might allow the output of a tool to be fed as an input to another, thus allowing a full chain for software development, from the initial design to the deployment and test phase. The SDE is described in detail in the deliverables for WP2, whereas its integration in the JVRL is in Chapter 8 of this document;

- a web-based tool is also the Basic Support for Cooperative Work (BSCW), located at http://www.nessos-project.eu/bscw, but it is visible only to NESSoS partners. Although it has several functionalities, including a personal and customisable web portal, within NESSoS it is used only for storage of shared documents. Meeting information, slides of talks, completed deliverables, instructions for using the various JVRL tools are all stored within the BSCW;

- with a functionality in part similar to that of BSCW, the Subversion repository (Chapter 11) is used to store temporary versions of work-in-progress content. It is used to store the ongoing project deliverables, and for collaborative development of software among several partners. The repository also provides a web interface which can be accessed in read-only mode (but only for authenticated users) at http://www.nessos-project.eu/svn. Chapter 11 gives some report on the status of the activity within the NESSoS SVN repository;

- the NESSoS Cloud Execution Environment (CEE) (Chapter 9) is the latest addition to the NESSoS JVRL. It consists of a cloud system which allows NESSoS users to execute virtual machines. Its
main purpose is to provide a machine to run test suites of standalone and integrated NESSoS-related tools;

- the NESSoS mailing lists provide a quick and efficient means of mass communication among the partners. NESSoS uses several mailing lists, depending on the purpose and domain of interest of each communication. http://www.nessos-project.eu/mls points to the mailing list preferences page, where a user can select the mailing lists to subscribe to. Some reports on the mailing lists are provided in Chapter 10;

- the LDAP directory represents a potential infrastructure for the integration of all the above tools. Since most of them can be bound to the LDAP directory, this can be used as the core of the network integration, becoming the central hub for storing user information such as access credentials to the various services. During the first year, the LDAP was mainly an authentication back-end for the NESSoS web portal, but during the second year several tools have been extended to support it as well, as described in Chapter 12. The LDAP directory has a web interface for its management, but it is accessible only to the administrators, therefore no public URL for it has been provided.

At a first glance, some of these components may appear quite similar, and it is true that some might be used to perform tasks done by another. However, different components are used for different purposes. Specifically:

- both the BSCW and the Subversion repository can be used for versioning and collaborative documents. However, uploading documents into the BSCW requires the use of a user-interface, while the SVN does not provide an immediate overview of its content. For this reason, documents which are subject to frequent changes (such as work-in-progress deliverables, software under development, joint papers, and so on) are more fit for the SVN, especially if they are text-based (e.g., programming code or LaTeX documents). On the other hand, the BSCW is used for documents which don’t need a lot of changes (completed deliverables, meeting agendas, reviewer comments) and can be conveniently stored for historical purposes;

- some overlapping also exists between the BSCW and the web portal. However, the main difference is that the full content of the web portal is public, whereas the BSCW contains documents which pertain the consortium only. For example, completed deliverables are stored both in the BSCW and in the web portal, but the former contains all deliverables, including internal and confidential ones, while the latter only contains those deliverables which are open to the public community.

Table 3.1 summarises the various components, the partners and people in charge of them, and a link to the resource (if applicable). Figure 3.1 shows a schematic diagram of the components and the connections among them (both existing and under development); dotted lines represent parts of the JVRL which are currently work in progress.

<table>
<thead>
<tr>
<th>Component</th>
<th>Partner</th>
<th>Manager</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web portal</td>
<td>CNR</td>
<td>Cesare Bartolini</td>
<td><a href="http://www.nessos-project.eu/">http://www.nessos-project.eu/</a></td>
</tr>
<tr>
<td>Common body of knowledge</td>
<td>UDE</td>
<td>Widura Schwittek</td>
<td><a href="http://www.nessos-project.eu/cbk">http://www.nessos-project.eu/cbk</a></td>
</tr>
<tr>
<td>Mobility portal</td>
<td>IMDEA</td>
<td>Manuel Clavel</td>
<td><a href="http://www.nessos-project.eu/mbp">http://www.nessos-project.eu/mbp</a></td>
</tr>
<tr>
<td>Virtual Education Centre</td>
<td>UMA</td>
<td>Miguel Guerrero</td>
<td><a href="http://www.nessos-project.eu/vec">http://www.nessos-project.eu/vec</a></td>
</tr>
<tr>
<td>Service Development Environment</td>
<td>LMU</td>
<td>Marianne Busch</td>
<td><a href="http://www.nessos-project.eu/sde">http://www.nessos-project.eu/sde</a></td>
</tr>
<tr>
<td>BSCW</td>
<td>UNITN</td>
<td>Sebastiano Perisi</td>
<td><a href="http://www.nessos-project.eu/bscw">http://www.nessos-project.eu/bscw</a></td>
</tr>
<tr>
<td>Mailing lists</td>
<td>CNR</td>
<td>Marinella Petrocchi</td>
<td><a href="http://www.nessos-project.eu/mls">http://www.nessos-project.eu/mls</a></td>
</tr>
<tr>
<td>Subversion repository</td>
<td>CNR</td>
<td>Paolo Mori</td>
<td><a href="http://www.nessos-project.eu/svn">http://www.nessos-project.eu/svn</a></td>
</tr>
<tr>
<td>Cloud Execution Environment</td>
<td>CNR</td>
<td>Paolo Mori</td>
<td><a href="http://www.nessos-project.eu/cee">http://www.nessos-project.eu/cee</a></td>
</tr>
<tr>
<td>LDAP directory</td>
<td>CNR</td>
<td>Cesare Bartolini</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.1: JVRL components overview.

Future components providing a web access will also be bound to the domain. The URLs are only redirection, because every component resides on a separate server, with different administrators. The option of transferring all components to the same server has been discussed and rejected, because the
current setup is considered satisfactory in that it confers each partner complete control over its components. Individual management allows a more prompt reaction in responding to emerging needs, whereas a centralized management would not be so efficient because any problem or request would need to be forwarded to the developers or the maintainers of the individual component, with a consequent delay in the time needed to fix or improve it.
4 Web portal

The web portal is the central hub of the JVRL. Every component of the JVRL which must have an external visibility (web sites, public tools, etc.) is linked on the web portal. Additionally, the web portal contains public information not pertaining to the JVRL, such as the planning of events or the NESSoS-related publications. Ideally, the web portal is the starting point for the NoE’s dissemination, in terms of information about the project and its partners, results achieved, or publicly available improvements in the research fields.

The web portal is located at the URL http://www.nessos-project.eu/. At first sight (Figure 4.1 depicts an overview of the front page), the portal describes the project at large, illustrating the objectives of the NoE, the partners of the consortium, and the project structure in Work Packages. It also features some links to the FP7 homepage, and a login to a reserved area.

![Figure 4.1: The main page of the NESSoS web portal.](image)

The web portal is enhanced with usage statistics. This is a feature that has been implemented on the main web portal and the CBK. Statistics are collected using an attached Google Analytics account. Google Analytics collects data from all connected sources (both the web portal and the CBK and in its most basic form provides a unified report. The data show a frequent usage of NESSoS resources: in the time window between July 2011 (when statistics were implemented) and July 2013, the resources have received a total of 13031 visits, 7778 by unique users, with spikes occurring around meetings and NESSoS-related conferences. Around 60% of the traffic comes from searches, and 20% from referrals. Further details on the traffic data is part of the dissemination deliverables.

---

1^http://www.google.com/analytics/.
With respect to deliverable D1.3 [13], there have been several updates to the web portal. The following list summarises the changes from the previous deliverable (the last two are visible only to certain members of the consortium):

- integration of the CBK in the usage analytics;
- integration and linking of the CEE;
- page for Y2 deliverables;
- survey on the NESSoS roadmap;
- query for reporting citations of NESSoS publications;
- query for the results of the survey on the NESSoS roadmap.

4.1 New sections

The content structure has not undergone any major revision. Some minor sections have been added or highlighted. Specifically, links to the most prominent JVRL components have been highlighted on a separate menu. This has been done in order to facilitate the access to other resources of the JVRL, making the web portal more efficient as a central hub for the project.

After the approval of the deliverables at the end of the second review, the Deliverables section has been updated. The Y2 deliverables which have been approved by the Commission and are not marked as reserved documents have been uploaded to the web portal and linked therein; the storage is on the same machine hosting the web portal with a uniform file-naming policy. The deliverables’ page will keep being updated up to the approval of all the project’s deliverables.

4.2 Roadmap survey

During the second year of the project, a tool for managing surveys, a Joomla! plugin called JQuarks4S, had been added to the web portal. Despite some imperfections and issues with the tool, experienced in its first usage (described in [10]), when the need arose to create a new survey to assess the NESSoS roadmap, the same tool was used. Knowledge of the tool and its free availability provided a full compensation for its drawbacks.

The survey on the NESSoS roadmap was implemented in April and kept open to the general public (without the need for access credentials to the JVRL) for a couple of months. A query was then developed to report the results in an efficient way.

The content of the survey and its results will be presented in D4.3 part II [16].

4.3 Queries

Similarly to what was described in [10], a script to query and report the NESSoS publications crafted during the second year of the project was developed. This is because the J!Research plugin used for managing publications, while providing an efficient way to store publications and their metadata, is not very efficient at filtering and reporting them, especially in the perspective of a per-partner/per-WP production report.

In addition to that, some scripts have been developed to discover the number of citations for every publication. These scripts work by querying Google Scholar² and organizing the results in a table. Although some inaccuracies do exist because of the (third-party) scripts that have been used to query Google Scholar (which unfortunately does not provide an API for querying it), the script is useful in that it gives an approximate idea of the relevance of all publications endorsed by the NESSoS project.

²http://scholar.google.it/.

NESSoS - 256980 16
Additionally, the report from the survey on the NESSoS roadmap (to drive future research directions in secure service engineering) are queried and reported using a separate script.

In all these cases, if a need arises to make said results available to all partners, the queries could be integrated into the registered area of the NESSoS web portal, with visible links for reaching them.
5 Common body of knowledge

The Common Body of Knowledge, or CBK for short (http://www.nessos-project.eu/cbk), aims at providing a guide to the different existing bodies of knowledge that comprise NESSoS’s field of interest “engineering secure software and services”. In addition to this, the CBK provides mechanisms to establish a common terminology to which each author can relate his/her own terms to. Elaborate means have been introduced to search and browse the CBK using a faceted search mechanism, a tool to compare knowledge objects side by side and a tool to identify CBK research gaps to support researchers conducting a mapping study.

A brief overview is given on the Common Body of Knowledge platform as part of the JVRL. A more detailed description can be found in deliverables D5.1 [12], D5.2 [15] and D5.3 [17]. In the following we differentiate between technical and content updates.

5.1 Technical changes

On the technical side, many new updates of the existing third-party software components have been made, which included bug fixes and some features, that contribute to the overall CBK experience.

We executed several software version updates on the CBK platform. We migrated the platform from SMW+ 1.5.1 to SMW+ 1.7.0 manually. This resulted in an improved layout, the faceted search works on the basis of Apache Solr, the formerly so-called Ontology browser turns into the Improved Data Explorer.

In addition, the new version of the SMW+ can execute automatic platform updates without server restarts or manual migration steps. We also moved from MediaWiki 1.16 to MediaWiki 1.17 and we moved from Semantic MediaWiki 1.5.6 to Semantic MediaWiki 1.7.1.

We improved our backup strategy to ensure business continuity in case of an attack. The reason for establishing business continuity instead of finding and fixing all vulnerabilities of the system, e.g., via penetration testing is that we lack the resource to conduct intensive penetration testing to ensure resilience against all kinds of attacks. Hence, we implemented daily incremental VM-Image backups and we store these for five days. We also conduct a full backup once a month to ensure the availability of the system.

Furthermore, we create a daily dump of the CBK database and the SMW+ installation. This data is stored on an external ftp server and versioned using the subversion tool.

The CBK is now fully integrated with the NESSoS LDAP system. It is still possible to log in with the formerly registered user but a second option allows to authenticate against the NESSoS LDAP.

A CBK blog has been established, on which CBK updates are announced, changes are documented and CBK stats such as the Knowledge Object (KO) count are given. The underlying system is the open source blog software WordPress, which has been installed in the newest version (3.4.1) and is linked from the CBK’s main menu. The CBK blog follows a minimalist design paradigm serving the sole purpose of keeping stakeholders and interested people up-to-date on the current CBK’s development activities.

We added an overview of the kind and amount of knowledge objects in the CBK to the main page (see Figure 5.1).

Figure 5.1: An overview of knowledge objects in the CBK
Finally, we integrated plugins to generate PDFs out of the CBK. This lays the basis for the upcoming handbook feature which the user can generate from the CBK.

5.2 Changes to contents

We conducted the following contextual updates. We added a long and a brief description of the CBK’s background and goals on the main page of the CBK. In addition, we added an image and an introductory video on the main page to give a quick overview of the main goals. We also added a detailed description on how to add knowledge objects to the CBK including many screenshots for a better understanding. Furthermore, we added help texts to each attribute of all forms.

The set of knowledge objects have been increased quantitatively. We added the mature knowledge objects Microsoft’s Security Development Lifecycle (SDL), McGraw’s Touchpoints, and the OWASP Comprehensive, Lightweight Application Security Process (CLASP). Many new knowledge objects have been contributed by the partners as well (see Figure 5.2).

![Development of Knowledge Object Types over time](image)

**Figure 5.2: Development of knowledge object numbers over time**

From the quality point of view, a peer-review process has been established in between the NESSoS partners. As such, all knowledge objects have been reviewed by at least two other NESSoS partners to ensure a certain level of quality of the CBK’s contents.

We also updated the ontology with a new KO type for security standards.
6 Mobility portal

The mobility of network fellows within NESSoS is a mechanism that supports the integration of activities across various sites. It brings together researchers working on related topics; it drives knowledge exchange and knowledge generation through union and diversity. It increases the capability of joint cooperation among researchers.

The Mobility Portal (http://www.nessos-project.eu/mbp) supports the NESSoS Mobility Programme in different ways. Basically, for non-registered visitors,

- it provides up-to-date information about mobility actions already performed within the Mobility Programme: namely, for each of them, the starting date, the participant, the institutions which are involved, the duration, the source for the funding, and the summary of the action;

In addition, for those registrants in the Mobility Portal,

- it provides an on-line application form to submit a request for a mobility grant, by filling the following fields: the destination partner, the estimated duration, the objectives of the mobility action, and the requested grant (for travel and lodging);
- it provides on-line access to their requests for grants, while they have not been yet accepted: in particular, requests can be reviewed, modify, and even withdraw by their own creators;
- it provides an on-line form to report about a mobility actions already performed, including the starting date, the duration, the budget (only travel and lodging), the funding source, and a summary. This report is automatically published, and available for visitors, in the Mobility Portal.

Last but not least, for the coordinators of the Mobility Programme,

- it provides on-line access to the pending requests for mobility grants, from which they can accept or reject the applications;
- it provides on-line access to the Mobility Portal’ activity logs.

Tables 6.1 and 6.2 show statistics about the Mobility Portal’s visitors during the period from September 1st, 2012 to July 31st, 2013. They were automatically generated with Visitors Web Log Analyzer version 0.7. (Note: multiple hits with the same IP, user agent and access day, are considered a single visit).

The Mobility Portal is now integrated with the NESSoS LDAP system.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of unique visitors</td>
<td>1,049</td>
<td>1,347</td>
<td>427</td>
</tr>
<tr>
<td>Number of unique visitors from Google</td>
<td>72</td>
<td>20</td>
<td>34</td>
</tr>
<tr>
<td>Number of unique visitors from NESSoS Web Portal</td>
<td>44</td>
<td>68</td>
<td>75</td>
</tr>
</tbody>
</table>

Table 6.1: Mobility Portal: General statistics.
Table 6.2: Mobility Portal: Unique visitors in each month.

<table>
<thead>
<tr>
<th></th>
<th>2013</th>
<th>2012</th>
<th>2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>88</td>
<td>136</td>
<td>–</td>
</tr>
<tr>
<td>Feb</td>
<td>62</td>
<td>70</td>
<td>80</td>
</tr>
<tr>
<td>Mar</td>
<td>93</td>
<td>71</td>
<td>45</td>
</tr>
<tr>
<td>Apr</td>
<td>61</td>
<td>103</td>
<td>45</td>
</tr>
<tr>
<td>May</td>
<td>72</td>
<td>130</td>
<td>38</td>
</tr>
<tr>
<td>Jun</td>
<td>133</td>
<td>166</td>
<td>62</td>
</tr>
<tr>
<td>Jul</td>
<td>117</td>
<td>238</td>
<td>80</td>
</tr>
<tr>
<td>Aug</td>
<td>–</td>
<td>200</td>
<td>77</td>
</tr>
<tr>
<td>Sept</td>
<td>–</td>
<td>110</td>
<td>87</td>
</tr>
<tr>
<td>Oct</td>
<td>–</td>
<td>95</td>
<td>36</td>
</tr>
<tr>
<td>Nov</td>
<td>–</td>
<td>118</td>
<td>41</td>
</tr>
<tr>
<td>Dec</td>
<td>–</td>
<td>100</td>
<td>69</td>
</tr>
</tbody>
</table>
7 Virtual Education Centre

The Virtual Education Centre (VEC) is a tool that establishes a technical infrastructure solution used to spread the knowledge acquired, as well as for education and training purposes. It has been deployed as a standalone service due to the plan to persist beyond the life of NESSoS, but it uses common authentication base of the JVRL platform.

As we described in [10] the VEC is composed of a Learning Management System, which is Moodle in our case, and an e-portfolio, Mahara. Whereas in [13] we gave an overview of the Learning Management System, this section provides insight on Mahara.

7.1 Mahara

Mahara is a fully featured web application to build your electronic portfolio. You can create journals, upload files, embed social media resources from the web and collaborate with other users in groups, among other activities, as shown in Figure 7.1.

![Figure 7.1: Mahara e-portfolio](image)

The Mahara core is compatible and pluggable with the Moodle core. This means that we can export Moodle resources to Mahara e-portfolio. Some considerations to take are the following:

- Each type of activity can be exportable or not.
- Each type of activity has its own permissions to determine what can be exported.
- Students can only export content created by themselves by default (i.e. 'their own content').

An example with a forum type activity is shown next. First, we must go into an existing forum and click on any discussion, as depicted in Figure 7.2.

Then, we can export the forum post by using Leap2A format (although we could use HTML format as well). After checking the export details, we can click on 'Continue' and we will be able to see the new contents in Mahara, as shown in Figure 7.3.
Figure 7.2: Exporting a Forum Discussion

Figure 7.3: Forum Discussion in Mahara
8 SDLC tools workbench

The development of secure software and systems implies dealing with multiple programming languages, platforms and tools. The tasks carried out during the development process are ranged from requirements specification to testing comprising modelling, and implementation as well as validation and verification. To enable developers to find, use, and combine security-related tools, a SDLC tools workbench is used, namely the SDE [20].

The SDE provides an overview of available tools and their area of application and allows developers to use these tools in a homogeneous way, re-arranging tool functionality as required, and last but not least enables users to stay on a chosen level of abstraction, hiding details of underlying tools and formal details as much as possible.

In this chapter, we give an overview on the core of the SDE platform, the tools integrated so far in the SDE and present the connections between the SDE and the CBK (for detailed information about the CBK see Chapter 5).

8.1 SDE – Service Development Environment

The SDE is based on the Eclipse platform [5] and its underlying, service-oriented OSGi [19] framework. OSGi is based on so-called bundles, which are components grouping a set of Java classes and metadata providing among other things name, description and version. An OSGi bundle may provide arbitrary services to the platform and therefore all tools are integrated as bundles which offer certain functions for invocation by the SDE platform. This means that tools can be seamlessly integrated in the SDE by writing a wrapper, which defines how functions of each tool can be used.

Furthermore, the SDE provides the ability to compose new tools out of existing ones, a process known as orchestration in services literature. Creating a new service as orchestrations of other existing services is possible using either a textual, JavaScript-based approach, or a graphical work-flow approach. With the
latter the user can build tool chains by connecting boxes that represent the integrated tools. Figure 8.1 illustrates these features of the SDE tool workbench:

- the tool browser lists available tools by category;
- the function browser lists the functionalities of a tool and it is possible to execute them by filling the parameters manually or by using a set of predefined examples from the web;
- the blackboard stores data in-between tool function calls;
- the graphical orchestration defines tool chains and the data flow between tools;
- the shell allows for direct access to tool functions.

The SDE is available for download at [20]. The website also contains a tutorial for tool integration and videos demonstrating the SDE in action. Further information about the SDE can also be found in D2.2 [11] and D2.3 [14].

8.2 Integrated Security-related Tools

Each NESSoS partner is responsible for the integration of their own tools into the SDE and for investigating reasonable ways of orchestration using several tools. During the third year of the project further tools were integrated into the SDE, complementing the set of tools already integrated during the first two years of the project. A list of integrated tools and a brief description of them can be found in the following and at the NESSoS SDE site [20]:

AbsInt Verification tool for safety-critical software in embedded systems.

Arachni Vulnerability scanner, which helps penetration testers and administrators to evaluate the security of web applications.

Avantssar-atse (CL-ATSE) Constraint Logic based Attack Searcher for security protocols and services.

Avantssar Orchestrator Tool for automatic orchestration of Web Services taking into account their security policies. In short, it generates a service called mediator that is able to satisfy requests of a given client with the help of given community of available services.

CORAS Tool CORAS is a model-driven approach to risk analysis that consists of three tightly integrated building blocks, namely the CORAS method, the CORAS language and the CORAS tool.

EOS (Eye OCL Software) Java component for performing efficient evaluation of OCL expressions on medium-large size scenarios.

Jalapa Jalapa is a tools suite for the development of Java applications secured with local policies.

MagicUWE CASE tool that was created to support the development of web applications. It focuses on the modelling phase and uses the UML-based Web Engineering (UWE) methodology.

Nessus Vulnerability scanner, which audits configurations, patches and web applications.

Nexpose Vulnerability scanner, which proactively scans an environment for misconfigurations, vulnerabilities, and malware and provides guidance for mitigating risks.

Nikto Vulnerability scanner, which performs tests against web servers for multiple items, including potentially dangerous files, checks for outdated versions, and version specific problems.

PRRS The Platform for Runtime Re-configurability of Security (PRRS) is a tool that provides run-time management of Security and Dependability (S&D) solutions and monitoring of the system context.

SATMC generates a formula that represents all the possible evolutions of a transition system described by a security problem up to a given depth.
**STS-Tool** STS-ml is the Socio-Technical Security modelling language for the specification of security and trustworthiness requirements of systems operating in a cross-organizational environment.

**UML4PF** Tool to support requirements analysis and architectural design based on Michael Jackson’s problem frame approach.

**UWE2XACML** Transformer that converts UWE access control models to XACML policies.

**VeriFast** Verifier for single-threaded and multithreaded C and Java programs annotated with preconditions and postconditions written in separation logic.

**WS-TAXI** Using WS-TAXI, which is a combination of tools, allows to test a web service efficiently and automatically.

**XACML2FACPL** Transformer that converts XACML policy files to FACPL. FACPL stands for Formal Access Control Policy Language and it provides a more manageable alternative syntax to XACML and has a solid semantic ground.

**X-CREATE** Tool to test XACML policy evaluation engines and access control policies by systematic generation of a test suite of requests

As one can see, the SDE does not only comprise tools from NESSoS partners, but also tools from outside. Examples are AbsInt, Arachni, Nessus, Nikto and Nexpose, which were wrapped for the use from within the SDE by NESSoS (associated) partners, but developed by several companies.

Further information about the integrated tools and about the use of the tools’ wrappers can be found in the CBK.

### 8.3 Connections between CBK and SDE

For the NESSoS project the team play of the CBK and the SDE (Figure 8.2) is an advantage, because the CBK contains information about all the tools – not only those that are already integrated in the SDE. In the SDE the information about tools and their functions can be reused for the integration of a tool. Furthermore, the CBK allows to compare several of them in order to find those that might be used in a tool chain. Naturally, not all tools fit together immediately and it might be necessary to write some converters for the input/output of the tools in order to be able to connect them. In order to ease this task, possible applications for each tool are described in the CBK’s example section. Consequently, the researchers can become acquainted with a new tool by straightforwardly working with it and some example data using the SDE workbench.

![Figure 8.2: Interplay of CBK and SDE.](image)

A typical work-flow is outlined in the following (Researchers should have Eclipse & SDE installed. For a good start, the NESSoS SDE site [20] provides an Eclipse installation with most of the security-related
tools preinstalled. We also provide a Virtual Machine Image in the NESSoS NESSoS Cloud Execution Environment with the SDE and most of the integrated tools installed, see section 9.4.):

- Tool owners describe their tools in the CBK and integrate some of them in the SDE.
- Based on the tools' descriptions in the CBK and once a researcher has installed a SDE platform (using the update site link from within the CBK) he is able to try out an example that is described in detail in the CBK. For many integrated tools, example projects can be imported, which contain information how integrated functions can be executed using given data.
- Using the contact data of the CBK tool page, the researcher can contact the tool owner, in order to discuss if the output of that tool can be connected with his tool.
- If they realise that they need a converter between the output of the first and the input of the second tool, a new software that performs the conversion can be written and included in the SDE as tool.
- Finally, the three resulting tools are orchestrated and executed in a tool chain.
- Afterwards, this tool chain will be described in the CBK.
- Everyone can use the tool chain being only aware of the input to be provided and the final results it provides (there is no need to become familiar with the details of the orchestrated tools).

These collaborations are thought to foster fruitful research ideas to come up in such a way that researchers are encouraged to further develop their methodologies and tools.
9 NESSoS Cloud Execution Environment (CEE)

Cloud systems are becoming very common in the last years because they allow users to exploit the resources or services they need for their computation exactly when they need them and for the time they need them. The user is then relieved from the burden of managing those resources (e.g., configuring and managing the physical machines that provide services to the users), because this is a task of the Cloud service provider, so he can focus on his specific business.

A Cloud Execution Environment is provided as a part of the JVRL to ease the cooperation among the NESSoS users, because they can exploit this environment to easily share their tools and software.

In the following, we give a brief description of the main features of the Cloud, of a very common Open Source software to deploy Cloud IaaS services (OpenNebula), and of the Cloud Execution Environment that has been deployed in the Pisa data centre of CNR.

9.1 Introduction to Cloud

A definition of Cloud systems has been given in [8]. The most notable advantages of Cloud systems is the big computational power available on demand, and the possibility to get further resources or to release unused ones while the computation is running. As a matter of fact, the Cloud environment allows the user to exploit the resources he needs for his computation only when he actually needs them. Distinct Cloud service models are available, depending on the kind of resources or services provided to users. Here, we focus on the Infrastructure as a Service (IaaS) model, the one supported by the NESSoS CEE, where the resources that are provided to users are virtual machines connected by a virtual network. When requesting virtual machines, users can choose the features of these machines, such as the computational power and the storage capacity, the most proper network configuration and operating system and can run on these virtual machines the applications they need. Once requested, the machines are available in few time and the number and/or the features of these machines can be updated by the users (increased or decreased) on demand during the computation according to their needs.

Cloud computing facilities are currently provided by several big companies (Public Clouds); among them Amazon [3, 4], Google [6], IBM [1], Microsoft [2] and others. As an example, users exploiting Amazon Elastic Compute Clouds (EC2) [3], request to the Cloud provider a number of virtual machines with some features, such as the operating system (e.g., see Amazon Machine Image, AMI [3]) and the network configuration (e.g., see Amazon Virtual Private Clouds, VPC).

Instead of using a Public Cloud, users can create their own Cloud systems by deploying a Cloud framework in their data centre, such as Eucalyptus, OpenNebula, OpenStack and others, thus exploiting their physical machines to host virtual ones (Private Clouds).

9.2 OpenNebula

OpenNebula.org [18] is a project that aims at defining a reference and open source tool for the management of IaaS Cloud Computing platforms [7, 9]. OpenNebula is the result of many years of research and development in the field of efficient and scalable management of virtual machines deployed on a large scale infrastructure, and partnerships with the community of users and with the major players in the field of Cloud Computing. Many of its innovative features have been developed to solve the problems arising from the use cases of companies and industries proposed in the context of international projects.

Currently, OpenNebula is an industry standard for IaaS Cloud Computing, offering a complete solution for managing virtualised data centres to create both private, public and hybrid Clouds. The interoperability of OpenNebula allows to create cloud services using existing resources in the data centre. OpenNebula code can be freely downloaded from its website, which offers a good documentation, covering both the installation and the use, consisting of web pages, manuals, white papers, wikis, and forums. In addition, there is also a commercial version of the software, called OpenNebulaPro, provided by C12G Labs3 that has a support of commercial type.

OpenNebula supports the most common virtualisation tools to manage virtual machines such as Xen, KVM and VMware. In this way, in addition to avoiding vendor lock-in, it is very easy to integrate OpenNebu-
ula in existing data centres. For the interaction with the user, or to enable the user to create and manage your virtual machines, OpenNebula is very flexible in that it provides not only a proprietary graphical interface, the GUI called Sunstone, but also compatible interface with the services of Amazon EC2 and OCCI interface. In addition, OpenNebula provides powerful integration APIs that facilitate the development of new components, such as new drivers to support virtualisation hypervisor. To simplify the creation of virtual machines, OpenNebula provides a repository for images, which allows you to create and share virtual machine images (which may be operating systems or data), and a repository for templates, which allows you to record definitions machines virtual, which can be later used to create new virtual machines. Once a template has been instantiated in a virtual machine, a number of operations can be done to control the life cycle, how to stop, suspend, reactivate, or migrate from a data centre to another. A network system easily adaptable and customisable in OpenNebula is well integrated with the specific requirements of the network of existing data centres. It also provided support for VLAN and Open vSwitch. From the security point of view, OpenNebula supports user and group accounts, as well as different mechanisms of authentication and authorisation. In addition, OpenNebula provides access control list, which supports the management of roles and fine-grained permissions. Finally, OpenNebula, beside the command line interface, also offers a graphical interface, SunStone, that allows administrators and users to easily manage their virtual and physical resources. A simplified version for users of that GUI, called SelfService, is also available.

### 9.3 NESSoS CEE

The NESSoS CEE is an OpenNebula-based Cloud IaaS system deployed in the Pisa data centre of CNR, that is shared with the NESSoS user as a JVRL component. As a matter of fact, the CEE provides to NESSoS's users the execution environment where their tools and software can be deployed and executed. The Cloud paradigm eases the task of tool and software sharing. As a matter of fact, each partner can act as a software provider, by preparing a Virtual Machine image properly configured for running his software, or can act as software user, and in this case he can run instances of the images prepared by the provider partners to exploit the software he need avoiding any installation and configuration effort.

The CEE runs over a cluster is composed by five machines. One master node and four slave nodes. The master node is based on two Intel Xeon CPU E5630 processors, each one is a quad-core CPU running at 2.53GHz. The master node has 64 GBytes of DDR3 RAM. The slaves have a different hardware configuration: each slave is based on a single Intel Xeon CPU E31230 quad-core processor running at 3.20 Ghz. Each slave node hosts 16 GBytes of DDR3 RAM.

All the nodes run as operating system Linux Ubuntu 12.04 server, 64-bit. For performing virtualisation the machines exploit Kernel-based Virtual Machine (KVM), a virtualisation infrastructure for the Linux kernel. KVM can be used only on processors with hardware virtualisation extensions, like the ones installed in the cluster machines. Some standard virtual machine images (with Linux Ubuntu 12.04 32 and 64 bit) are already available on the CEE, and can be exploited by the users by simply selecting them through the graphical interface. Users can also produce their customized virtual machines images starting from the standard ones. Moreover, a virtual machine images including the tools developed in the NESSoS project has been configured, as described in the next section.

The OpenNebula version currently deployed on the CEE is v3.4, and we plan to update to version 4 in a few months.

The authentication system of OpenNebula has been configured to exploit the NESSoS LDAP service, thus allowing NESSoS's users to access the CEE and exploit the Cloud services provided by Conero exploiting the usual NESSoS login and password.

NESSoS's user can easily access and exploit the resources provided by the CEE for their computation through the OpenNebula graphical interface called SelfService, that has been customized for NESSoS (see Figure 9.1. The interface is accessible through a web browser (e.g., Mozilla Firefox or Google Chrome), at the URL: conero.isti.cnr.it:4567/ui.

In the last year, the CEE has been up and running 24 hour a day, 7 days a week, except for a few number of stop due to software updates.
9.4 NESSoS Virtual Machine Image (VMI)

In order to ease the exploitation of the tools provided by the NESSoS project, a customized Virtual Machine Image has been configured and uploaded on the NESSoS CEE. Hence, NESSoS users can select this image (see Figure 9.2) when they instantiate their new virtual machine.

The Operating system version installed on the NESSoS VMI is Linux Ubuntu server 12.04 LTS 64bit, with Gnome2 graphical interface. The machine is equipped with Java openjdk-7-jdk, and Eclipse Indigo Service Release 2 Build Id: 20120216-1857.

The Nessos tools installed on NESSoS VMI are:

- AbsInt
- Arachni
- Avantssar-atse (CL-ATSE)
- Avantssar Orchestrator
- CORAS Tool
- EOS (Eye OCL Software)
- Jalapa
- MagicUWE
- Nessus
- Nexpose
- Nikto
The login and password of the provided VMI are `ubuntu/ubuntu`, and the user must change them as soon as he creates his own VM instance, in order to prevent other users to access his VM instance. Obviously, the changes that are done on one VM instance do not affect the other VM instances.

The user can customize his VM instance by uploading his data and installing his own applications and tools, and before stopping the VM, he can save it in case he wants to exploit it again in the future.
10 Mailing lists

As mentioned in previous Deliverables 1.2 [10] and 1.3 [13], specific mailing lists support e-mail communication among partners. The project management team maintains such mailing lists. Here, for the sake of completeness, we remind the reader the description and functionalities of all the mailing lists, and we give some statistic data about the traffic and the number of subscribers of the main mailing lists, covering the period September 1st, 2012 – August 20th, 2013.

Purely project related mailing lists are denoted with the suffix @iit.cnr.it. The NESSoS community at large has its own mailing lists with the suffix nessos-project.eu. The same suffix is for the list devoted to communication with the webmaster (webmaster@nessos-project.eu) as well as for the list for asking generic information (info@nessos-project.eu). Table 10.1 presents the lists.

Lists with suffix @iit.cnr.it are closed. Only CNR people can add registrants, and then only members that are subscribed to a list can send/receive e-mails. To prevent an avalanche of unsolicited messages, senders shall carefully target their messages to the narrowest audience, as reasonably possible. For changes, deletions, or additions of mailing lists, the interested reader should contact the PMT.

The mailing list nessos@iit.cnr.it has 6 subtopics: WP6, WP7, WP8, WP9, WP10, WP11. Senders may subscribe to a subtopic as follows:

- going to the http://www.nessos-project.eu/mls Login Web Page. A user without a listserv password may just ask for a new password following the appropriate link;
- in the subscriber corner, the mailing lists may be chosen (generally users will be already subscribed to many for NESSoS) in the settings page, by ticking the checkboxes for the appropriate topics. The Other tick means all the other WPs;
- when sending an email to the whole list, a user should just send it to nessos@iit.cnr.it;
- when sending an email to a specific Work Package (i.e., to a specific subtopic), users should send it to nessos@iit.cnr.it and put in the subject WPx: (where x = 6...11) and the usual subject.

In addition, community@nessos-project.eu is the mailing list for the open research community. Registration to the list is open, and it is available from the project public web site (http://www.nessos-project.eu/).

Finally, webmaster@nessos-project.eu is the mailing list to ask for modifications to the project public web site (apart from publications, news, and events that can be inserted by the users themselves).

During the period under investigation (September 2012 – August 2013), 140 emails have been sent over the main working mailing list nessos@iit.cnr.it. Until August 20th, 2013, the subscribers to this list were 88. During the same period, 14 emails have been sent over the Community list community@nessos-project.eu. Until August 20th, 2013, the subscribers to the Community list were 281.

Table 10.1: NESSoS Mailing Lists

<table>
<thead>
<tr>
<th>Mailing List</th>
<th>Description</th>
<th>Addressees</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:nessos@iit.cnr.it">nessos@iit.cnr.it</a></td>
<td>The main working mailing list</td>
<td>All project members</td>
</tr>
<tr>
<td><a href="mailto:nessos-wpl@iit.cnr.it">nessos-wpl@iit.cnr.it</a></td>
<td>For WP leaders and their proxy</td>
<td>All WP leaders and their proxy</td>
</tr>
<tr>
<td><a href="mailto:nessos-ga@iit.cnr.it">nessos-ga@iit.cnr.it</a></td>
<td>For the General Assembly</td>
<td>All GA members</td>
</tr>
<tr>
<td><a href="mailto:nessos-afm@iit.cnr.it">nessos-afm@iit.cnr.it</a></td>
<td>For administration and financial issues</td>
<td>Persons in charge of management and financial aspects</td>
</tr>
<tr>
<td><a href="mailto:nessos-associated@iit.cnr.it">nessos-associated@iit.cnr.it</a></td>
<td>For the NESSoS associated partners</td>
<td>All associated partners</td>
</tr>
<tr>
<td><a href="mailto:nessos-iab@iit.cnr.it">nessos-iab@iit.cnr.it</a></td>
<td>For the NESSoS Industry Advisory board</td>
<td>All the Industry Advisory board</td>
</tr>
<tr>
<td><a href="mailto:nessos-pmt@iit.cnr.it">nessos-pmt@iit.cnr.it</a></td>
<td>For the project management team at CNR</td>
<td>All the project management team members</td>
</tr>
<tr>
<td><a href="mailto:community@nessos-project.eu">community@nessos-project.eu</a></td>
<td>General list for events of interest</td>
<td>All the subscribers</td>
</tr>
<tr>
<td><a href="mailto:webmaster@nessos-project.eu">webmaster@nessos-project.eu</a></td>
<td>For communication with the webmaster</td>
<td>All the subscribers</td>
</tr>
<tr>
<td><a href="mailto:info@nessos-project.eu">info@nessos-project.eu</a></td>
<td>For requesting generic information</td>
<td>All the subscribers</td>
</tr>
</tbody>
</table>
11 Subversion repository

The SVN repository manages the shared documentation and the project resources which require cooperative work. The SVN consists of a folders tree and a list of commands. The tree root contains all the project files and folders. It is accessible from a link within the web portal, at http://www.nessos-project.eu/svn. In particular, the root folder contains general project files, e.g., the description of work\footnote{http://www.nessos-project.eu/svn/DoW.pdf} and a sub-folder for each work package (WP). The WP folders are meant to contain the files and folders used by the WP contributors for the cooperating activities, e.g., WP deliverables.

The SVN repository is extremely active and frequently used among the JVRL components. Tools, papers and deliverables are constantly written or developed using the cooperation and versioning facilities of Subversion. As of 2013, July 30\textsuperscript{th}, the repository is at version 3864, with commits being split among the partners as shown in Table 11.1.

<table>
<thead>
<tr>
<th>Partner</th>
<th>Commits</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNR</td>
<td>230</td>
</tr>
<tr>
<td>ATOS</td>
<td>127</td>
</tr>
<tr>
<td>ETH</td>
<td>226</td>
</tr>
<tr>
<td>IMDEA</td>
<td>132</td>
</tr>
<tr>
<td>INRIA</td>
<td>225</td>
</tr>
<tr>
<td>KUL</td>
<td>122</td>
</tr>
<tr>
<td>LMU</td>
<td>675</td>
</tr>
<tr>
<td>Siemens</td>
<td>448</td>
</tr>
<tr>
<td>SINTEF</td>
<td>405</td>
</tr>
<tr>
<td>UDE</td>
<td>570</td>
</tr>
<tr>
<td>UMA</td>
<td>449</td>
</tr>
<tr>
<td>UNITN</td>
<td>314</td>
</tr>
</tbody>
</table>

Table 11.1: SVN commits by partner.

During the third year of the project, the SVN repository has been enhanced with LDAP authentication, thus merging with the single sign-on facilities of the JVRL. The same credentials used in the NESSoS web portal and other JVRL components are now usable to access the SVN repository as well.
12 LDAP directory

LDAP is used in NESSoS as a Single Sign-On (SSO) facility. This is because most of the main cooperation tools are bundled with a binding for LDAP, which generally is used for authentication:

- Joomla!, the CMS used for hosting the web portal (Chapter 4), has an efficient support for LDAP, partly in the native build and partly through plugins;
- MediaWiki, the technology underlying the CBK (Chapter 5), natively supports LDAP as one of the possible authentication engines;
- the Virtual Education Centre (Chapter 7), based on Moodle, was implemented right out with LDAP authentication;
- OpenNebula, the technology supporting the CEE (Chapter 9) also supports LDAP as an authentication service;
- the BSCW tool provides support for LDAP;
- even the SVN server sports out LDAP support, when used over Hypertext Transfer Protocol (HTTP).

These softwares can connect to LDAP for authentication, instead of using their own databases. Users stored in the LDAP directory have a “password” attribute, hidden and encrypted, which serves the purpose.

Most of the JVRL components have been configured to support LDAP authentication. The web portal, the CEE and the Virtual Education Centre were deployed using LDAP as their primary authentication engine since their initial releases, after an initial (pre-deployment) testing phase during which native authentication was used. The CBK, after an initial deployment with an autonomous authentication system and individual credentials for each partner, was moved to full LDAP support and can be accessed with the same credentials used for other JVRL components. Finally, the SVN repository, after a long period of accessibility using individual credentials, has been moved to LDAP authentication. The only tool that has not been integrated with LDAP for authentication is the BSCW, mainly for liability purposes, since it is not a resource dedicated to NESSoS but it also hosts other projects, and unwanted interaction between NESSoS and non-NESSoS resources suggested to maintain proprietary credentials for the BSCW.

Table 12.1 summarizes the integration status of the JVRL components into the LDAP. Additional details will be provided in Chapter 13.

<table>
<thead>
<tr>
<th>JVRL component</th>
<th>LDAP-enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Portal</td>
<td>Yes</td>
</tr>
<tr>
<td>Common Body of Knowledge</td>
<td>Yes</td>
</tr>
<tr>
<td>Mobility portal</td>
<td>Yes</td>
</tr>
<tr>
<td>Virtual Education Centre</td>
<td>Yes</td>
</tr>
<tr>
<td>SDLC tools workbench (SDE)</td>
<td>Not needed</td>
</tr>
<tr>
<td>BSCW</td>
<td>No</td>
</tr>
<tr>
<td>Cloud Computing Environment (CEE)</td>
<td>Yes</td>
</tr>
<tr>
<td>Mailing lists</td>
<td>Not needed</td>
</tr>
<tr>
<td>Subversion repository</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 12.1: Status of LDAP integration.

The structure of the LDAP directory in itself is quite simple. Figure 12.1 is an excerpt of an administrative UI used for managing the NESSoS LDAP. The root DN is dc=nessos-project,dc=eu, and the bulk of the content are the members’ data, contained in uid=username,ou=members,dc=nessos-project,dc=eu. Additionally, the moodle organisational unit has been created, with two POSIX groups, cn=creators and cn=students, used by the Virtual Education Centre (VEC) to identify different roles for users. An organisational unit called cbkguests has also been created to allow the addition of users to the CBK without providing them with credentials for the whole JVRL. This feature is under implementation at the time of this writing.

The current structure for the LDAP directory is final.
Figure 12.1: The NESSoS LDAP tree.
13 Integration

Although consisting of a set of tools which can be very useful per se, there is more to the JVRL than just individual tools. The main point of the JVRL is to collect these tools within a unique, integrated infrastructure whose components can communicate and interact among themselves. This is the main focus of WP1, and the final target of the Work Package over the course of the NESSoS project.

The collaborative development of the JVRL results in an allocation of components which is summarised in Table 13.1.

<table>
<thead>
<tr>
<th>JVRL component</th>
<th>Providing partner</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web portal</td>
<td>CNR</td>
</tr>
<tr>
<td>Common Body of Knowledge</td>
<td>UDE</td>
</tr>
<tr>
<td>Mobility portal</td>
<td>IMDEA</td>
</tr>
<tr>
<td>Virtual Education Centre</td>
<td>UMA</td>
</tr>
<tr>
<td>SDLC tools workbench (SDE)</td>
<td>LMU</td>
</tr>
<tr>
<td>BSCW</td>
<td>UNITN</td>
</tr>
<tr>
<td>Cloud Computing Environment (CEE)</td>
<td>CNR</td>
</tr>
<tr>
<td>Mailing lists</td>
<td>CNR</td>
</tr>
<tr>
<td>Subversion repository</td>
<td>CNR</td>
</tr>
<tr>
<td>LDAP directory</td>
<td>CNR</td>
</tr>
</tbody>
</table>

Table 13.1: JVRL components by partner.

13.1 Integration status

Integration-wise, there have been several steps forward with respect to the previous year. Although made up of different tools, with different purposes, and maintained by different people, the JVRL is supposed to give its user a feeling of something bigger than the individual component.

Progress has been made toward a single sign-on procedure. As already mentioned in Chapter 12, the LDAP directory is being used as a database for login credentials. Most of the components of the JVRL have been designed or extended to support authentication via the LDAP directory:

- the web portal (Chapter 4) was designed with LDAP support as its only means of authentication from the beginning, except in its early stages of development;
- the Virtual Education Centre (Chapter 7) underwent a long development and internal testing phase during which it had its own individual authentication; however, the deployed service supported LDAP authentication from the beginning. The LDAP structure was modified to accommodate the needs of the VEC’s developers;
- the Common Body of Knowledge (Chapter 5) was developed using its own authentication method. For the first year of deployment, the CBK used an autonomous authentication procedure. However, it has recently been extended to support LDAP authentication, and now it is possible to use the CBK with both authentication methods;
- the mobility portal (Chapter 6) was also enhanced to support the LDAP authentication;
- the Subversion repository (Chapter 11) has recently been updated with LDAP support;
- the CEE was introduced into the JVRL with LDAP credentials straight out.

Not all services are hooked to LDAP yet, but for some of them this is not supposed to be done. Most notably:

- the BSCW is not going to be integrated with LDAP, because that service also has a number of users not related to NESSoS, and for this reason it was chosen to have a separate authentication method;
• additionally, the mailing lists do not have any form of authentication; although it would theoretically be possible to have the mailing lists retrieve the e-mail addresses from the LDAP directory and use Sendmail\(^1\) to assign mailing list users based on LDAP attributes, this approach, albeit interesting from a technological point of view, is probably excessive and too complicated for a simple purpose like managing a few mailing lists with a limited number of users.

The SDE is also strictly connected to the CBK. The tools provided by the partners are described in detail in the CBK, with examples, guides, requirements and so on. The tools implemented in the SDE are a part of those described in the CBK, and the objective is to have all of them implemented by the end of the project. More details are given in the deliverables for WP2 and WP5.

The integration of the JVRL has led many NESSoS members, who initially were operating on a single JVRL component, to request a more generalized access to other components. In particular, the number of actual JVRL users among NESSoS members grew from the initial number of 13 (one per partner plus the administrator) to 44 as additional components were being integrated into the JVRL. This number obviously represents only the registered users of the JVRL technologies, and does not account for the visitors of the CBK (but these statistics are included in the Google Analytics reports), or non-registered individuals using the SDE tools, and so on. For reports on the usage of individual components of the JVRL, the deliverables regarding those components should be referred to.

### 13.2 Usage analytics

Since the first year of the project, the NESSoS web portal has been enhanced with usage analysis. Statistics are collected using Google Analytics, and reports are shown periodically in the deliverables of WP12.

In recent times, the need has emerged to collect statistics from other components of the JVRL as well. The main interest was toward the CBK and the SDE, which are the components with the highest public visibility after the web portal. The integration of usage analytics with these two components required to add some scripts to the websites’ source code, allowing Google Analytics to collect data from their traffic as well.

However, both the SDE and the CBK are hosted by German partners, and German regulations presented an obstacle to this decision. For the SDE, the solution with LMU has not been found for the moment, so the integration of Google Analytics has not been carried out. On the other hand, UDE managed to solve the issue without going against German regulations, and the CBK statistics are now being collected together with those related to the web portal.

In case future needs arise for the missing JVRL components, or if LMU manages to find an acceptable solution to integrate the statistics without infringing German laws, the components will be hooked to Google Analytics accordingly.

### 13.3 Future integration

Some final integration activities will be carried out during the final months of the project, and in addition some plans are currently under development for the time after the end of the project’s life. In particular, one of the decisions that have already been made is to integrate an Eclipse update site into the JVRL, collecting all the tools that have been developed by the NESSoS partners and integrated into the SDE. For the time being, since many of these tools are under frequent development, the individual partners prefer to maintain their own update sites, but at the end of the project these tools will be frozen and they will be collected under a new Eclipse update site hosted on the same machine as the NESSoS web portal, and connected to the JVRL accordingly.

\(^1\)http://www.sendmail.com/sm/open_source/
14 Conclusions

The JVRL represents the technological core of the NESSoS project. The JVRL components have been key in progressing the project, and many of them had a use outside the project as well. A key to an efficient progress of the NESSoS project is having an efficient technological infrastructure, with reliable and updated tool that comply with the partners’ evolving needs. Most partners have contributed to the JVRL by providing something of their own, and work on the provided components is steady and reactive. Some partners have provided a licensed technology (such as BSCW), a previously-developed tool which is being maintained (like the SDE), or the effort needed to develop a service (such as Subversion, CBK or the VEC) or a web site (web portal, mobility portal). During the third year of the project, a major cooperation between the partners has increased the degree of integration between the various JVRL components.

To summarise the outcome of the three past years: during the first year, the JVRL components have been developed, made available or in any case been put into place by the partners, giving the other partners or the public the opportunity to benefit from them; in the second year, the major steps toward integration were undertaken, unifying a set of separate components into something that more closely resembled a single laboratory with many connected facilities; in the third years, the connection between these facilities has been strengthened, giving the project partners and the public a tighter view of the NESSoS tools as a network of interoperating components.

Although the project is reaching its final stages, as NESSoS is supposed to set up an enduring result in the perspective of secure service engineering, so the JVRL is expected to extend its usefulness beyond the duration of the project.

Although it is likely that there will be very little development on the JVRL in the last months and especially after the end of the project, the degree of maturity of the components and their integration ensures that they can provide their benefits in the future, both from the point of view of usability (especially regarding those components related to some state-of-the-art tool, such as the SDE or the CEE) and that of external visibility (e.g., web portal or CBK, the latter of which provides most of the documentation needed for the SDE tools).
Bibliography


